ADSTAR Distributed Storage Manager
for Sun** Solaris**

Administrator's Guide

Version 2

GC35-0263-00

Note!

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

First Edition (March 1997)

This edition applies to Version 2 of the ADSTAR Distributed Storage Manager for Sun Solaris (5639-B22) and to any subsequent releases until otherwise indicated in new editions or technical newsletters. 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 below.

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

IBM Corporation
Information Development, Department 61C
9000 South Rita Road
Tucson, AZ 85744-0001, 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 1995, 1997. 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

© Copyright IBM Corp. 1995, 1997

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 product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of the intellectual property rights of IBM 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 Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, New York 10594, U.S.A. Refer to the HONE SALESMANUAL or product announcement letters for the most current product information.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Information Enabling Requests, Dept. M13, 5600 Cottle Road, San Jose, CA 95193, U.S.A. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Programming Interface

This publication is intended to help the customer plan for and manage the ADSM server.

This publication also documents General-use Programming Interface and Associated Guidance Information, and Diagnosis, Modification or Tuning Information provided by ADSM.

General-use programming interfaces allow the customer to write programs that obtain the services of ADSM.

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

© Copyright IBM Corp. 1995, 1997

Diagnosis, Modification or Tuning Information is provided to help the customer to do diagnosis of ADSM.

Attention: Do not use this Diagnosis, Modification or Tuning Information as a programming interface.

Diagnosis, Modification or Tuning Information is identified where it occurs, either by an introductory statement to a chapter or section or by the following marking:

Diagnosis, Modification or Tuning Information

Diagnosis, Modification or Tuning Information...

End of Diagnosis, Modification or Tuning Information

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

ACF/VTAM
AD/Cycle
ADSTAR
Advanced Peer-to-Peer Networking
AIX
AIX/6000
AIXwindows
Application System/400
APPN
AS/400
AT
BookManager
C/370
CD Showcase
CICS
Common User Access
CUA
DATABASE 2
DB2/6000
DFSMS
DFSMS/MVS
DFSMSdss
ESCON
IBM
IBMLink
Language Environment
Library Reader
MVS/DFP
MVS/ESA
MVS/SP
MVS/XA
OpenEdition
Operating System/2
Operating System/400
OS/2
OS/400
POWERparallel
Proprinter
PS/2
RACF
RISC System/6000
RS/6000
SAA
SP2
System/370
System/390
Systems Application Architecture
SystemView
Virtual Machine/Enterprise Systems Architecture
Virtual Machine/Extended Architecture
VM/ESA
VM/XA
VSE/ESA
VTAM
WIN-OS/2

The following terms are trademarks of other companies:

Trademark	Company	Trademark	Company
Acrobat	Adobe Systems Inc.	NFS	Sun Microsystems, Inc.
Adobe	Adobe Systems Inc.	Novell	Novell, Inc.
Andataco	Andataco Corporation	Open Desktop	The Santa Cruz Operation, Inc.
Apple	Apple Computer, Inc.	OpenWindows	Sun Microsystems, Inc.
Attachmate	Attachmate Corporation	PARADOX	Borland International, Inc.
CompuServe	CompuServe, Inc.	PC/TCP	FTP Software, Inc.
dBASE	Borland International, Inc.	PTX	Sequent Computer Systems
DECstation	Digital Equipment Corporation	SCO	The Santa Cruz Operation, Inc.
DLT	Quantum Corporation	Sequent	Sequent Computer Systems
DPX/20	Groupe Bull	SINIX	Siemens Nixdorf Information Systems, Inc.
Dynatek	Dynatek Automation Systems		
DynaText	Electronic Book Technologies, Inc.	Solaris	Sun Microsystems, Inc.
Exabyte	Exabyte Corporation	Sony	Sony Corporation
Extra!	Attachmate Corporation	SPARC	SPARC International, Inc.
FOXPRO	Microsoft Corporation	StorageTek	Storage Technology Corporation
Hewlett-Packard	Hewlett-Packard Company	Sun	Sun Microsystems, Inc.
HP-UX	Hewlett-Packard Company	Sun Microsystems	Sun Microsystems, Inc.
Ice Box	Software International Microsystems	SunOS	Sun Microsystems, Inc.
iFOR/LS	Gradient Technologies, Inc.	Sun-3	Sun Microsystems, Inc.
INGRES	ASK Group, Inc.	Sun-4	Sun Microsystems, Inc.
Intel	Intel Corporation	SureStore	Hewlett-Packard Company
Iomega	Iomega Corporation	SyQuest	SyQuest Technology, Inc.
IPX/SPX	Novell, Inc.	Tivoli	Tivoli Systems, Inc.
IRIX	Silicon Graphics, Inc.	Tivoli Management Environment	Tivoli Systems, Inc.
Jetstore	Hewlett-Packard Company		
Lotus	Lotus Development Corporation	TME	Tivoli Systems, Inc.
Lotus Notes	Lotus Development Corporation	ULTRIX	Digital Equipment Corporation
Macintosh	Apple Computer, Inc.	WangDAT	Tecmar Technologies, Inc.
MacTCP	Apple Computer, Inc.	Windows 95	Microsoft Corporation
Motif	Open Software Foundation, Inc.	Windows NT	Microsoft Corporation
NetWare	Novell, Inc.	X Windows	Massachusetts Institute of Technology

C-bus is a trademark of Corollary, Inc.

Microsoft, Windows, and the Windows 95 logo are trademarks or registered trademarks of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names, which may be denoted by a double asterisk (**), may be trademarks or service marks of others.

Preface

ADSTAR Distributed Storage Manager (ADSM) is a client/server program that provides storage management solutions to customers in a multivendor computer environment. ADSM provides an automated, centrally scheduled, policy-managed backup, archive, and space-management facility for file servers and workstations.

Who Should Read This Publication

This guide is intended for anyone who has been assigned an ADSM administrator user ID and an administrative privilege class. While ADSM can be managed by a single administrator, administrative responsibilities can be divided among several people as an installation requires.

All of the administrator commands you need to operate and maintain ADSM can be invoked from a workstation connected to the server.

What You Should Know before Reading This Publication

You should be familiar with the operating system on which the server resides and the communication protocols required for the client/server environment.

For information on product requirements for ADSM, see *ADSTAR Distributed Storage Manager for Sun Solaris: License Information*. For information on installing ADSM, see *ADSTAR Distributed Storage Manager for Sun Solaris: Quick Start*.

You also need to understand the storage management practices of your organization, such as how you are currently backing up your workstation files and how you are using random access media and sequential access media.

Conventions Used in This Book

To help you recognize where example commands are to be entered, this book uses the following conventions:

- Command to be entered on the Sun Solaris command line:

```
> dsmlabel -drive=/dev/rmt/5mt
```

- Command to be entered on the command line of an administrative client:

```
query devclass
```

© Copyright IBM Corp. 1995, 1997

ADSTAR Distributed Storage Manager Publications

The ADSM publications are available softcopy on the product CD-ROM.

The following table lists ADSM publications.

Short Title	Publication Title	Order Number
ADSM Messages	*ADSTAR Distributed Storage Manager: Messages*	SH35-0133
ADSM License Information	*ADSTAR Distributed Storage Manager for Sun Solaris: License Information*	SC35-0261
ADSM Quick Start	*ADSTAR Distributed Storage Manager for Sun Solaris: Quick Start*	GC35-0262
ADSM Administrator’s Reference	*ADSTAR Distributed Storage Manager for Sun Solaris: Administrator’s Reference*	GC35-0264
ADSM Using the UNIX HSM Clients	*ADSTAR Distributed Storage Manager: Using the UNIX Hierarchical Storage Management Clients*	SH26-4030
ADSM V2 Using the Apple Macintosh Backup-Archive Client	*ADSTAR Distributed Storage Manager Version 2: Using the Apple Macintosh Backup-Archive Client*	SH26-4051
ADSM Using the UNIX Backup-Archive Clients	*ADSTAR Distributed Storage Manager Version 2: Using the UNIX Backup-Archive Clients*	SH26-4052
ADSM V2 Using the OS/2 Backup-Archive Client	*ADSTAR Distributed Storage Manager Version 2: Using the OS/2 Backup-Archive Client*	SH26-4053
ADSM V2 Using the DOS Backup-Archive Client	*ADSTAR Distributed Storage Manager Version 2: Using the DOS Backup-Archive Client*	SH26-4054
ADSM V2 Using the Novell NetWare Backup-Archive Client	*ADSTAR Distributed Storage Manager Version 2: Using the Novell NetWare Backup-Archive Client*	SH26-4055
ADSM V2 Using the Microsoft Windows Backup-Archive Clients	*ADSTAR Distributed Storage Manager Version 2: Using the Microsoft Windows Backup-Archive Clients*	SH26-4056
ADSM Using the Lotus Notes Backup Agent	*ADSTAR Distributed Storage Manager: Using the Lotus Notes Backup Agent*	SH26-4047
ADSM Installing the Clients	*ADSTAR Distributed Storage Manager: Installing the Clients*	SH26-4049
ADSM Client Reference Cards	*ADSTAR Distributed Storage Manager: Client Reference Cards*	SX26-6013
ADSM Using the Application Programming Interface	*ADSTAR Distributed Storage Manager: Using the Application Programming Interface*	SH26-4002

Short Title	Publication Title	Order Number
ADSM AFS/DFS Backup Clients	*ADSTAR Distributed Storage Manager AFS/DFS Backup Clients*	SH26-4048

Related IBM Hardware Products Publications

The following table lists related IBM hardware products publications.

Short Title	Title	Order Number
IBM 3490 Tape Subsystem User's Guide	*IBM 3490 Magnetic Tape Subsystem Enhanced Capability Models E01 and E11 User's Guide*	GA32-0298
IBM SCSI Device Drivers: Installation and User's Guide	*IBM SCSI Device Drivers: Installation and User's Guide*	GC35-0154
IBM 3494 Media Library Device Driver for AS/400 User's Guide	*IBM 3494 Tape Library Dataserver User's Guide: Media Library Device Driver for AS/400*	GC35-0153
IBM 3494 Operator's Guide	*IBM 3494 Tape Library Dataserver Operator's Guide*	GA32-0280
IBM 3590 Tape Subsystem User's Guide	*IBM 3590 High Performance Tape Subsystem User's Guide*	GA32-0330
IBM 3495 Operator's Guide	*IBM 3495 Tape Library Dataserver Models L20, L30, L40, and L50 Operator's Guide*	GA32-0235
IBM 3570 Operator's Guide	*IBM 3570 Magnetic Tape Subsystem Operator's Guide*	GA32-0345

IBM International Technical Support Center Publications (Redbooks)

The International Technical Support Center (ITSC) publishes redbooks, which are books on specialized topics such as using ADSM to back up databases. You can order publications through your IBM representative or the IBM branch office serving your locality. You can also search for and order books of interest to you by visiting the IBM Redbooks home page on the World Wide Web at this address:

http://www.redbooks.ibm.com/redbooks

Software Developer's Program

The IBM Storage Systems Division (SSD) Software Developer's Program provides a range of services to software developers who want to use the ADSM application programming interface (API). Information about the SSD Software Developer's Program is available in:

- IBMSTORAGE forum on CompuServe

- SSD Software Developer's Program Information Package

To obtain the Software Developer's Program Information Package:

1. Call 800-4-IBMSSD (800-442-6773). Outside the U.S.A., call 408-256-0000.
2. Listen for the Storage Systems Division Software Developer's Program prompt.
3. Request the Software Developer's Program Information Package.

Do You Have Comments or Suggestions?

If you have difficulty using this publication or if you have comments and suggestions for improving it, please complete and mail the reader's comment form found in the back of this publication. Your comments and suggestions can contribute to the quality and usability of this publication.

You can send us comments electronically by using these addresses:

- IBMLink from U.S.: STARPUBS at SJEVM5
- IBMLink from Canada: STARPUBS at TORIBM
- IBM Mail Exchange: USIB3VVD at IBMMAIL
- Internet: starpubs@vnet.ibm.com (or starpubs at vnet.ibm.com)
- Fax from U.S. and Canada: 520 799-2906
- Fax from other countries: (1) 520 799-2906

Translations

Selected ADSM publications have been translated into languages other than American English. Contact your IBM representative for more information about the translated publications and whether these translations are available in your country.

Summary of Changes for ADSTAR Distributed Storage Manager

This section summarizes changes made for this edition of this book.

Changes for Version 2—March 1997

The new functions for ADSM Version 2 are:

Database backup and recovery
You can perform full and incremental backups of the server database to protect against loss or damage. You can use the backup copies to restore the database to its current state or to a specific point in time. You can back up the database while the server is available to clients.

Note: To allow for recovery of the database to its most current state, you may have to extend your recovery log space significantly.

See Chapter 16, “Protecting and Recovering Your Data” on page 327 for details.

Storage pool backup and recovery
You can back up client files stored on storage pools to sequential media. These media can be either onsite, to protect against media loss, or offsite, for disaster recovery purposes. See Chapter 8, “Managing Storage Pools” on page 107 for details.

Disaster Recovery Manager
The Disaster Recovery Manager (DRM) feature allows you to prepare for and helps you to recover from disasters that destroy the ADSM server and clients. See Chapter 17, “Using Disaster Recovery Manager” on page 359 for details.

Administrative command scheduling
You can define schedules for automatically issuing administrative commands once or periodically. See Chapter 11, “Automating Operations” on page 223 for details.

Hierarchical storage management
Hierarchical storage management (HSM) provides space management services to HSM clients. HSM clients can automatically migrate user files to storage pools to free up client storage space. A user can access a migrated file as if it were on local storage. See Chapter 10, “Managing Policies” on page 185 for details.

Device support enhancements
Device class types now include support for IBM 3570 and 3590 tape drives, DLT drives, rewritable optical drives, and write-once (WORM) optical drives.

Library device support now allows the following:

- The user can select whether media labels are read when volumes are checked in and checked out.
- ADSM can initiate a swap operation when an empty library slot is not available during check-in processing.

© Copyright IBM Corp. 1995, 1997

See “Managing the Volume Inventory in Automated Libraries” on page 76 for details.

ADSM’s support for devices is now licensed. See “Licensed Features” on page 280 for details.

Shared memory communication protocol
ADSM supports the shared memory communication protocol for communication between the server and a client on the same machine.

Part 1. ADSM Basics

© Copyright IBM Corp. 1995, 1997

Chapter 1. Introducing ADSTAR Distributed Storage Manager

ADSTAR Distributed Storage Manager (ADSM) is an enterprise-wide storage management application for the network. It provides automated storage management services to multivendor workstations, personal computers, and local area network (LAN) file servers. ADSM includes the following components:

Server
Allows a host system to provide backup, archive, and space management services to workstations. The server maintains a database and recovery log for ADSM resources, users, and user data.

The server controls the ADSM server storage, or storage pools. These are groups of random and sequential access media that store backed up, archived, and space-managed files.

Administrative client
Allows administrators to control and monitor server activities, define management policies for client files, and set up schedules to provide services at regular intervals.

Backup-archive client
Allows users to maintain backup versions of their files, which they can restore if the original files are lost or damaged. Users can also archive files for long-term storage and retrieve the archived files when necessary. Users themselves or administrators can register workstations and file servers as client nodes with an ADSM server.

Hierarchical storage management (HSM) client
Provides space management services for workstations on some platforms. ADSM users can free workstation storage by migrating less frequently used files to server storage. These migrated files are also called *space-managed files*. Users can recall space-managed files automatically simply by accessing them as they normally would.

Application programming interface (API)
Allows users to enhance existing applications with back up, archive, restore, and retrieve services. When users install the ADSM application client on their workstations, they can register as client nodes with an ADSM server.

© Copyright IBM Corp. 1995, 1997

Figure 1 shows an example of an ADSM client/server environment. In this example, an administrator monitors the system from a workstation on which the administrative client program has been installed.

The backup-archive client program and HSM client program have been installed on workstations connected through a LAN and registered as client nodes. From these client nodes, users can back up, archive, or migrate files to the server.

Based on ADSM policies assigned to files, the server stores client files on disk, optical, or tape volumes in server storage, which are grouped into storage pools.

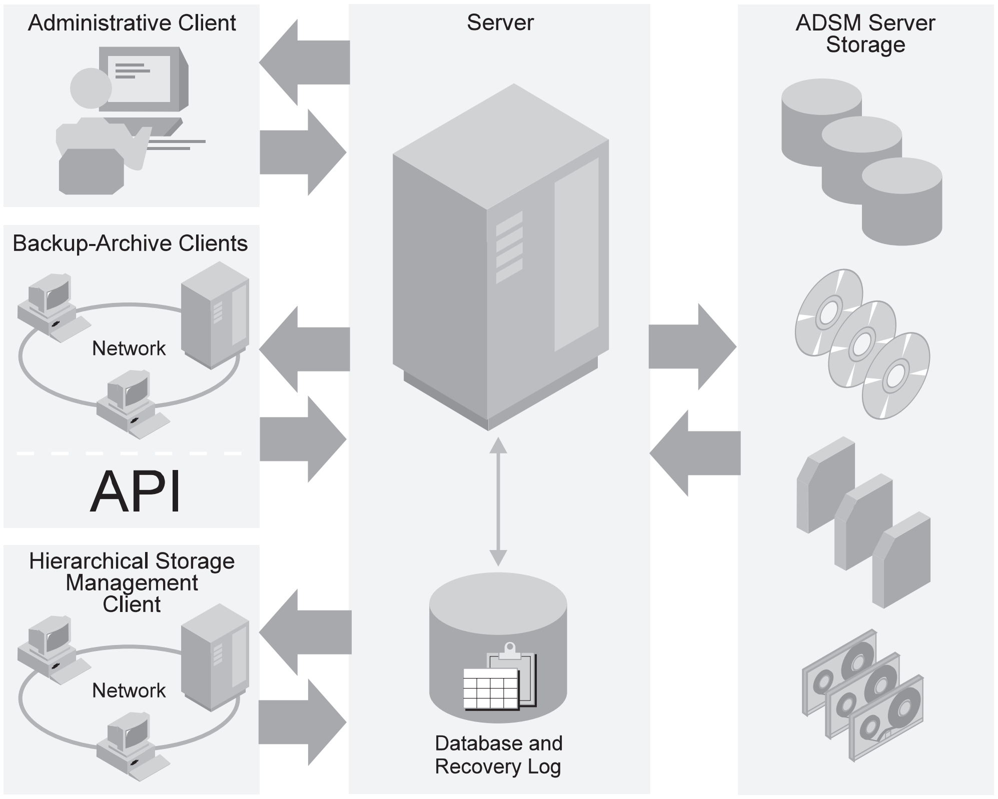


Figure 1. Sample Client/Server Environment

The rest of this chapter presents key ADSM concepts and information about storage for ADSM. It describes how ADSM manages client files based on information provided in administrator-defined policies, and manages devices and media based on information provided in administrator-defined ADSM storage objects.

How ADSM Stores Client Data

ADSM policy governs storage management including:

Backup
Copying files from client workstations to server storage to ensure against loss of data. Copies of multiple versions of a file can be stored.

Archiving
Copying files from client workstations to server storage for long-term storage.

Space Management
Freeing up client storage space by copying a file from client workstations to server storage. This process is also called client hierarchical storage management (client HSM). On the client, the original file is replaced with a stub file that points to the original in server storage. The process of moving the client file to server storage is also called **migration**.

Policy is defined by administrators in policy objects: policy domains, policy sets, management classes, and backup and archive copy groups. When you install ADSM, you have a set of policy objects named STANDARD. For information about this default policy, see “Using the Standard Storage Management Policies” on page 198.

Figure 2 on page 6 shows an overview of the ADSM process for storing client data. When users back up, archive, or migrate files, ADSM does the following:

1 Determines where to store the file

ADSM checks the management class bound to the file to determine the destination of the file, that is, where the file should be stored. The storage destination is an ADSM storage pool, which can be a group of disk volumes, tape volumes, or optical volumes. For backed up and archived files, storage destinations are assigned in the backup and archive copy groups, which are within management classes. For space-managed files, storage destinations are assigned in the management class itself.

See Chapter 10, “Managing Policies” on page 185 for information on assigning storage destinations in copy groups and management classes, and binding management classes to client files.

2 Stores information about the file in the ADSM database

ADSM saves information in the ADSM database about each file that it backs up, archives, or migrates. This information includes the file name, file size, file owner, management class, copy group, and location of the file in ADSM server storage.

See Chapter 13, “Managing the Database and Recovery Log” on page 261 for information on managing the database.

3 Stores the file in ADSM server storage

ADSM stores files from backup-archive clients and HSM clients on media that are associated with ADSM storage pools. The media can be disk, optical disk, or tape volumes.

For information about storage pools and storage pool volumes, see Chapter 8, “Managing Storage Pools” on page 107 and “Storage Pool Volumes” on page 160.

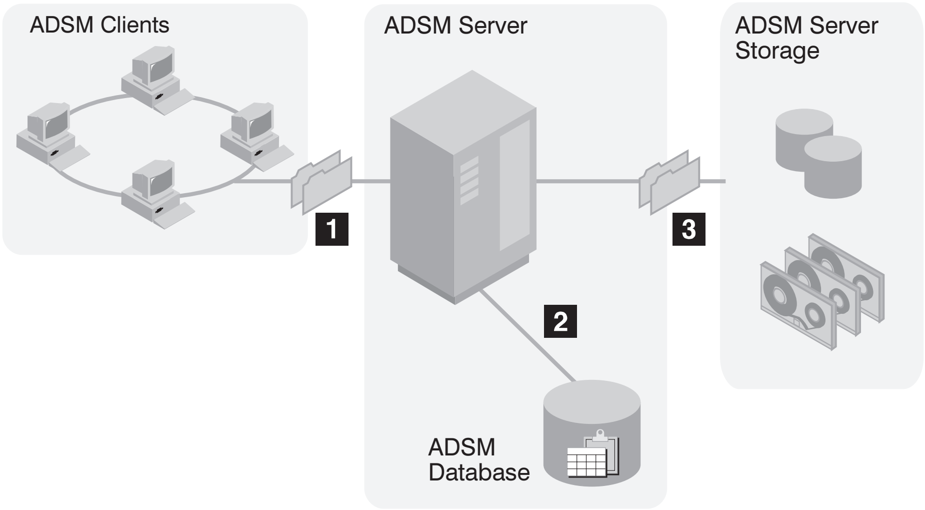


Figure 2. Overview of How ADSM Stores Client Data

Figure 3 on page 7 shows in more detail the interaction between ADSM policy objects and ADSM backup, archive, and migration operations.

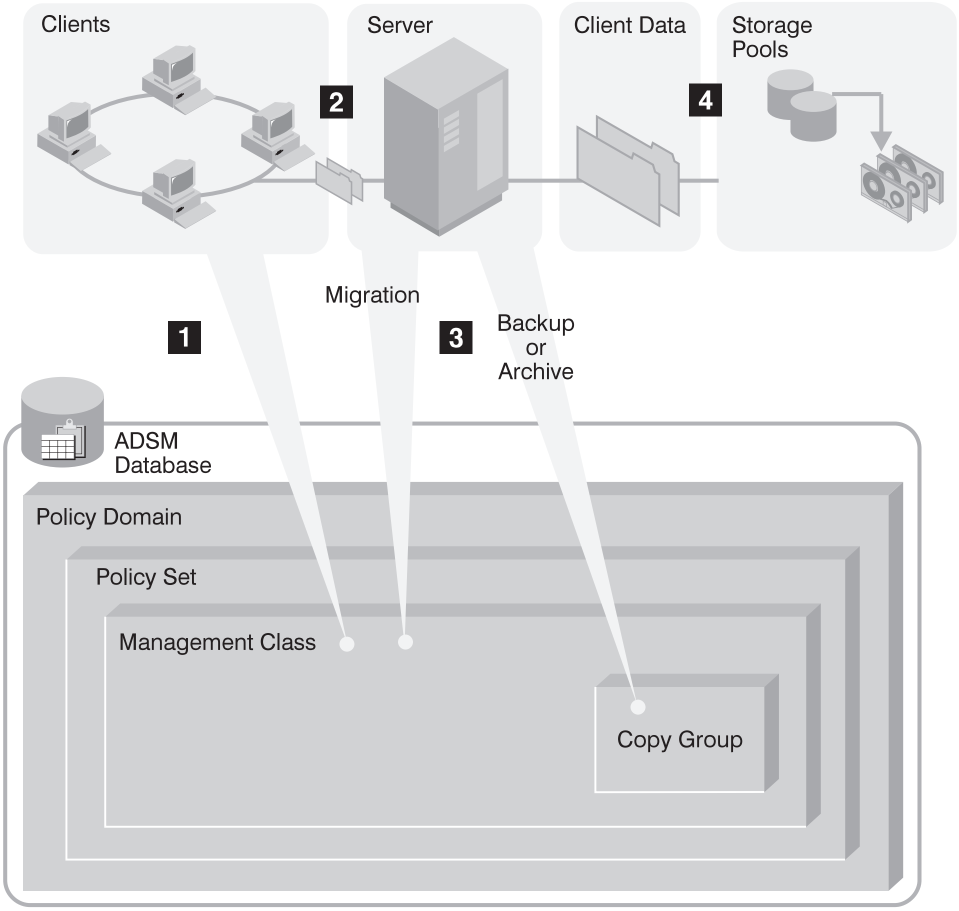


Figure 3. How ADSM Controls Backup, Archive, and Migration

1 An ADSM client initiates a backup, archive, or migration operation. The file involved in the operation is bound to a management class. The management class is either the default or one specified for the file in the client's include-exclude list.

2 If the file is a candidate for backup, archive, or migration based on information in the management class, the client sends the file and file information to the server.

3 The server checks the management class that is bound to the file to determine where to store the file within ADSM server storage. The storage destination for space-managed files is contained in the management class. The storage destination for backed up and archived files is contained in the copy groups, which are associated with the management class.

4 The server stores the file in the ADSM storage pool identified as the storage destination. Information about the file is stored in the server database.

If server storage is structured in a hierarchy, ADSM can later migrate the file to a different storage pool. For example, server storage may be set up so that ADSM migrates files from a disk storage pool to tape volumes in a tape storage pool.

Files remain in server storage until they expire and expiration processing occurs, or until they are deleted. A file expires because of criteria set in policy or because the file is deleted from the client's file system.

How ADSM Represents Devices

ADSM represents physical devices with administrator-defined ADSM storage objects: the device class, the library, and the drive. The storage objects, defined when devices are configured for ADSM, contain information for the management of devices and media.

At a minimum, each device requires a device class. Key factors in determining whether a library and drive object are also necessary are:

- Whether the device accesses the data on its media randomly or sequentially. Random access devices do not require the library and drive objects.
- Whether the device uses removable media. Most devices that use removable media require library and drive objects.
- Whether the device is managed by an external media management system. Such devices require a library but no drive definition.

Disk Devices

Magnetic disk devices are the only devices in the random access category so they all share the same ADSM device type and device class: DISK. ADSM predefines the DISK device class.

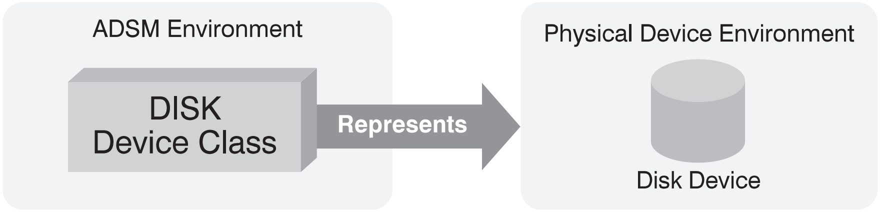


Figure 4. Magnetic Disk Devices Are Represented by Only a Device Class

Tape and Optical Devices

Figure 5 shows that a tape or optical device is represented by a library and a drive in addition to a device class.

Sequential devices for which an operator must perform volume mounts require a different ADSM library than devices that are associated with an automated library with robotics. ADSM provides a manual library type for stand-alone devices that are loaded by an operator and automated library types for devices loaded by a robot.

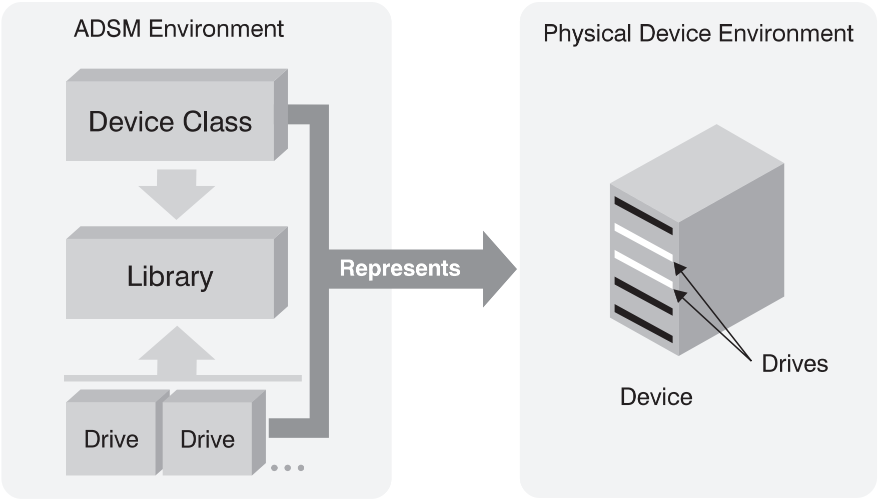


Figure 5. Removable Media Devices are Represented by a Library, Drive, and Device Class

Sequential devices that are managed by an external media management system require a library definition, but not a drive definition.

Files on Disk as Sequential Volumes

ADSM allows administrators to create volumes on server disk space that have the characteristics of sequential access volumes such as tape. ADSM supports these sequential volumes through the FILE device type. FILE is a special kind of sequential device type that, because it is on disk, does not require the administrator to define a library or drive object; only a device class is required.

FILE sequential volumes are often useful when transferring data for purposes such as electronic vaulting.

How ADSM Represents Storage Media

ADSM represents storage media with administrator-defined ADSM objects: storage pool volumes and storage pools. Figure 6 shows storage pool volumes grouped into a storage pool. Each storage pool represents only one type of media. For example, a storage pool for 8mm devices represents collections of only 8mm tapes.

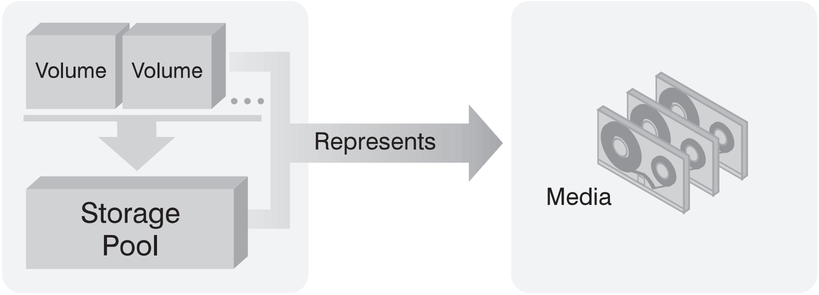


Figure 6. Relationships of Storage Pool Volumes, Storage Pools, and Media

What Are the ADSM Storage Objects?

The following ADSM storage objects are collections of information that the ADSM server uses to communicate with devices and to manage media:

- Device class
- Library
- Drive
- Storage pool
- Storage volume

Device Class

Each device is associated with an ADSM device class. A device class contains information about the device type and the way the device manages its media. See Chapter 7, "Defining Device Classes" on page 91 for more detailed information about device classes.

For devices that access data randomly, ADSM provides a predefined device class of DISK.

For devices that access data sequentially, the administrator must define the device class. (Devices that access data sequentially include FILE device classes, where data resides in files on the server's disk storage.) If the sequential device is a tape drive or optical disk, the device class is associated with a library. The library object is required for sequential devices because of the variations in media type (for example, 4mm, 8mm, and optical disk) and because of the need to manage multiple drives and automation.

Library

An ADSM library is an administrator-defined collection of one or more drives, and possibly robotic devices (depending on library type) sharing similar media mounting requirements. Each tape or optical disk device must be associated with an ADSM library.

An ADSM library can contain more than one physical device and can contain different types of devices. Use different libraries to identify devices that are mounted by different means (for example, an operator instead of robotics). You can define these types of libraries:

- MANUAL, for groups of devices that are loaded by an operator
- SCSI, for drives in a device that uses automation or robotics to load the drives
- 349X, for drives in an IBM 3494, an automated library
- EXTERNAL, for drives managed by an external media management program

See Chapter 6, “Defining Drives and Libraries” on page 83 for more information about ADSM libraries.

Drive

Each drive mechanism within a tape or optical disk device is represented by an ADSM drive. For devices with multiple drives, including automated libraries, each drive is separately defined to ADSM. Each drive is associated with an ADSM library. See Chapter 6, “Defining Drives and Libraries” on page 83 for more information about drives.

Storage Pools

A storage pool is a named collection of storage volumes that are associated with one device class. Each storage pool represents a collection of volumes sharing the same media type. For example, a storage pool associated with a device class for 8mm tape contains only 8mm tape volumes. You can control the characteristics of storage pools, such as whether scratch volumes are used, by specifying parameters. For details on the parameters, see Chapter 8, “Managing Storage Pools” on page 107.

ADSM supplies default disk storage pools named BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL. For more information, see “Using Random Access Volumes on Disk Devices” on page 32.

Storage Pool Volumes

An ADSM storage pool volume represents space on media that is available for storing ADSM client data. A storage pool volume is associated with a storage pool. For example, 8mm tapes and optical disks become storage pool volumes when they are assigned to an ADSM storage pool.

See Chapter 9, “Managing Storage Pool Volumes” on page 159 for more information about ADSM storage pool volumes.

What Does a Device Class Contain?

The contents of a device class are determined by whether the device accesses the data on its media randomly or sequentially.

Device Classes for Random Access Devices

Devices that access their media randomly share a common ADSM device type, and they do not require the administrator to define an ADSM library. ADSM provides a single, random-access device class, named DISK. You cannot define other random access device classes.

Random access device types store data in blocks of storage that can be scattered across the available space on a disk. As the server deletes data that has expired, the space occupied by that data can be reused.

Device Classes for Sequential Access Devices

Devices such as tape drives access their data sequentially. A device class for a sequential device contains a device type and media management information. For most sequential devices, the device class also specifies a library. Figure 7 shows the contents of a device class for a typical sequential access device.

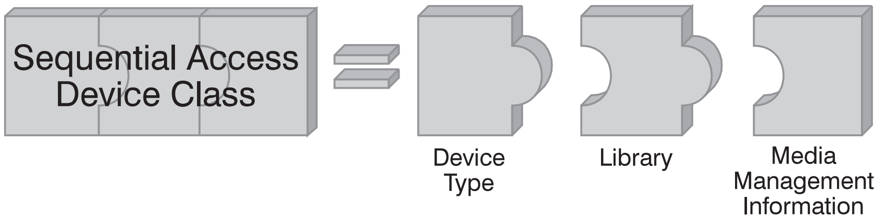


Figure 7. Contents of a Device Class for Sequential Access Devices

Sequential access device types begin to store data at the beginning of a volume, and append new data after existing data. As data is deleted, the space is not immediately reused. The server can reclaim space later by using the reclamation process (see “Space Reclamation for Sequential Access Storage Pools” on page 127 for details).

Tape devices, optical disk devices, and FILE type devices are members of the sequential access category of devices.

Device Type

Every sequential access device class requires one of the ADSM device types as part of its definition. A device type identifies a device as a member of a group of devices sharing similar media characteristics. ADSM provides device types for many devices, for example, DISK, 8MM, OPTICAL, and FILE device types. For example, 8mm tape devices require 8mm tapes; all 8mm tape devices share a device type of 8MM.

FILE is a special kind of ADSM sequential device type that allows the administrator to create sequential volumes on disk by creating files on the ADSM server that have the characteristics of a tape volume.

Library

For sequential access device types (excluding FILE), you must specify a library in the device class definition. The library you specify must be one that you have defined to ADSM, as discussed in “Library” on page 11.

Media Management Information

Every sequential access device class contains media management information, such as recording format and labeling prefixes. For more information about how ADSM helps to manage media, see “Using Disk for FILE Sequential Volumes” on page 33, Chapter 4, “Using Removable Media Devices with ADSM” on page 37, and Chapter 6, “Defining Drives and Libraries” on page 83.

Putting It All Together

Figure 8 on page 15 summarizes the relationships among the physical device environment, ADSM storage and policy objects, and ADSM clients. The numbers in the following list correspond to the numbers in the figure.

1 When clients are registered, they are associated with a policy domain. Within the policy domain are the other ADSM policy objects.

2 , **3**
When a file is backed up, archived, or migrated from a client, it is bound to a management class. A management class and the backup and archive copy groups within it specify where files are stored and how they are managed when they are backed up, archived, or migrated from a client (space-managed files).

4 , **5**
Storage pools are the destinations for backed up, archived, or space-managed files. Copy groups specify storage pools for backed up or archived files. Management classes specify storage pools for space-managed files.

Storage pools are mapped to device classes, which represent devices. The storage pool contains volumes as indicated by the device type associated with the device class. For example, a storage pool that is mapped to a device class with a device type of 8mm contains only 8mm tapes.

All devices require a device class that specifies at least a device type. Tape and optical devices also require a library and drive for management of media, including the mounting of that media.

6 Files that are initially stored on disk storage pools can migrate to tape or optical disk storage pools if the pools are set up in a storage hierarchy.

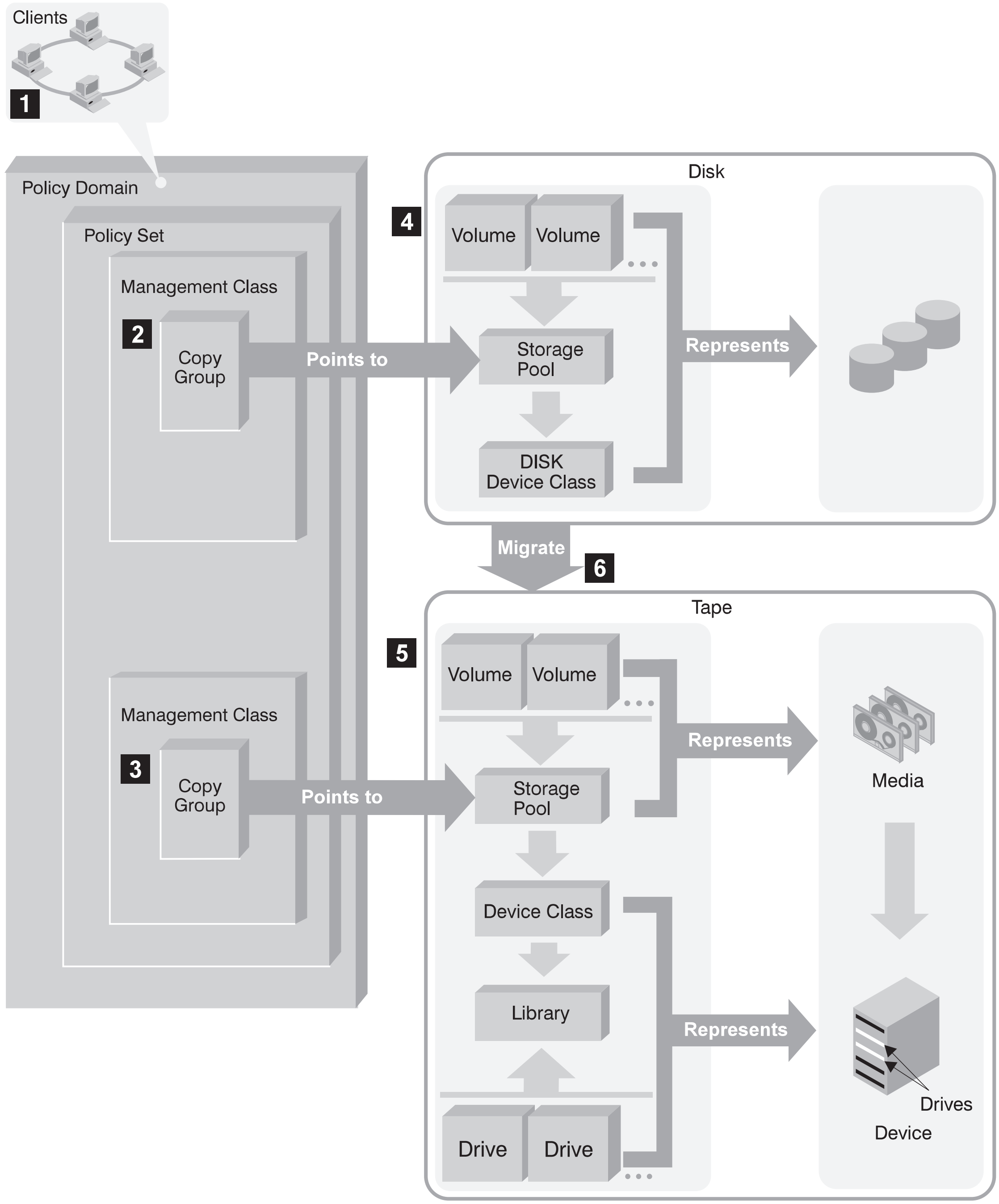


Figure 8. Putting It All Together

Planning to Configure the ADSM Storage Environment

Businesses often back up data to a variety of storage devices ranging from high-performance disk devices to slower and less expensive tape devices. Administrators must balance the data availability requirements of users with the costs of storage devices.

This section discusses how to evaluate your current environment to determine the device classes and storage pools for your ADSM storage environment.

Evaluating Your Storage Environment

Before configuring devices, evaluate the hardware available to ADSM.

1. Determine the storage devices that are available to ADSM. For example, determine how many tape drives you have that you will allow ADSM to use.

   ADSM expects to have exclusive use of the drives defined to it. If another application tries to use a drive defined to ADSM while ADSM is running, some ADSM server functions may fail.

2. Determine the ADSM device type and class for each of the available devices. Group together similar devices and identify their device classes. For example, create separate categories for 4mm and 8mm devices.

   Note: For sequential access devices, categorize the type of cartridge based on capacity. For example, standard length cartridge tapes and longer length cartridge tapes require different device classes.

3. Determine how the mounting of volumes is accomplished for the devices:

   - Devices that require operators to load volumes must be part of a MANUAL library in ADSM.
   - Devices that are automatically loaded must be part of a SCSI or 349X library in ADSM. Each automated library device is a separate ADSM library. The Exabyte EXB-210 is an example of a SCSI automated tape library.
   - Devices that are managed by an external media management system must be part of an EXTERNAL library in ADSM.

4. Determine the storage pools to set up, based on the devices you have and on user requirements. Gather users' requirements for data availability. Determine which data needs quick access and which does not.

5. Be prepared to label storage pool volumes. You may want to create a new labeling convention for ADSM storage pool volumes so that you can distinguish them from media used for other purposes.

Mapping Devices to Device Classes

As an example of mapping devices to device classes, assume the following ADSM storage environment:

- Internal disk drives
- An automated tape library with 8mm drives

- A manual DLT tape drive

You can map storage devices to device classes as shown in Table 1.

Table 1. Mapping Storage Devices to Device Classes

Device Class	Description
DISK	Storage volumes that reside on the internal disk drive ADSM provides one DISK device class that is already defined. You do not need and cannot define another device class for disk storage.
8MM_CLASS	Storage volumes that are 8mm tapes, used with the drives in the automated library
DLT_CLASS	Storage volumes that are DLT tapes, used on the DLT drive

You must define any device classes that you need for your removable media devices such as tape drives. See Chapter 7, “Defining Device Classes” on page 91 for information on defining device classes to support your physical storage environment.

Mapping Storage Pools to Device Classes and Devices

After you have categorized your storage devices, identify availability, space, and performance requirements for user data stored in ADSM storage. You can then create storage pools that are storage destinations for backed up, archived, or space-managed files to match those requirements.

For example, you determine that users in the business department have three requirements:

- Immediate access to certain backed up files, such as accounts receivable and payroll accounts.

  These files should be stored on disk. However, you need to ensure that data is moved from the disk to prevent it from becoming full. You can set up a storage hierarchy so that files can migrate automatically from disk to the automated tape library.

- Periodic access to some archived files, such as monthly sales and inventory reports.

  These files can be stored on 8mm tapes, using the automated library.

- Occasional access to backed up or archived files that are rarely modified, such as yearly revenue reports.

  These files can be stored using the DLT drive.

To match user requirements to storage devices, you define storage pools, device classes, and, for device types that require them, libraries and drives. To set up the storage hierarchy so that data migrates from the BACKUPPOOL to 8mm tapes, you specify BACKTAPE1 as the next storage pool for BACKUPPOOL. See Table 2 on page 18.

Table 2. Mapping Storage Pools to Device Classes, Libraries, and Drives

Storage Pool	Device Class	Library (Hardware)	Drives	Volume Type	Storage Destination
BACKUPPOOL	DISK	—	—	Storage volumes on the internal disk drive	For a backup copy group for files requiring immediate access
BACKTAPE1	8MM_CLASS	AUTOLIB_8MM (Exabyte EXB-210)	DRIVE01, DRIVE02	8mm tapes	For overflow from the BACKUPPOOL and for archived data that is periodically accessed
BACKTAPE2	DLT_CLASS	MANUAL_LIB (Manually mounted)	DRIVE03	DLT tapes	For backup copy groups for files that are occasionally accessed

Note: ADSM supplies default disk storage pools named BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL. For more information, see “Using Random Access Volumes on Disk Devices” on page 32.

Configuring Devices

Before a device can be used by ADSM, the device must be configured to the operating system as well as to ADSM. For information on these tasks, see the following:

- Chapter 3, “Using Magnetic Disk Devices with ADSM” on page 31
- Chapter 4, “Using Removable Media Devices with ADSM” on page 37

For devices that use removable media, you must choose a library type when you configure the devices. ADSM uses the library type to determine how volume mount operations are controlled on the drives in that library. The ADSM library types are:

MANUAL Volumes are mounted by an operator (a manual library)

SCSI Volumes are mounted automatically (by robotics, for example)

349X Volumes are mounted automatically within an IBM 3494 library

EXTERNAL Volumes are mounted under the control of an external media management system

MANUAL Libraries

In a *MANUAL* library, an operator mounts the volumes. Define a MANUAL library if you have one or more drives for which operators must mount volumes (drives that are not part of an automated library). Drives with different device types, such as DLT and 8MM, can be combined in a single MANUAL library.

When the ADSM server determines that a volume needs to be mounted in a drive that is part of a MANUAL library, the server issues mount request messages that prompt an operator to mount the volume. These messages are sent to the server console and to administrative clients that were started by using the special *mount mode* or *console mode* parameter.

For guidance on configuring a MANUAL library, see Chapter 4, “Using Removable Media Devices with ADSM” on page 37. For how to monitor mount messages for a MANUAL library, see “Mount Operations for Manual Libraries” on page 58.

SCSI Libraries

A *SCSI* library is a collection of drives for which volume mounts and demounts are handled automatically by a robot or other mechanism. This type applies to automated libraries other than the IBM 3494. The Exabyte EXB-210 is an example. When you define a SCSI library to the ADSM server, you must specify the library device name. To mount and dismount a volume in a drive that resides in the SCSI library, ADSM uses the library name.

IBM 3590 Drive with ACF: The IBM 3590 tape device with Automatic Cartridge Facility (ACF) must be defined as a separate SCSI library, with one drive.

For guidance on configuring a SCSI library, see Chapter 4, “Using Removable Media Devices with ADSM” on page 37. For an example of how to add volumes to a SCSI library, see “Prepare Volumes for Use by the Library” on page 55.

349X Libraries

A *349X* library is a collection of drives in an IBM 3494 Tape Library Dataserver. Volume mounts and demounts are handled automatically by the automation in the library.

When you define a 349X library to the ADSM server, you must specify the device name of one or more *library management control points* (LMCP). Each LMCP provides an independent interface to the robot mechanism within a given 349X library. To mount and dismount a volume in a drive that is in a 3494 library, ADSM uses an LMCP.

Note: For each 3494, you can define only one ADSM library.

For guidance on configuring a 349X library, see Chapter 4, “Using Removable Media Devices with ADSM” on page 37. For an example of how to add volumes to a 349X library, see “Prepare Volumes for Use by the Library” on page 55.

External Libraries

An *EXTERNAL* library is a collection of drives managed by a media management system that is not part of ADSM. ADSM provides an interface that allows external media management systems to operate in conjunction with the ADSM server. To use the interface for one or more devices, you must define a library with library type EXTERNAL.

For EXTERNAL libraries, ADSM uses the external media management system to perform the following functions:

- Volume mounts (specific and scratch)
- Volume dismounts
- Freeing of library volumes (return to scratch)

The external media manager selects the appropriate drive for media access operations. The drives in an EXTERNAL library are not defined to ADSM.

The EXTERNAL library type allows flexibility in grouping drives into libraries and storage pools. An EXTERNAL library may be one drive, a collection of drives, or even a part of an automated library.

For a definition of the interface that ADSM provides to the external media management system, see Appendix B, “External Media Management Interface Description” on page 461.

Automating Client Operations

You can automate operations such as backup for the ADSM clients. Figure 9 on page 21 shows the ADSM objects that may be involved in automated client operations. The key objects that interact are:

Include-exclude list (file for UNIX clients) on each ADSM client
Determines which files are backed up or space-managed, and determines management classes for files

Management class
Determines where client files are stored and how they are managed

Schedule
Determines when client operations such as backup occur

Association defined between client and schedule
Determines which schedules are run for a client

The client can specify a management class for a file or set of files, or can use the default management class for the policy domain. The client specifies a management class by using an INCLUDE option in the client’s include-exclude list or file. (See **A** in Figure 9 on page 21.)

The management class contains information that determines how ADSM handles files that clients backup, archive, or migrate. For example, the management class contains the backup copy group and the archive copy group. Each copy group points to a *destination*, a storage pool where files are stored when they are backed up or archived. (See **E** in Figure 9 on page 21.)

Clients are assigned to a policy domain when they are registered. Schedules that can automate client operations are also associated with a policy domain. (See **C** in Figure 9 on page 21.) To automate client operations, you define schedules for a domain. Then you define associations between schedules and clients in the same domain. (See **B** in Figure 9 on page 21.)

For a schedule to work on a particular client, the client machine must be turned on and must be running the client scheduler.

The scheduled client operations are called *events*, and information about events are stored in the ADSM database. (See **D** in Figure 9 on page 21.) For example, you can query the server to determine which scheduled events completed successfully and which failed.

For how to set up policy domains and management classes, see Chapter 10, “Managing Policies” on page 185. For more details on how to automate client operations, see “Automating Client Operations” on page 225.

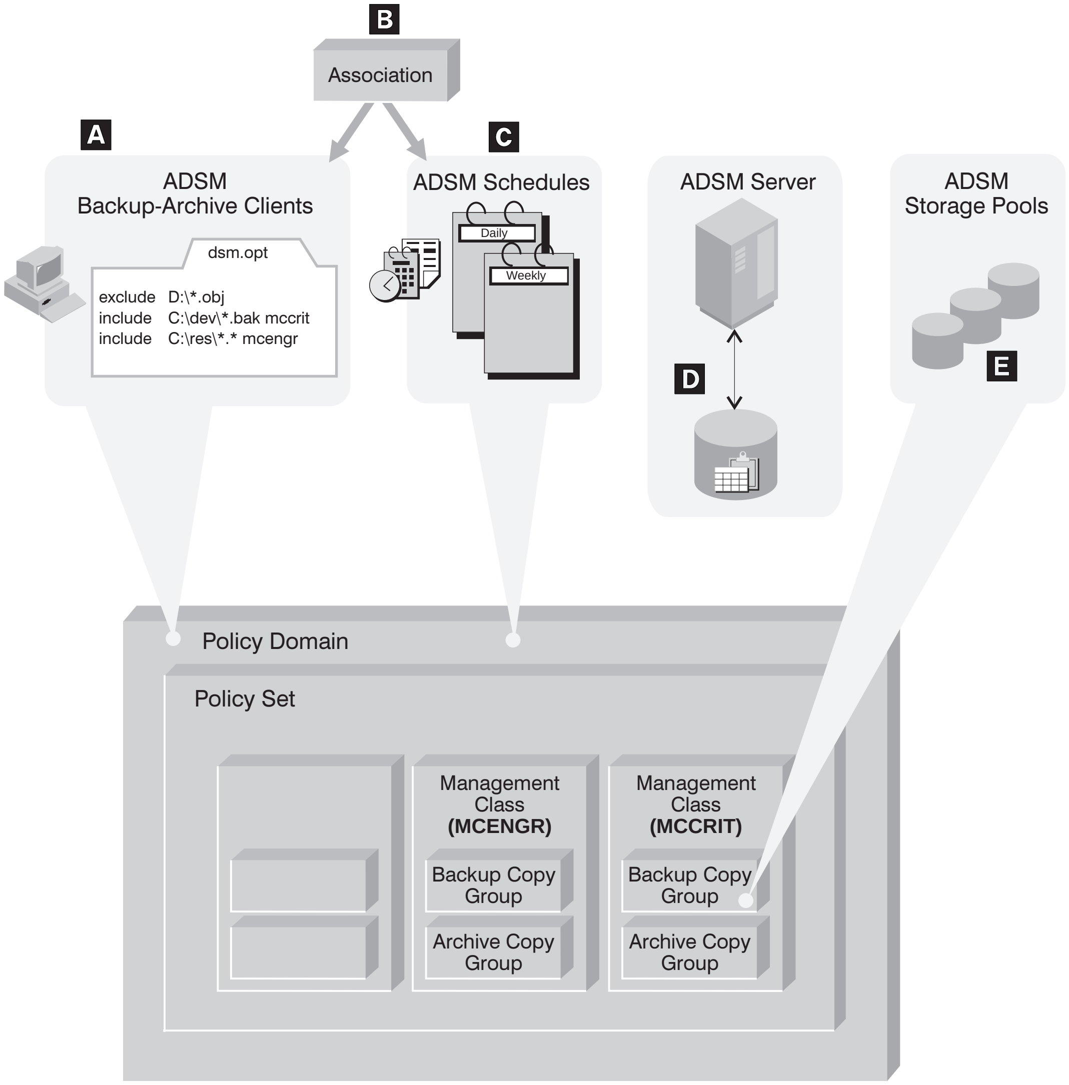


Figure 9. Automating Client Operations

Chapter 2. Administrator Tasks

This chapter provides a brief overview of the tasks that ADSM administrators can do. It also points to the sections in this publication that present the details of those tasks and the concepts you need to understand to complete them. The tasks are in the order in which they appear in the chapters of this book:

- Configuring and Managing Server Storage
  - Using magnetic disk devices with ADSM
  - Using removable media devices with ADSM
  - Managing removable media operations
  - Defining drives and libraries
  - Defining device classes
  - Managing storage pools
  - Managing storage volumes
- Policies
  - Managing ADSM policies
- Automating Operations
- Maintaining the Server
  - Managing server operations
  - Managing the database and recovery log
  - Managing licensing, privilege classes, and registration
  - Exporting and importing data
- Protecting and recovering your data

Interfaces to ADSM

There are three types of interfaces to ADSM:

- Graphical user interfaces (GUIs).

  For information about using the GUI, see the online information or refer to *ADSM Quick Start*.
- The command-line interface. For information about using the command-line interface of the administrative client, see *ADSM Administrator's Reference*. For information about using the command-line interface of the backup-archive client, see the ADSM user's guide for that client.
- The application programming interface. For more information, see *ADSM Using the Application Programming Interface*.

See Appendix C, "Interface Cross-Reference" on page 467 for a table that relates administrative commands with the administrative GUI.

© Copyright IBM Corp. 1995, 1997

Using Magnetic Disk Devices with ADSM

Magnetic disk devices can be used with ADSM for two purposes:

- Storage of the database and recovery log
- Storage of client data that is backed up, archived, or migrated from client nodes

ADSM can store data on magnetic disk in random access volumes by specifying a device type of DISK, or in sequential access volumes by specifying a device type of FILE.

For guidance setting up storage pools on disk devices, see Chapter 3, “Using Magnetic Disk Devices with ADSM” on page 31.

Using Removable Media Devices Devices with ADSM

Removable media devices can be used with ADSM for the following purposes:

- Storage of client data that is backed up, archived, or migrated from client nodes
- Storage of database backups
- Exporting data

For guidance and scenarios on configuring your removable media devices, see Chapter 4, “Using Removable Media Devices with ADSM” on page 37.

Managing Removable Media Operations

ADSM allows you to use and reuse removable media to store data. You must prepare removable media for initial use by ADSM. You also control how and when media are reused.

When the server requires that a volume be mounted, it generates a request. You need to monitor and respond to the requests.

For information about managing removable media operations, see Chapter 5, “Managing Removable Media Operations” on page 61.

Defining Drives and Libraries

To use removable media devices with ADSM, you must define libraries and drives.

For more information, see Chapter 4, “Using Removable Media Devices with ADSM” on page 37. For additional detailed information about these tasks, see Chapter 6, “Defining Drives and Libraries” on page 83.

Defining Device Classes

A device class represents a set of storage devices with similar availability, performance, and storage characteristics. You must define device classes for the types of drives available to an ADSM server. You specify a device class when you define a storage pool, which is a named collection of volumes for storing user data.

For more information about defining device classes, see Chapter 7, “Defining Device Classes” on page 91.

Managing Storage Pools

Backed up, archived, and space-managed files are stored in groups of volumes called storage pools. The data on these primary storage pools can be backed up to copy storage pools for disaster recovery purposes. Because each storage pool is assigned to a device class, you can logically group your storage devices to meet your storage management needs.

You can establish a hierarchy of storage pools. The hierarchy may be based on the speed or the cost of the devices associated with the pools. ADSM migrates client files through this hierarchy to ensure the most efficient use of a server’s storage devices.

When defining or modifying a storage pool, you can specify any or all of the following:

Cache When files are migrated from disk storage pools, duplicate copies of the files may remain in cache (disk storage) for faster retrieval and are deleted only when space is needed.

Collocation ADSM can keep each client’s files on a minimal number of volumes within a storage pool. Because client files are consolidated, restoring collocated files requires fewer media mounts.

Reclamation Files on sequential access volumes may expire, move, or be deleted. The reclamation process consolidates the active, unexpired data on many volumes onto fewer volumes. The original volumes can then be reused for new data.

For more information about storage pools and taking advantage of storage pool features, see Chapter 8, “Managing Storage Pools” on page 107.

Managing Storage Pool Volumes

You manage storage volumes by defining, updating, and deleting volumes, and by monitoring the use of server storage. Monitoring volumes can reveal inconsistencies that can be corrected between information in the database and client node files in storage pools. You can also move files within and across storage pools to optimize the use of server storage.

For more information about these tasks, see Chapter 9, “Managing Storage Pool Volumes” on page 159.

Managing Policies

From a client node, files can be backed up or archived to the server. This process ensures that current data can be restored or retrieved if it is accidentally deleted or corrupted on the workstations. Files from an HSM client can also be migrated from local file systems to ADSM server storage. Recall of migrated files is transparent and automatic when a client accesses a file, or the client can selectively recall files.

You define policies based on user requirements for backing up, archiving, or migrating data. You do this by defining policy objects, which identify backup, archive, and migration criteria, and by scheduling client operations.

For more information about establishing and managing policies for your organization, see Chapter 10, “Managing Policies” on page 185.

Automating Operations

You can define schedules for the automatic processing of most administrative commands and client operations such as backup and restore.

For more information about scheduling ADSM commands and operations, see Chapter 11, “Automating Operations” on page 223.

Managing Server Operations

You can manage server operations such as starting and stopping the server, maintaining and suspending client sessions with the server, and controlling server processes.

ADSM provides you with many sources of information about server and client status and activity, the state of the database, and resource usage. By monitoring this information, you can provide reliable services to users while making the best use of available resources.

For details about the day-to-day tasks involved in administering the server and about reports and information available to you, see Chapter 12, “Managing Server Operations” on page 245.

Managing the Database and Recovery Log

The ADSM database contains information about the client data in storage pools, registered client nodes, ADSM policies, and ADSM schedules. The server recovery log, which records changes made to the database, is used to restore the database to a consistent state.

You manage the database and recovery log space to tune database and recovery log performance.

For more information about the ADSM database and recovery log and about the tasks associated with administering them, see Chapter 13, “Managing the Database and Recovery Log” on page 261.

Managing Licensing, Privilege Classes, and Registration

You can monitor an installation’s compliance with the terms of its license agreement. ADSM lets you check license compliance and modify the terms.

An organization may name a single administrator or may distribute the workload among a number of administrators and grant them different levels of authority.

You register workstations as client nodes with the server. You can also provide client/server authentication by requiring the use of passwords to ensure that the client and the server are authorized to communicate with each other.

For more information about these tasks, see Chapter 14, “Managing Licensing, Privilege Classes, and Registration” on page 279.

Exporting and Importing Data

As your storage needs increase, you can move data from one server to another. You can *export* part or all of a server’s data to tape or a flat file so that you can then *import* the data to another server.

For more information about moving data between servers, see Chapter 15, “Exporting and Importing Data” on page 303.

Protecting and Recovering Your Data

ADSM provides a number of ways to protect and recover your data from media failure or from the loss of the ADSM database or storage pools due to a disaster. These recovery methods are based on the following preventive measures:

- Mirroring, by which the server maintains one or more copies of the database or recovery log, allowing the system to continue when one of the mirrored disks fails
- Periodic backup of the database
- Periodic backup of the storage pools
- Recovery of damaged files

For more information about protecting your data and for details about recovering from a disaster, see Chapter 16, “Protecting and Recovering Your Data” on page 327.

Using Disaster Recovery Manager

Disaster Recovery Manager (DRM) is an optional feature that assists an administrator with preparing a disaster recovery plan. The disaster recovery plan can be used to guide an administrator through disaster recovery as well as for audit purposes to certify the recoverability of the ADSM server.

DRM’s disaster recovery methods are based on the following measures:

- Enabling Disaster Recovery Manager
- Creating a backup copy of server primary storage pools and database
- Sending server backup volumes offsite
- Moving reclaimed or expired volumes back onsite
- Creating the ADSM server disaster recovery plan file
- Storing client machine information
- Defining and tracking client recovery media

Part 2. Configuring and Managing Server Storage

© Copyright IBM Corp. 1995, 1997

Chapter 3. Using Magnetic Disk Devices with ADSM

With ADSM, magnetic disk devices are used for these purposes:

- To store the database and the recovery log.

  For details, see Chapter 13, “Managing the Database and Recovery Log” on page 261.

- To store client data that has been backed up, archived, or migrated from client nodes. The client data is stored in storage pools.

  A summary of procedures for using disk storage for client data is in this chapter.

- To store backups of the ADSM database and to export and import ADSM data. See “Using Disk for FILE Sequential Volumes” on page 33.

The sections listed in the following table begin at the indicated pages.

Section	Page
Concepts:	
Using cache	34
Freeing space on disk	34
Scratch FILE volumes	34
FILE volumes used for database backups and export operations	35
Tasks:	
Using random access volumes on disk devices	32
Using disk for FILE sequential volumes	33

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator’s Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, “Interface Cross-Reference” on page 467 lists each command and shows if its function is also available on the administrative client GUI.

Setting Up Storage Pools on Disk Devices

ADSM stores data on magnetic disks in two ways:

- In random access volumes, as data is normally stored on disk. See “Using Random Access Volumes on Disk Devices” on page 32.
- In files on the disk. Each file is considered a sequential access volume. Within each file, data is stored sequentially, as it is on tape devices. See “Using Disk for FILE Sequential Volumes” on page 33.

© Copyright IBM Corp. 1995, 1997

Using Random Access Volumes on Disk Devices

For disk devices, ADSM provides a defined DISK device class that is used with all disk devices.

Note: For performance reasons, allocate storage pool volumes on disk drives that reside on the ADSM server machine, not on remotely mounted file systems.

1. Format a random access volume. For example, enter the following command on an operating system command line:

   ```
   > dsmfmt -m -data stgvol.002 21
   ```

   This command formats 21MB of space for storage pool volume stgvol.002.

   See Chapter 9, “Managing Storage Pool Volumes” on page 159 for details on using DSMFMT, the ADSM formatting utility for random access volumes.

2. Define a storage pool that is associated with the DISK device class, or use one of the default storage pools that ADSM provides (ARCHIVEPOOL, BACKUPPOOL, and SPACEMGPOOL).

   For example, enter the following command on an operating system command line:

   ```
   define stgpool engback1 disk maxsize=5M highmig=85 lowmig=40
   ```

   This command defines storage pool engback1.

   See “Example: Defining a Storage Pool Hierarchy” on page 139.

3. Define the DISK volumes formatted in step 1 to the storage pool.

   For example, enter the following command on an operating system command line:

   ```
   define volume engback1 vol1
   ```

   This command defines volume vol1 in storage pool engback1.

   See “Defining Storage Pool Volumes” on page 163.

4. Do one of the following:

   - Specify the new storage pool as the destination for client files that are backed up, archived, or migrated, by modifying existing policy or creating new policy. See Chapter 10, “Managing Policies” on page 185.

   - Place the new storage pool in the storage pool migration hierarchy by updating an already defined storage pool. See “Example: Updating a Storage Pool Hierarchy” on page 140.

Using Disk for FILE Sequential Volumes

Another way to use magnetic disk storage is to use files as volumes that store data sequentially (as on tape volumes). FILE sequential volumes are often useful when transferring data for purposes such as electronic vaulting.

Do the following:

1. Define a device class with device type FILE.

   For example, enter the following command on an operating system command line:

   ```
   define devclass filclas devtype=file mountlimit=2
   ```

   This command defines device class filclas for device type file.

   See “Defining and Updating FILE Device Classes” on page 101.

   To store ADSM database backups or exports on FILE volumes, this step is all you need to do to prepare the volumes. For more information, see “Defining Device Classes for Backups” on page 336 and “Planning for Sequential Media Used to Export Data” on page 305.

2. Define a storage pool that is associated with the new FILE device class.

   For example, enter the following command on an operating system command line:

   ```
   define stgpool engback1 file
   ```

   This command defines storage pool engback1 for device type file.

   See “Defining a Primary Storage Pool” on page 137.

   To allow ADSM to use scratch volumes for this storage pool, specify a value for the number of maximum scratch volumes when you define the storage pool. If you do not allow scratch volumes, you must define each volume to be used in this storage pool. See “Preparing Volumes for Sequential Access Storage Pools” on page 162.

3. Do one of the following:

   - Specify the new storage pool as the destination for client files that are backed up, archived, or migrated, by modifying existing policy or creating new policy. See Chapter 10, “Managing Policies” on page 185.

   - Place the new storage pool in the storage pool migration hierarchy by updating an already defined storage pool. See “Example: Updating a Storage Pool Hierarchy” on page 140.

Notes on Operations

This section contains information to be aware of when using magnetic disk devices for ADSM. The sections give pointers to additional information.

Using Cache

When you define a storage pool that uses disk random access volumes, you can choose to enable or disable cache. Using cache can improve the retrievability of files. When you use cache, a copy of the file remains on disk storage even after the file has been migrated to the next pool in the storage hierarchy, for example to tape. If the file needs to be restored or retrieved, the copy in cache can be used rather than the copy on tape, improving performance. However, using cache increases the space needed for the ADSM database. For more information, see “The Use of Cache on Disk Storage Pools” on page 120.

Freeing Space on Disk

As client files expire, the space they occupy is not freed for other uses until you run ADSM’s expiration processing.

Expiration processing deletes from the ADSM database information about any client files that are no longer valid according to the policies you have set. For example, suppose four backup versions of a file exist in ADSM server storage, and only three versions are allowed in the backup policy (the management class) for the file. Expiration processing deletes information about the oldest of the four versions of the file. The space that the file occupied in the storage pool becomes available for reuse.

You can run expiration processing by using one or both of the following methods:

- Use the ADSM command EXPIRE INVENTORY. See “Running Expiration Processing to Delete Expired Files” on page 213.
- Set the server option for the expiration interval, so that expiration processing runs periodically. You can set options by editing the dsmserv.opt file (see *ADSM Administrator’s Reference*).

Scratch FILE Volumes

You can specify a maximum number of scratch volumes for a storage pool that has a FILE device type. When ADSM needs a new volume, ADSM automatically creates a file that is a scratch volume, up to the number you specify. When scratch volumes used in storage pools become empty, the files are deleted.

FILE Volumes Used for Database Backups and Export Operations

When you back up the database or export server information, ADSM records information about the volumes used for these operations in the *volume history*. ADSM will not allow you to reuse these volumes until you delete the volume information from the volume history. To reuse volumes that have previously been used for database backup or export, use the DELETE VOLHISTORY command. For information about the volume history and volume history files, see “Saving the Volume History File” on page 340.

Chapter 4. Using Removable Media Devices with ADSM

ADSM can use removable media devices such as tape and optical drives for storing backed-up, archived, and space-managed client data, for storing backups of its database, and for exporting data. The devices must be configured for use by ADSM.

The sections listed in the following table begin at the indicated pages.

Section	Page
Concepts:	
Overview of configuring devices	37
Notes on configuring devices	56
Notes on operations	58
Tasks:	
Installing device drivers	40
Configuring a manual library	45
Configuring an automated library	51

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, "Interface Cross-Reference" on page 467 lists each command and shows if its function is also available on the administrative client GUI and the ADSM server utilities.

Configuring Devices—An Overview

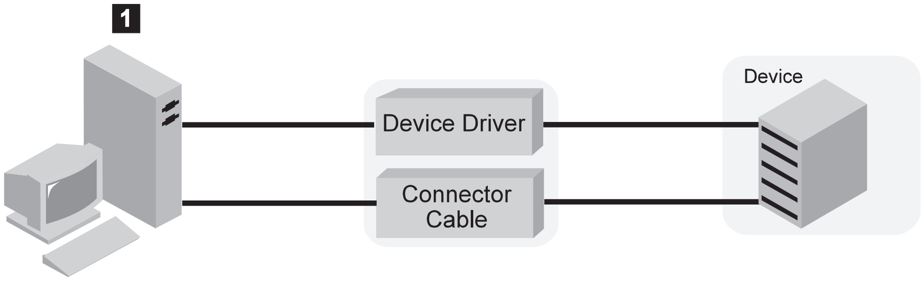
After tape or optical devices have been physically attached and defined to your server system, you must then configure them to ADSM so that ADSM can use and manage the devices and their media.

Sequential access devices typically require that the following steps be performed so that ADSM can use the devices.

1 Attach the device to the server system.

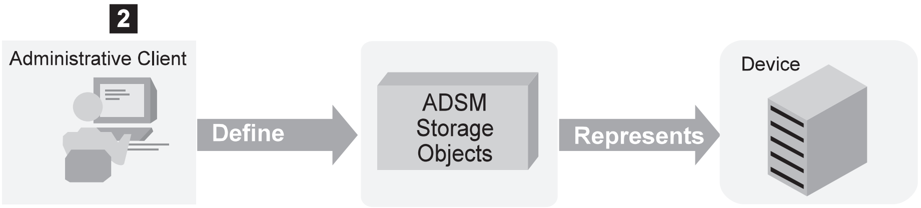
In addition to physically attaching the device to your system, you must ensure that the appropriate device driver is installed. For the IBM 3490E, 3570, and 3590 tape drives, and the IBM 3494 tape library, you must install the device drivers that came with the hardware. For other tape and autochanger devices and optical devices, you must ensure that you have installed the ADSM device drivers. See "Installing Device Drivers" on page 40.

© Copyright IBM Corp. 1995, 1997



2 Define the device to ADSM.

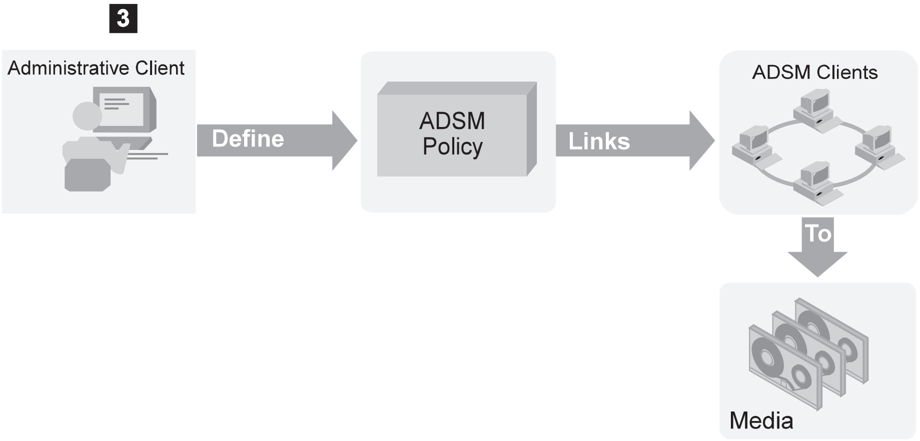
The administrator defines the storage objects that represent the physical device and media: library, drive, device class, storage pool, and storage volume. For an introduction to the ADSM storage objects, see “What Are the ADSM Storage Objects?” on page 10 and “Configuring Devices” on page 18.



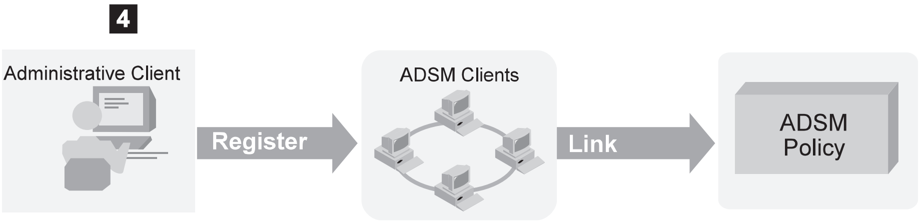
3 Define ADSM policy that links client data with media for the new device.

The administrator defines or updates the ADSM policy objects that will link clients to the pool of storage volumes and to the device. Do this by using the new storage pool as a destination for backed up, archived, or space-managed client data. For an introduction to the ADSM policy objects, see “How ADSM Stores Client Data” on page 5. For a description of the standard policy that is installed with ADSM, see “Using the Standard Storage Management Policies” on page 198.

An alternative is to simply place the new storage pool in the storage pool migration hierarchy by updating an already defined storage pool.



4 Register clients to the policy domain defined or updated in the preceding step. This step links clients and their data with storage volumes and devices.



5 Prepare volumes for use by the device. At a minimum, you must label volumes for the device. For automated libraries, you must also add the volumes to the device's volume inventory by checking in the volumes.

For an example of setting up a manual library, see "Example of a Manual Library: Setting Up Two 8mm Tape Drives" on page 45. For an example of setting up an automated library, see "Example of an Automated Library: Setting Up an 8mm Autochanger" on page 51.

Disadvantages of Libraries with One Drive: If you set up an ADSM library with only one drive, you cannot use ADSM's automatic volume reclamation and must reclaim volumes by a manual process. Reclamation allows reuse of volumes after data on the volumes expires, by moving any remaining unexpired data onto fewer volumes. If you will be using single-drive libraries, see "Space Reclamation for Sequential Access Storage Pools" on page 127 and "Reclamation in a Single-Drive Library" on page 132.

If you have a second drive of the same type that you could put in the same library, consider doing so to enable automatic reclamation.

Installing Device Drivers

For ADSM to use a device, you must install the appropriate device driver.

Tape drives and tape autochangers
You must ensure that you have installed the ADSM device drivers. When you install ADSM, you must choose whether to install the ADSM device drivers for tape and tape autochanger devices. See “Installing Device Drivers for Tape Drives and Libraries” on page 42 for how to install the device drivers.

IBM 3490E, 3494, 3570, and 3590 devices
Install the IBMtape device driver. For an IBM 3494, install the IBM Tape Library Driver for Sun Solaris, which includes the lmcp daemon. See “Installing Device Drivers for IBM 3490E, 3570, and 3590 Tape Drives” on page 41, “Installing Device Drivers for IBM 349X Libraries” on page 41, and *IBM SCSI Device Drivers: Installation and User’s Guide*.

Optical devices
Install the ADSM device drivers.

Before You Start: Names for Devices

To identify and work with removable media devices, ADSM needs the device’s name. For devices other than the IBM 3494, the name is the special file name. For a 3494, the device name is the symbolic name you entered in the configuration file for the library.

During configuration of devices other than the 3494, one or more device special file names are created. The following table shows the form of the names. In this table, *x* denotes a number (0–7).

Special File Name	Description
/dev/rmt/*x*mt	Used for tape drives that are supported by the ADSM device driver
/dev/rmt/*x*lb	Used for SCSI-attached libraries that are supported by the ADSM device driver
/dev/rmt/*x*op	Used for optical drives that are supported by the ADSM device driver
/dev/rmt/*x*st	Used for IBM 3490E, 3570, and 3590 tape drives, which are supported by the IBMtape device driver
/dev/rmt/*x*smc	Used for the medium changer (the robotics) of an IBM 3570 or an IBM 3590 B11 with Automatic Cartridge Facility (ACF) feature, which are supported by the IBMtape device driver

The special file is a symbolic link into the /devices tree of the file system. The symbolic link destination has the form:

```
/devices/.../mt@x,0:mt
```

where *x* is the SCSI ID of the device.

Installing Device Drivers for IBM 3490E, 3570, and 3590 Tape Drives

For IBM 3490E, 3570, and 3590 devices, see *IBM SCSI Device Drivers: Installation and User's Guide* for how to install the device drivers.

After completing the procedure in the manual, a set of special file names are available for each device. To see the special file names, use the command

```
ls -la /dev/rmt
```

The output shows that the special files are really symbolic links into the /devices tree of the file system.

- If you are installing the device driver for an IBM 3490E or an IBM 3590 without an ACF, you need to select one special file name for the drive. The name typically has the form:

  `/dev/rmt/`*x*`st`

  where *x* is a number. The name may have additional characters (n, b, c) at the end to indicate different operating characteristics.

  Enter the device name that you select in the Device Name field for the drive on the worksheet.

- If you are installing the device driver for an IBM 3590 Model B11 with an ACF, or an IBM 3570, you need to select two special file names for the device: one for the drive and one for the medium changer.

  The name for the drive typically has the form:

  `/dev/rmt/`*x*`st`

  where *x* is a number. The name may have additional characters (n, b, c) to indicate different operating characteristics. Enter the name that you select for the drive in the Device Name field for the drive. For example, to use the compression capability of the drive, enter:

  ```
  /dev/rmt/1stc
  ```

  The name for the medium changer has the form:

  `/dev/rmt/`*x*`smc`

  where *x* is a number. The `smc` stands for SCSI medium changer. Enter this name in the Device Name field for the library's medium changer (robotics). For example:

  ```
  /dev/rmt/0smc
  ```

Installing Device Drivers for IBM 349X Libraries

For an IBM 3494 Tape Library Dataserver, use the information in *IBM SCSI Device Drivers: Installation and User's Guide* to install the driver. The driver is the IBM Tape Library Driver for Sun Solaris, which consists of the *lmcpd* (a daemon) and other components.

To determine the name to use to define the library to ADSM, check what you entered in the /etc/ibmatl.conf file for the library's symbolic name. Use this information to complete the Device Name field on your worksheet.

For example, if the symbolic name for the library in the /etc/ibmatl.conf file is `3494a`, enter that name on the worksheet in the Device Name field for the library.

The drives that are in the library are set up separately. See “Installing Device Drivers for IBM 3490E, 3570, and 3590 Tape Drives” on page 41.

Installing Device Drivers for Tape Drives and Libraries

To use the ADSM device drivers, you must first edit configuration files that reside in the /usr/kernel/drv directory, and then issue the add_drv (add drive) command. The devices should already be connected to the system and powered on.

If you are configuring an optical library, you must set up the device drivers for both the drives in the library and the library itself (that is, the robotics).

Installing the ADSM Device Driver for Tape Drives

To install the ADSM device driver for tape drives, perform the following steps:

1. Verify that the device is connected to your system, and is powered on and active.

2. Edit the /usr/kernel/drv/mt.conf file. Add one stanza (as shown in the example at the top of the file) for each ADSM tape drive in the system that you want the ADSM server to use.

3. Log in as root user.

4. Add the drive by entering the following command:

```
> /usr/sbin/add_drv -m '* 0666 bin bin' /usr/kernel/drv/mt
```

5. Determine the special file names for the tape drives.

   The device special files for the tape drives appear in the /dev/rmt directory. Their names have the form /dev/rmt/*x*mt, where *x* is a number. To determine the relationship between a physical device and the special file that was created for it, issue the following command:

```
> ls -l /dev/rmt/*mt
```

   The output shows that the special file is really a symbolic link into the /devices tree of the file system. The symbolic link destination has the form /devices/.../mt@*x*,0:mt, where *x* is the SCSI ID of the device. This information lets you know which of the /dev/rmt/*x*mt special file names to provide to the server when you issue a DEFINE DRIVE command.

6 Record the SCSI IDs and special file names on the worksheets. See Appendix A, "Supported Devices and Device Configuration Worksheets" on page 423.

Installing the ADSM Device Driver for Tape Libraries

To install the ADSM device driver for tape libraries, perform the following steps:

1 Edit the /usr/kernel/drv/lb.conf file. Add one stanza (as shown in the example at the top of the file) for each ADSM autochanger device in the system that you want the ADSM server to use.

2 Log in as root user.

3 Add the library by entering the following command:

```
> /usr/sbin/add_drv -m '* 0666 bin bin' /usr/kernel/drv/lb
```

4 Determine the special file names for the tape libraries.

The device special files for libraries appear in the /dev/rmt directory. Their names have the form /dev/rmt/*x*lb, where *x* is a number. To determine the relationship between a physical device and the special file that was created for it, issue the following command:

```
> ls -l /dev/rmt/*lb
```

The output shows that the special file is really a symbolic link into the /devices tree of the file system. The symbolic link destination has the form /devices/.../lb@*x,y*:lb, where *x* is the SCSI ID of the device and where *y* is the LUN (logical unit number) of the device. This information lets you know which of the /dev/rmt/*x*lb special file names to provide to the server when you issue a DEFINE LIBRARY command.

5 Record the SCSI IDs and special file names on the worksheets. See Appendix A, "Supported Devices and Device Configuration Worksheets" on page 423.

Installing Device Drivers for Optical Devices

To use the ADSM device drivers, you must first edit configuration files that reside in the /usr/kernel/drv directory, and then issue the add_drv (add drive) command. The devices should already be connected to the system and powered on.

If you are configuring an optical library, you must set up the device drivers for both the drives in the library and the library itself (that is, the robotics) for ADSM to use.

Installing the ADSM Device Driver for Optical Drives

To install the ADSM device driver for optical drives, perform the following steps:

1. Verify that the device is connected to your system, and is powered on and active.

2. Edit the /usr/kernel/drv/op.conf file. Add one stanza (as shown in the example at the top of the file) for each ADSM tape drive in the system that you want the ADSM server to use.

3. Log in as root user.

4. Add the drive by entering the following command:

```
> /usr/sbin/add_drv -m '* 0666 bin bin' /usr/kernel/drv/op
```

5. Determine the special file names for the drives.

   The device special files for the optical drives appear in the /dev/rmt directory. Their names have the form /dev/rmt/*x*op, where *x* is a number. To determine the relationship between a physical device and the special file that was created for it, issue the following command:

```
> ls -l /dev/rmt/*op
```

   The output shows that the special file is really a symbolic link into the /devices tree of the file system. The symbolic link destination has the form /devices/.../op@*x*,0:op, where *x* is the SCSI ID of the device. This information lets you know which of the /dev/rmt/*x*op special file names to provide to the server when you issue a DEFINE DRIVE command.

6. Record the SCSI IDs and special file names on the worksheets. See Appendix A, “Supported Devices and Device Configuration Worksheets” on page 423.

Installing the ADSM Device Driver for Optical Libraries

To install the ADSM device driver for an optical library, perform the following steps:

1. Edit the /usr/kernel/drv/lb.conf file. Add one stanza (as shown in the example at the top of the file) for each ADSM autochanger device in the system that you want the ADSM server to use.

2. Log in as root user.

3. Add the library by entering the following command:

```
> /usr/sbin/add_drv -m '* 0666 bin bin' /usr/kernel/drv/lb
```

4 Determine the special file names for the optical libraries.

The device special files for libraries appear in the /dev/rmt directory. Their names have the form /dev/rmt/*x*lb, where *x* is a number. To determine the relationship between a physical device and the special file that was created for it, issue the following command:

```
> ls -l /dev/rmt/*lb
```

The output shows that the special file is really a symbolic link into the /devices tree of the file system. The symbolic link destination has the form /devices/.../lb@*x*,*y*:lb, where *x* is the SCSI ID of the device and where *y* is the LUN (logical unit number) of the device. This information lets you know which of the /dev/rmt/*x*lb special file names to provide to the server when you issue a DEFINE LIBRARY command.

5 Record the SCSI IDs and special file names on the worksheets. See Appendix A, “Supported Devices and Device Configuration Worksheets” on page 423.

Example of a Manual Library: Setting Up Two 8mm Tape Drives

In the following example, two 8mm drives are attached to the server system. The example takes you through the steps necessary to get ADSM to use the devices for storing client data.

Because an operator must mount tapes for these drives, you must define them as part of a manual library to ADSM. You can use this example as a guide when configuring other manual tape and optical devices. This example presents the procedure with a minimum of customization. If you want to do more, see the references in the steps for more details.

Attach the Device to the Server System

1 Install the SCSI adapter card in your system, if not already installed.

2 Determine the SCSI IDs available on the SCSI adapter card to which you are attaching the device. Find one unused SCSI ID for each drive.

3 Follow the manufacturer’s instructions to set the SCSI ID for the device to the unused SCSI ID that you found. Usually this means setting switches on the back of the device.

Attention: Each device connected in a chain to a single SCSI bus must be set to a unique SCSI ID. If each device does not have a unique SCSI ID, you may have serious system problems.

4 Follow the manufacturer's instructions to attach the device to your server system hardware.

Attention:

a. Power off your system before attaching a device to prevent damage to the hardware.

b. You must attach a terminator to the last device in the chain of devices connected on one SCSI adapter card. Detailed instructions should be in the documentation that came with your hardware.

5 Install the appropriate device driver. See "Installing Device Drivers" on page 40.

6 Determine the name for the device, which is needed to define the device to ADSM.

For a tape drive, the device name is a special file name with the form /dev/rmt/*x*mt, where *x* is the number assigned when the drive was set up with its device driver.

Define the Device to ADSM

1 Define a manual library for ADSM by entering the following command on the command line of an ADSM administrative client. The name of the library is MANUAL8MM. The library type is *manual* because an operator must mount the tapes.

```
define library manual8mm libtype=manual
```

2 Define the drives that belong to this manual library.

```
define drive manual8mm drive01 device=/dev/rmt/1mt

define drive manual8mm drive02 device=/dev/rmt/2mt
```

Both drives belong to the MANUAL8MM library. In this example, the drive known to the device driver by the special file name /dev/rmt/1mt is given the ADSM name DRIVE01. The device /dev/rmt/2mt is given the ADSM name DRIVE02. You might prefer to have the device driver name and the ADSM name match. See "Installing Device Drivers for Tape Drives and Libraries" on page 42 for determining the device driver's name for the device.

See "Defining Drives" on page 88.

3 Classify drives according to type and format by defining ADSM device classes. For example, if you want to use the 8500 recording format for the two drives in the MANUAL8MM library, use the following command to define a device class named TAPE8MM_CLASS:

```
define devclass tape8mm_class devtype=8mm
format=8500 library=manual8mm mountlimit=2
```

Key choice: Mount limit (number of drives available in this device class) has a default value of 1. The mount limit should be equal to the number of drives of the same type in that library.

A closer look: You can group several different types of manually mounted drives in one manual library. For example, you can group 8mm drives and 4mm drives in the same ADSM library. You define a different device class (associated with the same library) for each device type in the manual library. ADSM selects the type of device in the library based on the device class specified for the operation it is performing.

See “Defining and Updating Device Classes for Tape Devices” on page 93.

4 To check what you have defined, enter the following commands:

```
query library

query drive

query devclass
```

See “Requesting Information about Libraries” on page 86, “Requesting Information about Drives” on page 88, and “Requesting Information about a Device Class” on page 102.

5 Create the storage pool to use the devices in the device class you just defined. For example, define a storage pool named TAPE8MM_POOL associated with the device class TAPE8MM_CLASS:

```
define stgpool tape8mm_pool tape8mm_class maxscratch=20
```

Key choices:

a. Scratch volumes are labeled, empty volumes that are available for use. If you allow scratch volumes for the storage pool by specifying a value for the maximum number of scratch volumes, ADSM can use any scratch volumes (up to the number of scratch volumes available) without further action on your part. If you do not allow scratch volumes, you must perform the extra step of explicitly defining each volume to be used in the storage pool.

b. Collocation is turned off by default. Collocation is a process by which the server attempts to keep all files belonging to a client node or client file space on a minimal number of volumes. Once clients begin storing data in a storage pool with collocation off, you cannot easily change the data in the storage pool so that it is collocated. To understand the advantages and disadvantages of collocation, see "Collocation on Sequential Access Storage Pools" on page 121 and "How Collocation Affects Reclamation" on page 131.

See "Defining a Primary Storage Pool" on page 137.

Update ADSM Policy

You can do one of the following:

- Have clients back up data directly to tape.
- Have clients back up data to disk storage. Then let ADSM migrate the data to tape when the amount of disk storage used reaches the migration threshold.

The following steps assume that you are modifying the standard, IBM-supplied policy objects, named STANDARD, to allow clients to back up data directly to tape. However, if you want some clients to back up directly to tape and some to disk, keep the standard policy as is. For the clients that need to back up directly to tape, define new policy (policy domain, management class, copy groups) and assign these clients to the new policy domain. For details on the standard policy, see "Using the Standard Storage Management Policies" on page 198. For how to define new policy, see "Creating Your Own Storage Management Policies" on page 199.

Clients Back Up Directly to Tape

You can choose to have clients back up directly to the new tape storage pool that you defined.

Key choice: If you back up directly to tape, the number of clients that can back up data at the same time is equal to the number of drives available to the storage pool (through the mount limit of the device class). If you have only one drive, only one client at a time can back up data.

Performance of tape drives is often lower when backing up directly to tape than when backing up to disk and then migrating to tape. Backing up data directly to tape usually means more starting and stopping of the tape drive. Backing up to disk then migrating to tape usually means the tape drive moves more continuously, meaning better performance.

1 Update the backup copy group so that the destination for backups is the new tape storage pool, TAPE8MM_POOL. For example:

```
update copygroup standard standard standard
type=backup destination=tape8mm_pool
```

Note: You may want clients in the STANDARD policy domain to be able to *choose* whether to back up directly to disk or to tape. If so, instead of updating the copy group in the STANDARD management class, you can define a new management class and a new copy group in the STANDARD domain. See “Defining and Updating a Backup Copy Group” on page 205.

2 Activate this modified policy:

```
activate policyset standard standard
```

See “Activating Policy Sets” on page 212.

Clients Back Up to Disk Then Data Migrates

You can have clients back up data to disk storage. Then let ADSM migrate the data to the new tape storage pool when the amount of disk storage used reaches the migration threshold. For example, you can have data migrate from the default disk storage pool, BACKUPPOOL, to the new storage pool, TAPE8MM_POOL, by using the following command:

```
update stgpool backuppool nextstgpool=tape8mm_pool
```

If you have not changed the defaults for BACKUPPOOL, ADSM will migrate data from this disk pool to the TAPE8MM_POOL when the disk pool is 90% full. See “Defining or Updating Storage Pools” on page 137.

Register Clients to the Policy Domain

If you updated the default STANDARD policy to use the new storage pool as a destination for backups from clients, the clients must be registered to that policy domain. To register a client named ASTRO to the STANDARD policy domain and assign the client the password CADET, enter this command:

```
register node astro cadet
```

You do not need to specify a policy domain because the STANDARD policy domain is the default.

For information on options when registering clients, see “Administrator Registration of Client Nodes” on page 294.

Prepare Volumes for Use by the Library

Ensure that volumes are available to ADSM in the library.

1. You must label volumes that do not already have a standard label. Use the utility DSMLABEL from an operating system command line. For example, to use one of the 8mm drives, enter this command:

```
> dsmlabel -drive=/dev/rmt/2mt
```

   For this command, the drive name is the name by which the device driver knows the drive, not the ADSM drive object name.

   ADSM prompts you to put a new volume into the drive and enter the volume name (1–6 characters). Insert the volume, type the name, and press Enter. When the first volume is labeled, you can continue to label more volumes. For more information, see “Labeling Sequential Access Volumes” on page 68 and *ADSM Administrator’s Reference*.

   Tips: The drives that you use for labeling volumes must not be in use by the server for any other purpose. You can do one of the following:

   - Halt the server while you label volumes.
   - Prevent the server from using the drives you want to use for labeling volumes. One way to do this is to temporarily delete the ADSM drive definitions and lower the mount limit for the associated device class.

   Keep enough labeled volumes on hand so that you do not run out during an operation such as client backup.

2. What you need to do next depends on whether you are using scratch volumes or private volumes:

   - If you are using only scratch volumes, you have nothing more to do except ensure that there are enough scratch volumes available. For example, you may need to label more volumes. As volumes are used, you may also need to increase the number of scratch volumes allowed in the storage pool that you defined for this library.
   - If you want to use private volumes in addition to or instead of scratch volumes in the library, you must define volumes to the storage pool you defined. The volumes you define must have been already labeled. For information on defining volumes, see “Defining Storage Pool Volumes” on page 163.

Example of an Automated Library: Setting Up an 8mm Autochanger

For the following example, an Exabyte EXB-210 library containing two drives is attached to the server system. The example takes you through the steps necessary to get ADSM to use the devices in the library for storing client data.

You can use this example as a guide when configuring other automated tape devices. This example presents the procedure with a minimum of customization. If you want to do more, see the references in the steps.

Attach the Device to the Server System

1 Install the SCSI adapter card in your system, if not already installed.

2 Determine the SCSI IDs available on the SCSI adapter card to which you are attaching the device. Find one unused SCSI ID for each drive, and one for the library or autochanger controller.

Note: In some automated libraries, the drives and the autochanger share a single SCSI ID, but have different LUNs. For these libraries, only a single SCSI ID is required. Check the documentation for your device.

3 Follow the manufacturer's instructions to set the SCSI ID for the drives and library controller to the unused SCSI IDs that you found. Usually this means setting switches on the back of the device.

Attention: Each device connected in a chain to a single SCSI bus must be set to a unique SCSI ID. If each device does not have a unique SCSI ID, you may have serious system problems.

4 Follow the manufacturer's instructions to attach the device to your server system hardware.

Attention:

a. Power off your system before attaching a device to prevent damage to the hardware.

b. You must attach a terminator to the last device in the chain of devices connected on one SCSI adapter card. Detailed instructions should be in the documentation that came with your hardware.

5 Install the appropriate device drivers. See "Installing Device Drivers" on page 40.

6 Find the device worksheet that applies to your device. See Appendix A, "Supported Devices and Device Configuration Worksheets" on page 423.

7 Determine the name for the device, which is needed to define the device to ADSM, and record the name on the device worksheet.

The device name for each drive has the form /dev/rmt/*x*mt. The device name for the library controller has the form /dev/rmt/*x*lb. In both cases, *x* is the number assigned when the drive and library was set up with its device driver.

Keep the Worksheets: The information you record on the worksheets can help you when you need to perform operations such as adding volumes to an autochanger. Keep them for future reference.

Define the Device to ADSM

1 Define the library to ADSM. For example, for an Exabyte EXB-210 library, name the library AUTO8MMLIB. The library type is *SCSI* because the library is a SCSI-attached automated library, and is not an IBM 3494. Enter the following command on the command line of an ADSM administrative client:

```
define library auto8mmlib libtype=scsi device=/dev/rmt/3lb
```

For automated libraries, the library type is either SCSI or 349X. The DEVICE parameter gives the device driver's name for the library, the special file name. See “Installing Device Drivers for Tape Drives and Libraries” on page 42 for determining the device driver's name for the device.

See “Defining Libraries” on page 85 and “SCSI Libraries” on page 19.

2 Decide whether all drives in that library will be used by ADSM. Define the drives that ADSM will use. For example:

```
define drive auto8mmlib 4mt device=/dev/rmt/4mt element=82

define drive auto8mmlib 5mt device=/dev/rmt/5mt element=83
```

Both drives belong to the AUTO8MMLIB library. The DEVICE parameter gives the device driver's name for the drive. In this example, each drive is given an ADSM name that matches its device driver name. See “Installing Device Drivers for Tape Drives and Libraries” on page 42 for determining the device driver's name for the drives.

Element address: The element address is a number that indicates the physical location of a drive within an automated library. ADSM needs the element address to connect the physical location of the drive to the drive's SCSI address. When you define a drive, the element address is required if there is more than one drive in an automated library. The element numbers are taken from the device worksheet filled out in step 7 on page 51.

See “Defining Drives” on page 88.

3 Classify drives according to type and recording format by defining ADSM device classes. For example, if you want to use the 8500 recording format for the two drives in the AUTO8MMLIB library, use the following command to define a device class named AUTO8MM_CLASS:

```
define devclass auto8mm_class devtype=8mm format=8500
library=auto8mmlib mountlimit=2
```

Key choice: Mount limit (number of drives available in this device class) has a default value of 1. The mount limit should be equal to the number of drives of the same type in that library.

See “Defining and Updating Device Classes for Tape Devices” on page 93.

4 To check what you have defined, enter the following commands:

```
query library

query drive

query devclass
```

See “Requesting Information about Libraries” on page 86, “Requesting Information about Drives” on page 88, and “Requesting Information about a Device Class” on page 102.

5 Create the storage pool to use the devices in the device class you just defined. For example, define a storage pool named AUTO8MM_POOL associated with the device class AUTO8MM_CLASS:

```
define stgpool auto8mm_pool auto8mm_class maxscratch=20
```

Key choices:

a. Scratch volumes are labeled, empty volumes that are available for use. If you allow scratch volumes for the storage pool by specifying a value for the maximum number of scratch volumes, ADSM can choose from the scratch volumes available in the library, without further action on your part. If you do not allow scratch volumes, you must perform the extra step of explicitly defining each volume to be used in the storage pool.

b. Collocation is turned off by default. Collocation is a process by which the server attempts to keep all files belonging to a client node or client file space on a minimal number of volumes. Once clients begin storing data in a storage pool with collocation off, you cannot easily change the data in the storage pool so that it is collocated. To understand the advantages and disadvantages of collocation, see “Collocation on Sequential Access Storage Pools” on page 121 and “How Collocation Affects Reclamation” on page 131.

See “Defining a Primary Storage Pool” on page 137.

Update ADSM Policy

You can do one of the following:

- Have clients back up data directly to tape.
- Have clients back up data to disk storage. Then let ADSM migrate the data to tape.

The following steps assume that you are modifying the standard, IBM-supplied policy objects, named STANDARD, to allow clients to back up data directly to tape. However, if you want some clients to back up directly to tape and some to disk, keep the standard policy as is. For the clients that need to back up directly to tape, define new policy (policy domain, management class, copy groups) and assign these clients to the new policy domain. For details on the standard policy, see “Using the Standard Storage Management Policies” on page 198. For how to define new policy, see “Creating Your Own Storage Management Policies” on page 199.

Clients Back Up Directly to Tape

You can choose to have clients back up directly to the new tape storage pool that you defined.

Key choice: If you back up directly to tape, the number of clients that can back up data at the same time is equal to the number of drives available to the storage pool (through the mount limit of the device class). If you have only one drive, only one client at a time can back up data.

Performance of tape drives is often lower when backing up directly to tape than when backing up to disk and then migrating to tape. Backing up data directly to tape usually means more starting and stopping of the tape drive. Backing up to disk then migrating to tape usually means the tape drive moves more continuously, meaning better performance.

1 Update the backup copy group so that the destination for backups is the new tape storage pool, AUTO8MM_POOL. For example:

```
update copygroup standard standard standard
type=backup destination=auto8mm_pool
```

Note: You may want clients in the STANDARD policy domain to be able to *choose* whether to back up directly to disk or to tape. If so, instead of updating the copy group in the STANDARD management class, you can define a new management class and a new copy group in the STANDARD domain. See “Defining and Updating a Backup Copy Group” on page 205.

2 Activate this modified policy:

```
activate policyset standard standard
```

See “Activating Policy Sets” on page 212.

Clients Back Up to Disk Then Data Migrates

You can have clients back up data to disk storage. Then let ADSM migrate the data to the new tape storage pool when the amount of disk storage used reaches the migration threshold. For example, you can have data migrate from the default disk storage pool, BACKUPPOOL, to the new storage pool, AUTO8MM_POOL, by using the following command:

```
update stgpool backuppool nextstgpool=auto8mm_pool
```

If you have not changed the defaults for BACKUPPOOL, ADSM will migrate data from this disk pool to the AUTO8MM_POOL when the disk pool is 90% full. See “Defining or Updating Storage Pools” on page 137.

Register Clients to the Policy Domain

If you updated the default STANDARD policy to use the new storage pool as a destination for backups from clients, the clients must be registered to that policy domain. To register a client named ASTRO to the STANDARD policy domain and assign the client the password CADET, enter this command:

```
register node astro cadet
```

You do not need to specify a policy domain because the STANDARD policy domain is the default.

For information on options when registering clients, see “Administrator Registration of Client Nodes” on page 294.

Prepare Volumes for Use by the Library

Ensure that enough volumes are available to ADSM in the library.

1 You must label volumes that do not already have a standard label. Use the DSMLABEL command on an operating system command line. For example, to use one of the drives in the 8mm library and search for all usable volumes in the library, enter this command:

```
> dsmlabel -drive=/dev/rmt/4mt,82 -library=/dev/rmt/3lb
  -search -keep
```

For this command, the drive and library names are the names by which the device driver knows the drive and library, not the ADSM object names.

The program searches for volumes in the library, puts each volume into the drive, and prompts you to enter the name of the volume (1–6 characters). Type each name, and press Enter. For more information, see “Labeling Sequential Access Volumes” on page 68 and *ADSM Administrator’s Reference*.

Tips: The drives that you use for labeling volumes must not be in use by the server for any other purpose. You can do one of the following:

- Halt the server while you label volumes.
- Prevent the server from using the drives you want to use for labeling volumes. One way to do this is to temporarily delete the ADSM drive definitions and lower the mount limit for the associated device class.

Keep enough labeled volumes on hand so that you do not run out during an operation such as client backup.

2 Check in the volumes to ADSM. You must use the CHECKIN LIBVOLUME command on the command line of an ADSM administrative client. For example, to check in all volumes in the library that ADSM does not yet know about, enter the following command:

```
checkin libvolume auto8mmlib status=scratch search=yes
```

ADSM mounts each volume and checks the label recorded on the media. Each volume found is checked in to the library as a scratch volume, with the volume name being taken from the recorded label.

3 What you need to do next depends on whether you are using scratch volumes or private volumes:

- If you are using only scratch volumes, you have nothing more to do except ensure that there are enough scratch volumes available. For example, you may need to label more volumes. As volumes are used, you may also need to increase the number of scratch volumes allowed in the storage pool that you defined for this library.
- If you want to use private volumes in addition to or instead of scratch volumes in the library, you must define volumes to the storage pool you defined. The volumes you define must have been already labeled and checked in. See “Defining Storage Pool Volumes” on page 163.

Notes on Configuring Devices

The following sections present choices and procedures you need to be aware of when configuring devices for ADSM. The sections give pointers to additional information where applicable.

Troubleshooting Problems with Devices

You cannot share tape drives defined to ADSM with other applications. Errors and operation failures can occur when other applications attempt to use devices defined to ADSM. If you want to temporarily use a drive with another application, you can delete the drive definition from ADSM and lower the mount limit for the corresponding device class. Define the drive to ADSM again when the drive is available for ADSM use.

Also, you cannot use a device for labeling volumes with the ADSM DSMLABEL command at the same time that the server uses the device for other processes. For example, if the server is using a device to back up client data, you cannot use that device for labeling volumes.

If you have not already done so, register the license for the device support module that you purchased. For more information on licensing, see “Managing ADSM Licenses” on page 279.

Setting the Library Mode

For ADSM to access a SCSI library, the device must be set for the appropriate mode. The mode that ADSM requires is usually called *random* mode; however, terminology may vary from one device to another. Two examples follow:

- Some libraries have front panel menus and displays that can be used for explicit operator requests. However, if the device is set to respond to such requests, it typically will not respond to requests made by ADSM.
- Some libraries can be placed in *sequential* mode, in which volumes are automatically mounted in drives by using a sequential approach. This mode conflicts with how ADSM accesses the device.

Refer to the documentation for your device to determine how to set it to a mode appropriate for ADSM.

Enabling Support for IBM 3590 Drives in Existing 3494 Libraries

If you are currently sharing a 3494 library between ADSM and other applications or systems and you enable ADSM’s support for 3590 drives, you need to be careful. The new category that ADSM creates for 3590 scratch volumes can duplicate a category already assigned to another application and cause loss of data. ADSM automatically creates a third category for 3590 scratch volumes by adding one to the existing scratch category for any 3494 libraries defined to ADSM. (See “Categories in 349X Automated Libraries” on page 66 for more information about the categories.)

To prevent loss of data, do either one of the following before enabling 3590 support:

- Update other applications and systems to ensure that there will be no conflicting use of category numbers.
- Delete the existing ADSM library definition and then define it again using a new set of category numbers that do not conflict with categories used by other systems or applications using the library. Do the following:

1. Use an external utility (such as mtlib) to reset all of the ADSM volumes to the insert category.
2. Delete the 3494 library from ADSM.
3. Define the 3494 library to ADSM again, using new category numbers.
4. Enable 3590 support by adding the following line to the server options file (dsmserv.opt):

   ```
   ENABLE3590LIBRARY  YES
   ```

   Stop and start the server to make this change effective.
5. Check in the ADSM volumes you put in the insert category in step 1. For 3490E volumes, use the following command:

```
checkin libvolume libraryname search=yes
```

For 3590 volumes, use the following command:

```
checkin libvolume libraryname search=yes devtype=3590
```

For more information about checking in volumes, see “Informing the Server about New Volumes in a Library” on page 71.

Notes on Operations

The following sections summarize choices and procedures you need to be aware of when operating removable media devices for ADSM. The sections give pointers to additional information.

Mount Operations for Manual Libraries

Volumes are mounted as a result of mount requests from ADSM. For manual libraries, you can monitor the mount requests on the server console or by using an administrative client in mount mode or console mode. Someone you designate as the operator must respond to the mount requests by putting in tape volumes as requested.

For more details, see Chapter 5, “Managing Removable Media Operations” on page 61.

Handling Messages for Automated Libraries

For automated libraries, ADSM works with the library to accomplish volume mounts. Mount messages are not sent to an operator. However, information about problems with the library are still sent to the mount message queue. You can see these messages on administrative clients that have been started with either the mount mode or console mode parameter. However, you cannot use the ADSM REPLY command to respond to these messages. For more details, see Chapter 5, “Managing Removable Media Operations” on page 61.

Cleaning Drives in Automated Libraries

When you want to use a cleaning tape, manually enter the tape into the library and follow the manufacturer’s procedures.

ADSM does not support cleaning operations for SCSI libraries, and does not mount cleaning cartridges or tapes. For most libraries, ADSM uses the cleaning slots, or fixed slots, for data tapes. Keeping a cleaning cartridge in a SCSI library can slow ADSM operations. If the library has a bar-code reader and the cleaning cartridge has a bar-code label, ADSM notes the location of the cleaning cartridge and then ignores it.

Collocation

Collocation is a process by which the server attempts to keep all files belonging to a client node or a client file space on a minimal number of sequential access storage volumes. You set collocation on or off for each sequential access storage pool, which includes tapes. By using collocation, you can reduce the number of volume mounts required when users restore, retrieve, or recall many files. However, when collocation is on, more volume mounts are required when clients store files.

To understand the advantages and disadvantages of collocation, see “Collocation on Sequential Access Storage Pools” on page 121 and “How Collocation Affects Reclamation” on page 131.

Single Drive Libraries

ADSM does not automatically reclaim volumes from a library that contains only one drive because at least two drives in the same device class are required for reclamation. To reclaim a volume in a single drive library, use the MOVE DATA command. See “Moving Files from One Volume to Another Volume” on page 174 for more information.

Chapter 5. Managing Removable Media Operations

ADSM allows you to use and reuse removable media to store data. You need to perform some steps to prepare removable media for initial use by ADSM. You also need to perform some steps to control how and when media is reused.

For manually mounted devices, ADSM sends messages that request that volumes be mounted when they are needed. For devices in automated libraries (such as a tape autochanger), ADSM interacts with the library to mount volumes, but sends messages when the library needs attention from an operator. ADSM also tracks the inventory of media in each automated library.

The sections listed in the following table begin at the indicated pages.

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, "Interface Cross-Reference" on page 467 lists each command and shows if its function is also available on the administrative client GUI.

How ADSM Uses and Reuses Removable Media

ADSM helps you to manage removable media by providing ways to control how removable media are used and reused. The following describes a typical life cycle for

© Copyright IBM Corp. 1995, 1997

a piece of media. The numbers (such as **1**) refer to numbers in Figure 10 on page 62.

1. You label the media **1**. ADSM provides a utility to label media.
2. You put the media on a shelf (for manual libraries only) or you check in the media **2** (for automated libraries only). Check-in makes ADSM aware that volumes are available to it in an automated library.
3. If you choose not to use scratch volumes in the storage pool associated with the device, you define volumes. However, use of scratch volumes is more convenient in most cases.

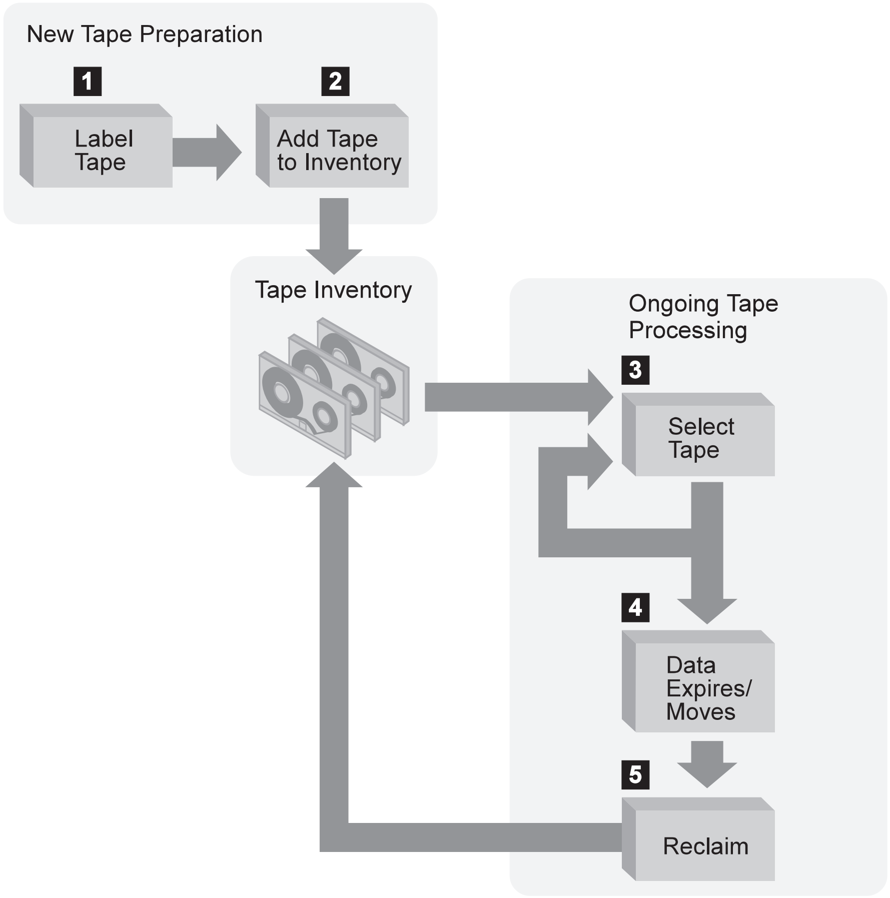


Figure 10. Simplified View of the Life Cycle of a Tape

4. A client sends data to the server for backup, archive, or space management. The server stores the client data on the volume. Which volume the server selects (**3**) depends on:
   - The policy domain to which the client is assigned.

- The management class for the data (either the default management class for the policy set, or the class specified by the client in the client's include/exclude list or file).
- The storage pool specified as the destination in either the management class (for space-managed data) or copy group (for backup or archive data). The storage pool is associated with a device class, which determines which device and which type of media is used.
- Whether the storage pool is already using the maximum number of scratch volumes allowed for it (with the MAXSCRATCH value).
- Whether collocation is enabled for that storage pool. When collocation is enabled, ADSM attempts to place data for different clients or client file spaces on separate volumes.

See Figure 11.

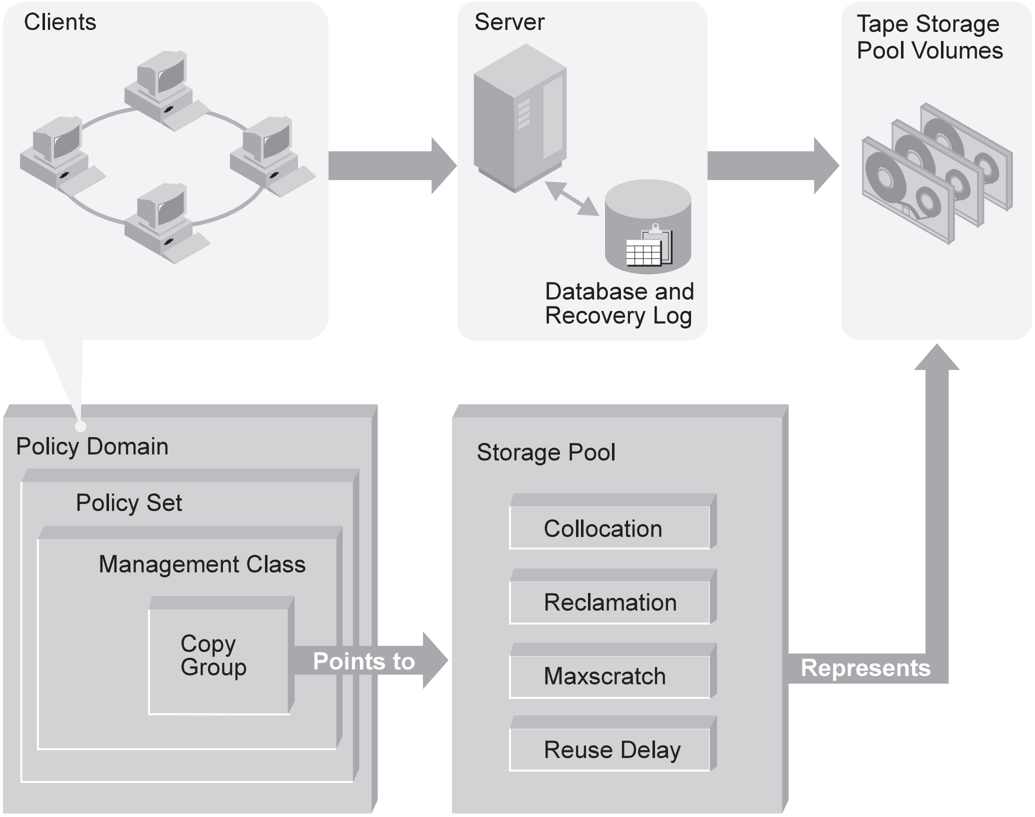


Figure 11. How ADSM Affects Media Use

5. The contents of the volume change over time as a result of:

    - Expiration of files **4** (affected by management class and copy group attributes, and the frequency of expiration processing)
    - Movement and deletion of file spaces (by an ADSM administrator)
    - Automatic reclamation of media by ADSM **5** (in a library with more than one drive)

      The amount of data on the volume and the reclamation threshold set for the storage pool affects when the volume is reclaimed. When the volume is reclaimed, any valid, unexpired data is moved to other volumes.

    If the volume becomes empty because all valid data either expires or is moved to another volume, the volume is available for reuse (after any time delay specified by the REUSEDELAY parameter for the storage pool). The empty volume becomes a scratch volume if it was initially a scratch volume. The volume starts again at step 4 on page 62.

6. You determine when the media has reached its end of life.

    For volumes that you defined (private volumes), check the statistics on the volumes by using the QUERY VOLUME command. The statistics include the number of write passes on a volume (compare with the number of write passes recommended by the manufacturer) and the number of errors on the volume.

    You must move any valid data off a volume that has reached end of life. Then, if the volume is in an automated library, check out the volume from the library. If the volume is not a scratch volume, delete the volume from the ADSM database.

Using Scratch Volumes and Private Volumes

A scratch volume is a labeled volume that is empty or contains no valid data, and can be used to satisfy any request to mount a scratch volume. A private volume is a volume that is in use or owned by an application, and may contain valid data. Volumes that you define to ADSM are private volumes. A private volume is used to satisfy only a request to mount that volume by name. For each storage pool, you must decide whether to use scratch volumes.

If you use scratch volumes, ADSM uses volumes as needed, and returns the volumes to scratch when they become empty (for example, when all data on the volume expires). If you do not use scratch volumes, you must define each volume you want ADSM to use. Volumes that you define to ADSM are private volumes, and do not return to scratch when they become empty.

For more information on defining volumes, see “Preparing Removable Media for ADSM” on page 67.

Private and Scratch Volumes in Automated Libraries

For each automated library, ADSM tracks in its volume inventory for the library whether a volume has scratch or private status. If you allow scratch volumes to be used for a storage pool, ADSM will choose a scratch volume from the scratch volumes that are checked in for the automated library.

When ADSM uses a scratch volume, ADSM changes the volume's status to private by defining it. ADSM tracks whether defined volumes were originally scratch volumes. Volumes that were originally scratch volumes return to scratch status when they become empty. You lose the usage statistics on the volumes when the status of the volumes is changed.

For information on changing the status of a volume in an automated library, see "Changing the Status of a Volume in a Library" on page 77.

The Volume Inventory for an Automated Library

ADSM maintains a volume inventory for each automated library that you define. The inventory for a library includes only those volumes that you have *checked in* to that library.

The list of volumes that are checked in to a library is not necessarily identical to the list of volumes in the storage pools associated with the library. A volume may be checked in to the library but not in a storage pool (a scratch volume). A volume may be defined to a storage pool associated with the library (a private volume), but not in the library's volume inventory.

For more information on how to check in volumes, see "Informing the Server about New Volumes in a Library" on page 71.

Private Volumes in an Automated Library

You may want to use the private status for volumes if you carefully regulate which volumes are used by individual storage pools in your environment. You must define the volumes (DEFINE VOLUME command) for each storage pool. To mount a private volume, you must provide the volume name. If you are doing database backup, dump, or load, or import or export operations, you must list the volumes to use if you want to use private volumes.

Scratch Volumes in an Automated Library

When the ADSM server needs a new volume for a drive in an automated library, the server can choose *any* volume in the library whose status indicates that it is a scratch volume. (A scratch volume is selected only when the MAXSCRATCH value is greater than zero.) After the volume is mounted, its status is changed to private and the volume is automatically defined as part of the storage pool for which the mount request was made. ADSM tracks in the database that this defined volume was originally a scratch volume. When that volume is deleted from the storage pool (for example, all the data it contains expires), the volume returns to scratch status and can be reused by the same or a different storage pool that uses the library.

One of the benefits of using scratch volumes is that different storage pools that share the same automated library can dynamically acquire volumes from the library's pool of scratch volumes. The volumes need not be preallocated to the different storage pools.

Another benefit of using scratch volumes, even if only a single storage pool is associated with an automated library, is that you need not explicitly define all of the volumes for the storage pool using DEFINE VOLUME commands. Volumes are automatically added to and deleted from the storage pool by the server.

If a scratch volume is used for a database backup or export operation, ADSM changes the volume's status to private. The volume returns to the scratch pool only when an ADSM administrator determines that the volume's data is no longer needed, and uses the UPDATE LIBVOLUME command to change the status of the volume to scratch.

Categories in 349X Automated Libraries

A 349X library has an intelligent control unit that tracks the inventory of all volumes in the library. The control unit tracks the *category* number of each volume. A single category number identifies all volumes used for the same purpose or application.

These category numbers are useful when multiple systems share the resources of a single library. Typically, a software application that uses a 349X library device uses volumes only in one or more categories that are reserved for that application. To avoid loss of data, ensure that each application sharing the library uses unique categories.

When a volume is first inserted into the library, either manually or automatically at the convenience I/O station, the volume is assigned to the *insert* category (X'FF00'). A software application, such as ADSM, can then interact with the library control unit to change a volume's category number to a different value. To do this for ADSM, use the CHECKIN LIBVOLUME command (see "Informing the Server about New Volumes in a Library" on page 71).

The number of categories that ADSM requires depends on whether you have enabled support for 3590 drives by using the ENABLE3590LIBRARY parameter in the server options file.

Without 3590 Support Enabled

An ADSM server reserves two different category numbers in each 349X library that it accesses. One of these categories is used for private volumes that belong to ADSM, and the other is used for scratch volumes that belong to ADSM. See "Private and Scratch Volumes in Automated Libraries" on page 65 for more information on private and scratch volumes.

When you define a 349X library, you can specify the category numbers for volumes that ADSM owns in that library by using the PRIVATECATEGORY and SCRATCHCATEGORY parameters. The default values for these parameters are acceptable in most situations.

Attention: If you connect other systems or ADSM servers to a single 349X library, ensure that each uses a unique set of category numbers. Otherwise, two or more systems may try to use the same volume, and cause a corruption or loss of data.

With 3590 Support Enabled

The ENABLE3590LIBRARY server option supports the use of 3590 tape drives within 349X libraries. If support for 3590 drives is enabled for the server, the server reserves three different categories in the 3494 library. The categories are: private, scratch for 3490 drives, and scratch for 3590 drives.

Attention: If you are currently sharing a 3494 library between ADSM and other applications or systems, be careful when enabling 3590 support to prevent loss of data. See "Enabling Support for IBM 3590 Drives in Existing 3494 Libraries" on page 57.

When you define a 349X library, you can specify the category numbers for volumes that ADSM owns in that library by using the PRIVATECATEGORY and SCRATCHCATEGORY parameters. The default values for these parameters are the same as when 3590 support is not enabled. However, ADSM automatically creates a third category, a scratch category for 3590 drives, by adding one to the SCRATCHCATEGORY parameter you specify. For example, suppose you enter the following command:

```
define library my3494 libtype=349x device=3494a
privatecategory=400 scratchcategory=401
```

ADSM then uses the following categories in the new MY3494 library:

400 (X'190')	Private volumes (for both 3490 and 3590 drives)
401 (X'191')	Scratch volumes for 3490 drives
402 (X'192')	Scratch volumes for 3590 drives

As shown in this example, to avoid overlapping categories, the number you specify for the private category must *not* be equal to the scratch category plus 1.

Attention: The default values for the categories may be acceptable in most cases. However, if you connect other systems or ADSM servers to a single 349X library, ensure that each uses unique category numbers. Otherwise, two or more systems may try to use the same volume, and cause a corruption or loss of data.

For information on the ENABLE3590LIBRARY server option, see *ADSM Administrator's Reference*.

Preparing Removable Media for ADSM

For sequential access storage pools with other than FILE device type, you must prepare volumes for use. When the server accesses a sequential access volume, it checks the volume name in the header to ensure that the correct volume is being accessed. To prepare a volume:

1 Label the volume. Any volumes associated with the following device types must be labeled before the server can use them:

- 3570
- 3590
- 4MM
- 8MM
- CARTRIDGE (3490 drives)
- DLT
- OPTICAL

See “Labeling Sequential Access Volumes.”

2 For storage pools in automated libraries, use the CHECKIN LIBVOLUME command to check the volume into the library. See “Informing the Server about New Volumes in a Library” on page 71.

3 You can skip this step if you allowed scratch volumes in the storage pool by specifying a nonzero MAXSCRATCH parameter.

If you have not allowed scratch volumes in the storage pool, identify the volume, by name, to the ADSM server so that it can be accessed later. For details, see “Defining Storage Pool Volumes” on page 163.

Labeling Sequential Access Volumes

ADSM includes a labeling utility that writes labels to new volumes. The utility writes special header information to the beginning of a sequential volume. This header data includes the name of the volume.

Use the DSMLABEL utility from a command line of the operating system. When you use the utility, you provide parameters that specify:

- Whether the utility overwrites a label that already exists on a volume
- The drives to be used for labeling
- If the drives are in an automated library:
  - Whether to search the library for volumes to label
  - Whether to use a bar-code reader, if available
  - Whether to leave volumes in the library after they are labeled

Overwriting Existing Volume Labels

By default, the labeling utility does not overwrite an existing label on a volume. However, if you want to overwrite existing volume labels, you can specify `-overwrite` when you use the DSMLABEL utility.

Attention: By overwriting a volume label, you destroy all of the data that resides on the volume. Use caution when overwriting volume labels to avoid destroying important data.

Identifying Drives to Use for Labeling

You must specify one or more drives for the DSMLABEL utility to use. If you specify more than one drive, the utility attempts to use them concurrently for maximum performance. If the drives are not in an automated library, you are prompted to insert a new volume into a given drive and then to enter the volume name to be written in the label area on the media.

To identify a drive when using the DSMLABEL utility from an operating system command line, specify the drive's device name string with the `-drive=`*`devicename`* parameter. If the drive resides in a SCSI library, you must also identify the drive's element number within the library. The element number can be obtained from the worksheet that was filled in when the library was configured for use by the server. See Appendix A, "Supported Devices and Device Configuration Worksheets" on page 423.

Note: The drives that you use for labeling volumes must not be in use by the server for any other purpose. You can do one of the following:

- Halt the server while you label volumes.
- Prevent the server from using the drives you want to use for labeling volumes. One way to do this is to temporarily delete the ADSM drives and lower the mount limit for the associated device class.

Using an Automated Library for Labeling

You can use an automated library to mount volumes for labeling by supplying a library device name when you use the labeling utility. Specify the library device name with the `-library=`*`devicename`* parameter of the DSMLABEL utility. When you specify a library, the utility assumes that each drive that you specify is in that library. You can specify only one library device. If you want to use more than one library, you must start the utility separately for each one.

You can label volumes one at a time or let ADSM search the library for volumes.

Labeling Volumes One at a Time: The utility assumes that you will insert volumes into the library when prompted to do so. The labeling utility then mounts each inserted volume into a drive and writes a label to it using a name that you enter at a prompt. This is the default mode of operation when you specify a library for use with the labeling utility.

If the library does not have an entry/exit port, the utility prompts you to remove each labeled volume from a drive. If the library has an entry/exit port, the utility by default returns each labeled volume to the entry/exit port of the library. If instead you want labeled volumes to be stored in storage slots inside the library, you must specify the `-keep` parameter when starting DSMLABEL.

Searching the Library: The labeling utility searches all of the storage slots in the library for volumes and labels each one that it finds. You choose this mode by specifying a library for use with the labeling utility, and the `-search` parameter. After a volume is labeled, the volume is returned to its original location in the library, even if the `-keep` parameter was not specified.

As the utility finds and mounts each volume in the library, you are prompted to enter the name of each volume. If the library has a bar-code reader, you can avoid the prompting and data entry by using the bar-code reader.

Using a Bar-Code Reader on SCSI Libraries: If the library has a bar-code reader, the DSMLABEL utility can use the reader to obtain volume names, instead of prompting you for volume names. Use the parameters `-search` and `-barcode` when starting the DSMLABEL utility.

If a volume has six or fewer characters on its bar-code label, the utility uses the characters on the label as the name for the volume. If a volume has no bar-code label or has a label with more than six characters, the volume is not labeled.

Using a 349X Library for Labeling

The labeling utility only attempts to label volumes that reside in the INSERT category in the library. All other volumes are ignored by the DSMLABEL utility. This precaution prevents the inadvertent destruction of data on volumes being actively used by other systems connected to the library device.

Formatting Optical Volumes

The media for optical drives needs to be formatted before use. You can use the DSMLABEL utility to format and label 5.25-inch optical disks. Use the `-format` parameter when starting the DSMLABEL utility.

Volume Labeling Examples

The following are some examples of labeling volumes.

Labeling All of the Volumes in a SCSI Library: Suppose you want to label all of the volumes that reside in a SCSI library. The library device is an Exabyte EXB-120 and, although it contains four drives, you only want to use two of them to label volumes. The drives are at element addresses 116 and 117. Enter the following command:

```
> dsmlabel -drive=/dev/rmt/0mt,116 -drive=/dev/rmt/1mt,117
  -library=/dev/0lb -search
```

Labeling New Volumes in a SCSI Library: Suppose you want to label a few new volumes for use in your existing Exabyte EXB-120 library. You want to manually insert each new volume into the library, and you want the volumes to be placed in storage slots inside the library after their labels are written. You know that none of the new volumes contains valid data, so it is acceptable to overwrite existing volume labels. You only want to use one of the library's four drives for these operations. Enter the following command:

```
> dsmlabel -drive=/dev/rmt/0mt,116 -library=/dev/rmt/0lb -overwrite
  -keep
```

Labeling Volumes Using a Manual Drive: Suppose you want to label a few new volumes using a tape drive that is not part of an automated library. The drive is attached at SCSI address 5. Enter the following command:

```
> dsmlabel -drive=/dev/rmt/5mt
```

Labeling Volumes in an Insert Category in a 3494 Library: Suppose you want to label all of the volumes that are in the INSERT category in an IBM 3494 tape library and you want to use both of the two drives within the library. Enter the following command:

```
> dsmlabel -drive=/dev/rmt/1st -drive=/dev/rmt/2st -library=3494a
```

Informing the Server about New Volumes in a Library

Task	Required Privilege Class
Inform the server when a new volume is available in an automated library	System or unrestricted storage

You inform the server that a new volume is available in an automated library by checking in the volume with the CHECKIN LIBVOLUME command. When a volume is checked in, the server adds the volume to its library volume inventory.

Note: Do not mix volumes with bar-code labels and volumes without bar-code labels in a library device because bar-code scanning can take a long time for unlabeled volumes.

Processing Time: Because the CHECKIN LIBVOLUME command involves device access, it may take a long time to complete. For this reason, the command always executes as a background process.

When you check in a volume, you must supply the name of the library and the status of the volume (private or scratch).

To check in one or just a few volumes, you can specify the name of the volume with the command, and issue the command for each volume. See “Checking In Volumes One at a Time (SEARCH=NO)” on page 73.

To check in a larger number of volumes, you can use the search capability of the CHECKIN command. See “Searching the Library (SEARCH=YES)” on page 73.

When using the CHECKIN LIBVOLUME command, be prepared to supply some or all of the following information:

Library name
Specifies the name of the library where the storage volume is to be located.

Volume name
Specifies the name of the storage volume being checked in.

Status
Specifies the status that is assigned to the storage volume being checked in. See “Specifying the Status of a Volume” on page 74.

Check label
Specifies whether ADSM should read sequential media labels of volumes during CHECKIN command processing, or use a bar-code reader. See “Checking Media Labels” on page 74.

Swap
Specifies whether ADSM will initiate a swap operation when an empty slot is not available during CHECKIN command processing. See “Allowing Swapping of Volumes When the Library Is Full” on page 74.

Mount wait
Specifies the maximum length of time, in minutes, to wait for a storage volume to be mounted.

Search
Specifies whether ADSM searches the library for volumes that have not been checked in. See “Checking In Volumes One at a Time (SEARCH=NO)” on page 73 and “Searching the Library (SEARCH=YES)” on page 73.

Device type
This parameter only applies to 349X libraries containing 3590 devices. This parameter allows you to specify the device type for the volume being checked in.

Example: Checking In One Volume

To check in volume VOL001 manually, enter the following command:

```
checkin libvolume tapelib vol001 search=no status=scratch
```

You are prompted to insert a cartridge into one of the slots in the library. These slots are identified by element addresses. You can find these element addresses in the worksheet for the device (use Table 18 on page 426 to find the worksheet).

For example, ADSM finds that the first empty slot is at element address 5. The message is:

```
ANR8306I 001: Insert 8MM volume VOL001 R/W in slot with element
address 5 of library TAPELIB within 60 minutes; issue 'REPLY' along
with the request ID when ready.
```

Check the worksheet for the device if you are not sure of the location of element address 5 in the library. When you have inserted the volume as requested, respond to the message from an ADSM administrative client. Use the request number (the number at the beginning of the mount request):

```
reply 1
```

Checking In Volumes One at a Time (SEARCH=NO)

Specify this option if you want to check in only a single volume that is not currently in the library. ADSM requests that the mount operator load the volume in the entry/exit port of the library.

If the library does not have an entry/exit port, ADSM requests that the mount operator load the volume into a slot within the library. The request specifies the location with an *element address*. For any library or autochanger that does not have an entry/exit port, you need to know the element addresses for the cartridge slots and drives. If there is no worksheet listed for your device in Table 18 on page 426, see the documentation that came with your library.

Note: Element addresses are sometimes numbered starting with a number other than one. Check the worksheet to be sure.

You can use this option for a 349X library, to search for volumes that have already been inserted into the library via the convenience or bulk I/O station. If the volume has already been inserted, the server finds and processes it. If not, you can insert the volume into the I/O station during the processing of the command.

Searching the Library (SEARCH=YES)

Specify this option if you want ADSM to automatically search the library for new volumes that have not already been added to the library volume inventory. Use this mode when you have a large number of volumes to check in, and you want to avoid issuing an explicit CHECKIN LIBVOLUME command for each volume.

With this option, you cannot specify a volume name because the server searches for multiple new volumes in the library.

For example, for a SCSI library you can simply open the library access door, place all of the new volumes in unused slots, close the door, and issue the CHECKIN LIBVOLUME command with SEARCH=YES.

If you are using a 349X library, the server searches only for new volumes in the following categories:

- Insert
- ADSM’s private category (PRIVATECATEGORY, specified when the library was defined to ADSM)
- ADSM’s scratch category (SCRATCHCATEGORY, specified when the library was defined to ADSM)

  If 3590 support is enabled, the server searches for two scratch categories: SCRATCHCATEGORY, and SCRATCHCATEGORY + 1.

This restriction prevents the server from using volumes owned by another application that is accessing the library simultaneously.

Checking Media Labels

When you check in a volume, you can specify whether ADSM should read the labels of the media during check-in processing. When label-checking is on, ADSM mounts each volume to read the internal label and only checks in a volume if it is properly labeled. This can prevent future errors when volumes are actually used in storage pools, but also increases processing time at check-in. For information on how to label new volumes, see “Preparing Removable Media for ADSM” on page 67.

Using a Library’s Bar-Code Reader: To save time when checking in many volumes for a library with a bar-code reader, you can specify that the check-in process use the bar-code reader. If a volume has a bar-code label with six or fewer characters, ADSM uses the characters on the label as the name for the volume being checked in. If a volume has no label or has a label with more than six characters, ADSM mounts the volumes in a drive and attempts to read the recorded label.

For example, to use the bar-code reader to check in all volumes found in the TAPELIB library as scratch volumes, enter the following command:

```
checkin libvolume tapelib search=yes status=scratch
checklabel=barcode
```

Specifying the Status of a Volume

If you check in a volume that has already been defined in a storage pool, you must use a volume status of private. This status ensures that the volume is not overwritten when a scratch mount is requested. The server does not check in a volume with scratch status when that volume already belongs to a storage pool.

Allowing Swapping of Volumes When the Library Is Full

If no empty slots are available in the library when you are checking in volumes, the check-in fails unless you allow *swapping*. If you allow swapping and the library is full, ADSM selects a volume to eject before checking in the volume you requested.

ADSM selects the volume to eject by checking first for any available scratch volume, then for the least frequently mounted volume.

Maintaining the Volume Inventory

With ADSM, you maintain your volume inventory by:

- Controlling ADSM access to volumes
- Reusing tapes in storage pools

For automated libraries, some additional tasks are required. See "Managing the Volume Inventory in Automated Libraries" on page 76.

Controlling ADSM Access to Volumes

ADSM expects to be able to access all volumes it knows about. For example, ADSM tries to fill up tape volumes. If a volume containing client data is only partially full, ADSM will later request that volume be mounted to store additional data. If the volume cannot be mounted, an error occurs.

To make volumes that are not full unavailable to ADSM, you can change the access mode of the volumes. For example, use the UPDATE VOLUME command with ACCESS=UNAVAILABLE. The server will not attempt to mount a volume that has an access mode of unavailable.

If you want to make volumes unavailable to send the data they contain offsite for safekeeping, a more controlled way to do this is to use a copy storage pool. You can back up your primary storage pools to a copy storage pool and then send the copy storage pool volumes offsite. You can track these copy storage pool volumes by changing their access mode to offsite, and updating the volume history to identify their location. For more information, see "Backing Up Storage Pools" on page 142.

Reusing Tapes in Storage Pools

To reuse tapes in ADSM storage pools, you must do two things:

- Run expiration processing regularly so that client files that have *expired* (are no longer valid) are deleted. See "Expiration Processing of Client Files."
- Move data to consolidate valid, unexpired files onto fewer tapes.

  ADSM offers an automated process called *reclamation* for manual or automated libraries with more than one drive. See "Reclamation for a Library with Multiple Drives" on page 76.

  For manual or automated libraries with only one drive, you must use a more manual process. See "Reclamation for a Library with One Drive" on page 76.

Expiration Processing of Client Files

Expiration processing deletes from the ADSM database information about any client files that are no longer valid according to the policies you have set. For example, suppose four backup versions of a file exist in ADSM server storage, and only three versions are allowed in the backup policy (the management class) for the file. Expiration processing deletes information about the oldest of the four versions of the file. The space that the file occupied in the storage pool can then be reclaimed.

You can run expiration processing by using one or both of the following methods:

- Use the ADSM command EXPIRE INVENTORY. See “Running Expiration Processing to Delete Expired Files” on page 213.
- Set the server option for the expiration interval, so that expiration processing runs periodically. You can change server options by editing the server options file, dsmserv.opt. For information on server options, see *ADSM Administrator’s Reference*.

Reclamation for a Library with Multiple Drives

If you are using libraries with multiple drives, you can have ADSM reclaim volumes that pass a *reclamation threshold*, a percentage of unused space on the volume. The reclamation threshold is set for each storage pool. See “Space Reclamation for Sequential Access Storage Pools” on page 127.

Reclamation for a Library with One Drive

To reclaim tapes in a library that has only one drive, you must use the ADSM command MOVE DATA. See “Reclamation in a Single-Drive Library” on page 132.

Reusing Volumes Used for Database Backups and Export Operations

When you back up the database or export server information, ADSM records information about the volumes used for these operations in the *volume history* file. ADSM will not allow you to reuse these volumes until you delete the volume information from the volume history file. To reuse volumes that have previously been used for database backup or export, use the DELETE VOLHISTORY command. For information about the volume history file, see “Saving the Volume History File” on page 340.

Managing the Volume Inventory in Automated Libraries

ADSM tracks the scratch and private volumes available in an automated library through a *library volume inventory*. ADSM maintains an inventory for each automated library. The library volume inventory is separate from the inventory of volumes for each storage pool. To add a volume to a library’s volume inventory, you *check in* a volume to that ADSM library. For details on the check-in procedure, see “Informing the Server about New Volumes in a Library” on page 71.

To ensure that ADSM’s library volume inventory remains accurate, you must *check out* volumes when you need to physically remove volumes from a SCSI library device. When you check out a volume that is being used by a storage pool, the volume remains in the storage pool, but ADSM marks it as not available for mounting.

While a volume is in the library volume inventory, you can change its status from scratch to private, or from private to scratch.

To check whether ADSM's library volume inventory is consistent with the volumes that are physically in the library, you can audit the library. The inventory can become inaccurate if volumes are moved in and out of the library without informing the server via volume check-in or check-out.

The sections listed in the following table begin at the indicated pages.

Section	Page
Tasks:	
Changing the status of a volume in a library	77
Removing volumes from a library	77
Returning volumes to a library	78
Auditing a library's volume inventory	78

Changing the Status of a Volume in a Library

Task	Required Privilege Class
Change the status of a volume in an automated library	System or unrestricted storage

The UPDATE LIBVOLUME command lets you change the status of a volume in an automated library from scratch to private, or private to scratch. However, you cannot change the status of a volume from private to scratch if the volume belongs to a storage pool.

You can use this command if you make a mistake when checking in volumes to the library and assign the volumes the wrong status.

Removing Volumes from a Library

Task	Required Privilege Class
Remove volumes from a library	System or unrestricted storage

You may wish to remove a volume from an automated library. The following are examples:

- You have exported data to a volume in the library and want to take it to another system for an import operation.
- All of the volumes in the library are full, and you want to remove some that are not likely to be accessed in order to make room for new volumes that can be used to store more data.

To remove a volume from an automated library, use the CHECKOUT LIBVOLUME command. By default, the server mounts the volume being checked out and verifies the internal label. When the label is verified, the server removes the volume from the library volume inventory, and then moves it to the entry/exit port of the library. If the library does not have an entry/exit port, ADSM requests that the mount operator remove the volume from a drive within the library.

If you check out a volume that is defined in a storage pool, the server may attempt to access it later to read or write data. If this happens, the server requests that the volume be checked in.

Returning Volumes to a Library

Task	Required Privilege Class
Return volumes to a library	System or unrestricted storage

When you check out a volume that is defined to a storage pool, to make the volume available again, you must do the following:

1. Check in the volume for the library, with private status. Use the CHECKIN LIBVOLUME command with the parameter STATUS=PRIVATE.
2. Update the volume's ACCESS value. You must change the access from unavailable to read/write or read-only. Use the UPDATE VOLUME command with the ACCESS parameter.

Auditing a Library's Volume Inventory

Task	Required Privilege Class
Audit the volume inventory of a library	System or unrestricted storage

You can audit an automated library to ensure that ADSM's library volume inventory is consistent with the volumes that physically reside in the library. You may want to do this if the server's library volume inventory is disturbed due to manual intervention or movement of volumes within the library, or to problems with the server database. Use the AUDIT LIBRARY command to restore the inventory to a consistent state. Missing volumes are deleted and the locations of the moved volumes are updated; however, new volumes are not added during an audit. Unless your library has a bar-code reader, the server mounts each volume during the audit process to verify the internal labels on volumes.

Note: Audit library processing waits until all volumes have been dismounted from drives within the specified library. If one or more volumes are mounted, but are in the IDLE state, you can force the volumes to be dismounted by issuing the DISMOUNT VOLUME command. Otherwise, the audit operation remains in a wait state until the idle volumes have been dismounted (the idle volumes are dismounted after the MOUNTRETENTION period expires).

Using a Library's Bar-Code Reader

To save time when auditing a library that has a bar-code reader, you can specify that the audit process use the bar-code reader to verify the identity of volumes. If a volume has a bar-code label with six characters or less, ADSM uses the characters on the label as the name for the volume during the audit. The volume is *not* mounted to verify that the external bar-code name matches the internal, recorded volume name.

If a volume has no label or has a label with more than six characters, ADSM mounts the volume in a drive and attempts to read the recorded label.

For example, to audit the TAPELIB library using its bar-code reader, enter the following command:

```
audit library tapelib checklabel=barcode
```

Managing Media Mount Operations

ADSM generates an operator request when the ADSM server requires some kind of action with a drive or library. For example, when ADSM requires a volume mount in a manual library, it generates a request. The server sends mount request status messages to the server console and to all administrative clients that have been started with either the *mount mode* or the *console mode* parameter.

In many cases, an operator request has a time limit. If the requested action is not performed within the time limit, the operation times out and fails.

For most types of requests, such as volume mounts, the server detects when the operator performs the action. The operator does not usually need to respond to the ADSM server after carrying out the requested activity. However, sometimes the server cannot detect the completion of the requested action. When the server requires a reply, the message that is displayed by the server requests that the operator reply when the activity has been completed. For example, a request to mount a scratch volume requires that the operator reply when a scratch volume has been placed in the drive. ADSM waits for a reply to prevent the use of the wrong volume.

For most of the requests associated with automated (SCSI) libraries, the server cannot automatically detect when a requested activity has been completed. For such requests, the operator must use the REPLY command on the command line of an ADSM administrative client.

Using the Administrative Client for Mount Messages

The server sends mount request status messages to the server console and to all administrative clients that have been started with either the special *mount mode* or *console mode* parameter. For example, to start the OS/2 administrative client in mount mode, enter this command:

```
> dsmadmc -mountmode
```

Requesting Information about Pending Operator Requests

Task	Required Privilege Class
Request information about operator requests or mounted volumes	Any administrator

You can get information about pending operator requests either by using the QUERY REQUEST command or by checking the mount message queue on an administrative client started in mount mode.

When you issue the QUERY REQUEST command, ADSM displays requested actions and the amount of time remaining before the requests time out. For example, you enter the command as follows:

```
query request
```

The following shows an example of a response to the command:

```
ANR8352I Requests outstanding:
ANR8326I 001: Mount 8MM volume DSM001 R/W in drive TAPE01 (/dev/rmt/1mt) of
library MANUAL8MM within 60 minutes.
```

Replying to Operator Requests

Task	Required Privilege Class
Reply to operator requests	Operator

When the server requires that an explicit reply be provided when a mount request is completed, you can reply via the ADSM REPLY command. The first parameter for this command is the request identification number that tells the server which of the pending operator requests has been completed. This 3-digit number is always displayed as part of the request message. It can also be obtained by issuing a QUERY REQUEST command. If the request requires the operator to provide a device to be used for the mount, the second parameter for this command is a device name.

For example, enter the following command to respond to request 001 for tape drive TAPE01:

```
reply 1
```

Canceling an Operator Request

Task	Required Privilege Class
Cancel operator requests	Operator

If a mount request for a manual library cannot be satisfied, you can issue the CANCEL REQUEST command. This command forces the server to cancel the request and cause the operation that needed the requested volume to fail.

The CANCEL REQUEST command must include the request identification number. This number is included in the request message. You can also obtain it by issuing a QUERY REQUEST command, as described in "Requesting Information about Pending Operator Requests" on page 80.

You can specify the PERMANENT parameter if you want to mark the requested volume as UNAVAILABLE. This process is useful if, for example, the volume has been moved to a remote site or is otherwise inaccessible. By specifying PERMANENT, you ensure that the server does not try to mount the requested volume again.

For most of the requests associated with automated (SCSI) libraries, an operator must perform a hardware or system action to cancel the requested mount. For such requests, the ADSM CANCEL REQUEST command is not accepted by the server.

Responding to Requests for Volume Check-in

If the server cannot find a particular volume it needs to be mounted in an automated library, the server requests that the operator check in the volume. For example, a client requests that an archived file be retrieved. The file was archived in a storage pool in an automated library. The server looks for the volume containing the file in the automated library, but cannot find the volume. The server then requests that the volume be checked in.

If the volume that the server requests is available, put the volume in the library and check in the volume using the normal procedures ("Informing the Server about New Volumes in a Library" on page 71).

If the volume requested is unavailable (lost or destroyed), update the access mode of the volume to UNAVAILABLE by using the UPDATE VOLUME command. Then cancel the server's request for check-in by using the CANCEL REQUEST command. (Do *not* cancel the client process that caused the request.) To get the ID of the request to cancel, use the QUERY REQUEST command.

If you do not respond to the server's check-in request within the mount-wait period of the device class for the storage pool, the server marks the volume as unavailable.

Determining Which Volumes are Mounted

Task	Required Privilege Class
Request information about which volumes are mounted	Operator

For a report of all volumes currently mounted for use by the server, you can issue the QUERY MOUNT command. The report shows which volumes are mounted, which drives have accessed them, and if the volumes are currently being used.

Dismounting an Idle Volume

Task	Required Privilege Class
Request a volume dismount	Operator

After a volume becomes idle, the server keeps it mounted for a time specified by the mount retention parameter for the device class. Using mount retention can reduce the access time if volumes are repeatedly used.

An administrator can explicitly request that an idle volume be dismounted by issuing the DISMOUNT VOLUME command. This command causes the server to dismount the named volume from the drive in which it is currently mounted.

For information about setting mount retention times, see “Mount Retention Period” on page 94.

Chapter 6. Defining Drives and Libraries

Use this chapter for details on defining drives and libraries. In ADSM, a *library* is a collection of drives for which volume mounts are accomplished by using a common method, for example, by an operator or by robotic mechanisms. A *drive* is a hardware device capable of performing operations on a specific type of sequential media. ADSM categorizes each drive using a *device type* value that is based on the attributes of the hardware device.

One or more drives can be defined as part of each library. For examples of defining libraries and drives, see Chapter 4, “Using Removable Media Devices with ADSM” on page 37.

If the DEVCONFIG option is included in the dsmserv.opt file, the files you specify with that option are automatically updated whenever a library, drive, or device class is defined, updated, or deleted.

The sections listed in the following table begin at the indicated pages.

Section	**Page**
Concepts:	
How ADSM Uses Sequential Access Devices	83
Tasks:	
Defining and Managing Libraries	85
Defining and Managing Drives	88

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator’s Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, “Interface Cross-Reference” on page 467 lists each command and shows if its function is also available on the administrative client GUI.

How ADSM Uses Sequential Access Devices

Each ADSM library is a collection of drives. A device class, which governs how data is stored, is associated with one *library*. When you define a storage pool, you associate the pool with a device class. Volumes are associated with pools. Figure 12 on page 84 shows these relationships.

© Copyright IBM Corp. 1995, 1997

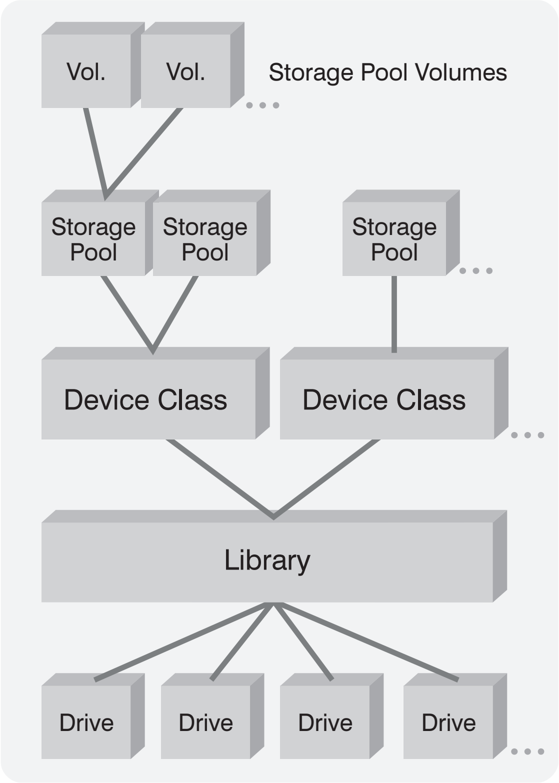


Figure 12. Relationships between Storage and Device Objects

When the ADSM server determines that data is to be stored into or retrieved from a storage pool, it performs the following procedure:

1. Selects a volume from the given storage pool. The selection is based on the type of operation:

   Retrieval The name of the volume is stored in the server database.

   Store If a defined volume in the storage pool can be used for the data being stored, the server chooses this volume name.

   If no defined volumes in the storage pool can be used for the data, and if the MAXSCRATCH parameter of the storage pool permits it, the server may try a *scratch mount*.

2. Determines the name of the library containing the drives that can be used for the operation by checking the device class associated with the storage pool.

   - The server evaluates the status of each drive in the library until an available drive is found or until all drives have been checked. Drive status can be:
     - The drive is offline. Attempt to *VARY ON* the drive.
     - The drive is busy and cannot be used for this mount.

   - The drive is in an error state and cannot be used for this mount.
   - The drive is available and can be used for this mount.

3. Performs the volume mount operation:

   - If the library is manually operated, the server displays request messages for a mount operator, asks that the desired volume, or a scratch volume, be mounted in the selected drive.
   - If the library is automated, the server directs a robotic device to move the volume from a storage slot into the selected drive. No manual intervention is required.

     If a scratch mount is requested, the server checks the library's volume inventory to see if there is a volume with a status of SCRATCH. The volume inventory is established and managed by using the commands described in "Managing the Volume Inventory in Automated Libraries" on page 76. Volume status is described in "Private and Scratch Volumes in Automated Libraries" on page 65. If a scratch volume is found, its volume status is changed to PRIVATE and it is mounted in the drive. Eventually, it is automatically defined as part of the original storage pool. However, if the library's volume inventory does not contain any volumes with a status of SCRATCH, the mount request fails.

4. Dismounts the volume from the drive when it has finished accessing the volume.

   - If the library is manually operated, the server ejects the volume from the drive so that a mount operator can place it in an appropriate storage location.
   - If the library is automated, the server interacts with a robotic device to move the volume from the drive back to its original storage slot in the library.

Defining and Managing Libraries

As an administrator, you manage all ADSM libraries. After you determine the type of library you require, you must define that library to ADSM. For information on ADSM library types, see "MANUAL Libraries" on page 18, "SCSI Libraries" on page 19, "349X Libraries" on page 19, and "External Libraries" on page 19.

Defining Libraries

Task	Required Privilege Class
Define libraries	System or unrestricted storage

Before you can use a drive, you must first define the library to which the drive belongs. This is true for both manually mounted drives and drives in automated libraries.

To define a new library, use the DEFINE LIBRARY command. At a minimum, you must specify the library name and the library type. For example, suppose you have several stand-alone tape drives that will need to be mounted manually by an operator. You could define a library named MANUALMOUNT for these drives by using the following command:

```
define library manualmount libtype=manual
```

For automated libraries, you use the DEFINE LIBRARY command to define a SCSI or 349X library along with the DEVICE parameter. The DEVICE parameter is required and specifies the name of the device by which the library's robotic mechanism is known.

The following example can apply to any SCSI library. It assumes that you have configured the robot device driver as described in "Installing Device Drivers" on page 40, and determined the appropriate device name string as shown in the example. If you have an Exabyte EXB-120 device, you may define a library named ROBOTMOUNT using the following command:

```
define library robotmount libtype=scsi device=/dev/rmt/0lb
```

If you have an IBM 3590 B11 device with device special file name /dev/rmt/0smc, you can define a library named MAINMOUNT using the following command:

```
define library mainmount libtype=scsi device=/dev/rmt/0smc
```

Suppose you have an IBM 3494 Tape Library Dataserver connected to your system, and that you have defined the symbolic name 3494a for the library in the /etc/ibmatl.conf file. You can define a library named AUTOMOUNT using the following command:

```
define library automount libtype=349x device=3494a
```

Requesting Information about Libraries

Task	Required Privilege Class
Request information about libraries	Any administrator

You can request information about any or all libraries by using the QUERY LIBRARY command. Either a standard or a detailed report can be requested.

For example, to display information about your libraries, issue the following command:

```
query library
```

Figure 13 on page 87 is an example of the output from this command:

```
Library    Library    Device        Private     Scratch     External
Name       Type                     Category    Category    Manager
-------    -------    -------       --------    --------    --------
MANLIB     MANUAL
EXB        SCSI       /dev/rmt/2lb
```

Figure 13. Standard Query Library Report

Updating Libraries

Task	Required Privilege Class
Update libraries	System or unrestricted storage

You can update a previously defined library by issuing the UPDATE LIBRARY command.

Automated Libraries (SCSI)

The only attribute on a SCSI library that can be updated is the device name. This may be necessary if your system or device is reconfigured, causing the device name to change. For example, you have defined a SCSI library named AUTOLIB, but the device is reconfigured and its name is changed. You can then issue the following command to inform the ADSM server of the change:

```
update library autolib device=/dev/rmt/1lb
```

Note: MANUAL libraries, which have no DEVICE attribute, cannot be updated.

Deleting Libraries

Task	Required Privilege Class
Delete libraries	System or unrestricted storage

Before deleting a library with the DELETE LIBRARY command, all of the drives that have been defined as part of the library must be deleted. See “Deleting Drives” on page 89.

For example, suppose you wish to delete a library named MANUALMOUNT. After deleting all of the drives defined as part of this library, you could issue the following command to delete the library itself:

```
delete library manualmount
```

Defining and Managing Drives

Administrators can define, query, update, and delete drives.

Defining Drives

Task	Required Privilege Class
Define drives	System or unrestricted storage

To inform the server about a drive that can be used to access storage volumes, issue the DEFINE DRIVE command. When issuing this command, you must provide some or all of the following information:

Library name
The name of the library in which the drive resides.

Drive name
The name assigned to the drive.

Device name
The device name to be used to access the drive.

Element address
The element address of the drive. The ELEMENT parameter is required for drives defined in SCSI libraries that support more than one drive; it is optional for SCSI libraries that support only one drive. This parameter is invalid for drives defined in non-SCSI libraries. The element address is a number that indicates the physical location of a drive within an automated library. ADSM needs the element address to connect the physical location of the drive to the drive's SCSI address.

For example, to define a drive that belongs to the manual library named MANLIB, enter this command:

```
define drive manlib tapedrv3 device=/dev/rmt/1mt
```

Requesting Information about Drives

Task	Required Privilege Class
Request information about drives	Any administrator

You can request information about drives by using the QUERY DRIVE command. This command accepts wildcard characters for both a library name and a drive name.

For example, to query all drives associated with your server, enter the following command:

```
query drive
```

The following shows an example of the results of this command.

```
Library   Drive    Device         Device           Element
Name      Name     Type
--------  -------  ---------      ------           -------
MANLIB    8MM.0    8MM            /dev/rmt/1mt
AUTOLIB   8MM.2    8MM            /dev/rmt/2mt     116
```

Updating Drives

Task	Required Privilege Class
Update drives	System or unrestricted storage

You can change the attributes of a drive by issuing the UPDATE DRIVE command. For example, you can change the device name if you are reconfiguring your system. If the drive resides within a SCSI library, its ELEMENT attribute can also be updated.

A drive cannot be updated if it is currently in use. If a drive has a volume mounted, but the volume is currently idle, you can explicitly dismount the volume as described in “Dismounting an Idle Volume” on page 82.

For example, suppose you have a drive DRIVE3 and you want to change the element address to 119. Enter the following command:

```
update drive auto drive3 element=119
```

Deleting Drives

Task	Required Privilege Class
Delete drives	System or unrestricted storage

A drive cannot be deleted if it is currently in use. If a drive has a volume mounted, but the volume is currently idle, you can explicitly dismount the volume as described in “Dismounting an Idle Volume” on page 82.

Note: A library cannot be deleted until all of the drives defined within it are deleted.

Chapter 7. Defining Device Classes

A device class represents a device type that can be used by ADSM. ADSM uses the device class to determine which device and storage volume type to use to:

- Store backup, archive, or space-managed data (primary storage pools)
- Store copies of primary storage pool data (copy storage pools)
- Store database backups
- Export or import ADSM data

One device class can be associated with multiple storage pools. Each storage pool is associated with just one device class.

Each device class is characterized by its *device type*, which indicates the type of storage volumes that are used to store data. For random access storage, ADSM supports only the DISK device class. The DISK device class is predefined by ADSM. However, you can define many storage pools that are categorized by the DISK device class.

For sequential access storage, ADSM supports the following device types:

3570 IBM 3570 cartridge tape devices

3590 IBM 3590 tape drives

4MM 4mm tape drives

8MM 8mm tape drives

CARTRIDGE IBM 3490E tape drives

DLT Digital linear tape (DLT) drives

FILE Storage volumes that are actually files in the file system of the server machine

OPTICAL Optical drives that use 5.25-inch rewritable optical cartridges

WORM Optical drives that use 5.25-inch write-once read-many optical cartridges

WORM12 Optical drives that use 12-inch write-once ready-many optical cartridges

WORM14 Optical drives that use 14-inch write-once ready-many optical cartridges

The sections listed in the following table begin at the indicated pages.

Section	Page
Tasks:	
Defining and updating device classes for tape	93
Defining and updating device classes for optical drives	99

© Copyright IBM Corp. 1995, 1997

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, "Interface Cross-Reference" on page 467 lists each command and shows if its function is also available on the administrative client GUI.

Defining and Updating Device Classes for Sequential Media

Task	Required Privilege Class
Define device classes	System or unrestricted storage

You can define multiple device classes for each device type. For example, you may need to specify different attributes for two different storage pools that use the same type of tape drive. Variations may be required that are not specific to the device, but rather to how you want to use the device (for example, mount retention or mount limit).

If you are using device classes that have any of the following device types, you must define libraries and drives to the ADSM server before you define device classes to access your sequential media:

- 3570
- 3590
- 4MM
- 8MM
- CARTRIDGE
- DLT
- OPTICAL
- WORM
- WORM12
- WORM14

See Chapter 6, "Defining Drives and Libraries" on page 83 for information about defining drives and libraries.

If you include the DEVCONFIG option in the dsmserv.opt file, the files you specify with that option are automatically updated with the results of this command. When you use this option, the files specified are automatically updated whenever a device class, library, or drive is defined, updated, or deleted.

The following sections explain the device classes for each supported device type.

Defining and Updating Device Classes for Tape Devices

To use the tape devices, you must define a device class by issuing a DEFINE DEVCLASS command with the DEVTYPE parameter.

Device	ADSM Device Type
4mm	4MM
8mm	8MM
DLT	DLT
IBM 3490E	CARTRIDGE
IBM 3570	3570
IBM 3590	3590

Other parameters specify how to manage data storage operations involving the new device class:

- MOUNTLIMIT
- MOUNTWAIT
- MOUNTRETENTION
- PREFIX
- FORMAT
- ESTCAPACITY
- LIBRARY

You can update the device class information by issuing the UPDATE DEVCLASS command.

Mount Limit

You can limit the number of concurrent volume mounts so that your storage device resources are properly managed. The *MOUNTLIMIT* parameter specifies the maximum number of volumes that can be simultaneously mounted for a device class.

The default mount limit value is 1; the maximum value for this parameter is 256.

When selecting a mount limit for a device class, be sure to consider the following questions:

- How many storage devices are connected to your system?

  Do not specify a mount limit value that is greater than the number of associated available drives in your installation. If the server tries to mount as many volumes as specified by the mount limit and no drives are available for the required volume, an error occurs and client sessions may be terminated.

  Note: ADSM cannot share drives between multiple device classes.

- Do you want reclamation of volumes to occur automatically?

  If the mount limit is set to one, then ADSM cannot automatically reclaim available space on storage volumes. During the automatic reclamation process, ADSM requires two drives to move data from one volume to another.

If you set the mount limit to one and want to reclaim volumes, you must use a more manual process, using the MOVE DATA command. See “Reclamation in a Single-Drive Library” on page 132.

- How many ADSM processes do you want to run at the same time, using devices in this device class?

  ADSM automatically cancels some processes to run other, higher priority processes. If the server is using all available drives in a device class to complete higher priority processes, lower priority processes must wait until a drive becomes available. For example, ADSM cancels the process for a client backing up directly to tape if the drive being used is needed for a server migration or tape reclamation process. ADSM cancels a tape reclamation process if the drive being used is needed for a client restore operation.

  If processes are often canceled by other processes, consider whether you can make more drives available for ADSM use. Otherwise, review your scheduling of operations to reduce the contention for drives.

Mount Wait Period

The *MOUNTWAIT* parameter specifies the maximum amount of time, in minutes, that the server waits for a manual (or operator controlled) volume mount request to be satisfied before canceling the request. The default mount wait period is 60 minutes; the maximum value for this parameter is 9999 minutes.

Mount Retention Period

The *MOUNTRETENTION* parameter specifies the amount of time that a mounted volume should remain mounted after its last I/O activity. If this idle time limit is reached, the server dismounts the volume. The default mount retention period is 60 minutes; the maximum value for this parameter is 9999 minutes.

For example, if the mount retention value is 60, and a mounted volume remains idle for 60 minutes, then the server dismounts the volume.

If a volume is used frequently, you can improve performance by setting a longer mount retention period to avoid unnecessary mount and dismount operations.

If mount operations are being handled by manual, operator-assisted activities, you may want to use a large mount retention period. For example, if only one operator supports your entire operation on a weekend, then define a long mount retention period so that the operator is not being asked to mount volumes every few minutes.

While ADSM has a volume mounted, the drive is allocated to ADSM and cannot be used for anything else. If you need to free the drive for other uses, you can cancel ADSM operation that are using the drive and then dismount the volume. For example, you can cancel server migration or backup operations. For information on how to cancel processes and dismount volumes, see “Canceling Server Processes” on page 253 and “Dismounting an Idle Volume” on page 82.

Tape Label Prefix

By using the *PREFIX* parameter, you can specify a prefix value that is used to construct the *file name* string that is stored in the label area of each tape volume.

The prefix string is used as the prefix of the file name that is written to the label of each tape.

The default value for the tape label prefix string is *ADSM*.

Recording Format

You can use the *FORMAT* parameter to specify the recording format used by ADSM when writing data to a tape. The following tables shows the values supported for the tape device classes:

Device Class	**Reference**
4MM	Table 3 on page 96
8MM	Table 4 on page 96
DLT	Table 5 on page 97
3570	Table 6 on page 97
3590	Table 7 on page 98
CARTRIDGE	Table 8 on page 98

Use the FORMAT=DRIVE parameter only if all drives that can be accessed by the device class are identical. If some drives associated with a device class support a higher density format than others, mount errors can occur when you specify FORMAT=DRIVE.

For example, suppose a device class uses two incompatible devices such as an IBM 7208-2 and a 7208-12. The server might select the high-density recording format of 8500 for each of two new volumes. Later, if the two volumes are to be mounted concurrently, one fails because only one of the drives is capable of the high-density recording format.

The recording format that ADSM uses for a given tape volume is selected when the first piece of data is written to the volume. Note that updating the FORMAT parameter of a device class does not affect tapes that already contain data until those tapes are rewritten from the beginning. This process may happen after a volume is reclaimed or deleted, or after all of the data on the volume expires.

4MM Device Classes

Table 3. Recording Format and Default Estimated Capacity for 4mm Tape Volumes

Recording Format	Estimated Capacity	Description
DDS1	1.3GB (60-meter) 2.0GB (90-meter)	Specifies that ADSM writes data using the DDS-1 recording format, without compression. This format only applies to 60- and 90-meter tapes.

Table 3. Recording Format and Default Estimated Capacity for 4mm Tape Volumes

Recording Format	Estimated Capacity	Description
DDS1C	1.3GB (60-meter) 2.0GB (90-meter)	Specifies that ADSM writes data using the DDS-1 recording format, with compression enabled. This format uses the tape drive's hardware compression feature, and the actual capacity may be greater, depending on the effectiveness of compression.
DDS2	4.0GB	Specifies that ADSM writes data using the DDS-2 recording format, without compression. This format only applies to 120-meter tapes.
DDS2C	8.0GB	Specifies that ADSM writes data using the DDS-2 recording format, with compression. This format uses the tape drive's hardware compression feature, and the actual capacity may be greater, depending on the effectiveness of compression. This format only applies to 120-meter tapes.
DRIVE	—	Lets the server select the recording format to use based on the drive on which the volume is mounted.

8MM Device Classes

Table 4. Recording Format and Default Estimated Capacity for 8mm Tape Volumes

Recording Format	Estimated Capacity	Description
8200	2.3GB	Basic recording format for a standard 112-meter tape cartridge.
8200C	2.3GB	Specifies that ADSM writes data using the 8200C recording format for standard 112-meter tape cartridges. However, this format uses the tape drive's hardware compression feature, and the actual capacity may be greater than 2.3GB, depending on the effectiveness of the compression feature.
8500	5.0GB	Enhanced recording format for a standard 112-meter tape cartridge.
8500C	5.0GB	Specifies that ADSM writes data using the 8500C recording format for standard 112-meter tape cartridges. However, this format uses the tape drive's hardware compression feature, and the actual capacity may be greater than 5.0GB, depending on the effectiveness of the compression feature.
DRIVE	—	Lets the server select the recording format to use based on the drive on which the volume is mounted.

DLT Device Classes

Table 5. Recording Format and Default Estimated Capacity for DLT Tape Volumes

Recording Format	Estimated Capacity	Description
DLT10	10GB	Specifies that ADSM writes data using the DLT10 recording format on CompacTape III or CompacTape IV tape cartridges.
DLT10C	10GB	Specifies that ADSM writes data using the DLT10C recording format on CompacTape III or CompacTape IV tape cartridges. However, this format uses the tape drive's hardware compression feature, and the actual capacity may be greater than 10GB, depending on the effectiveness of the compression feature.
DLT15	15GB	Specifies that ADSM writes data using the DLT15 recording format on CompacTape III XT tape cartridges. **Attention:** Use this format only with CompacTape III XT cartridges in a DLT2000 or DLT4000 tape drive.
DLT15C	15GB	Specifies that ADSM writes data using the DLT15 recording format on CompacTape III XT tape cartridges. However, this format uses the tape drive's hardware compression feature, and the actual capacity may be greater than 15GB, depending on the effectiveness of the compression feature. **Attention:** Use this format only with CompacTape III XT cartridges in a DLT2000 or DLT4000 tape drive.
DLT20	20GB	Specifies that ADSM writes data using the DLT20 recording format on CompacTape IV tape cartridges. **Attention:** Use this format only with CompacTape IV cartridges in a DLT4000 drive.
DLT20C	20GB	Specifies that ADSM writes data using the DLT20 recording format on CompacTape IV tape cartridges. However, this format uses the tape drive's hardware compression feature, and the actual capacity may be greater than 10GB, depending on the effectiveness of the compression feature. **Attention:** Use this format only with CompacTape IV cartridges in a DLT4000 drive.
DRIVE	—	Lets the server select the recording format to use based on the drive on which the volume is mounted.

3570 Device Classes

Table 6 (Page 1 of 2). Recording Format and Default Estimated Capacity for 3570 Tape Volumes

Recording Format	Estimated Capacity	Description
3570B	5GB	Specifies that ADSM writes data using the basic (uncompressed) recording format.

Table 6 (Page 2 of 2). Recording Format and Default Estimated Capacity for 3570 Tape Volumes

Recording Format	Estimated Capacity	Description
3570C	5GB	Specifies that ADSM writes data using the compressed recording format. This format uses the tape drive's hardware compression feature, which may allow for greater capacity depending on the effectiveness of the compression.
DRIVE	—	Lets the server select the highest recording format to be used based on the drive on which the volume is mounted.

3590 Device Classes

Table 7. Recording Format and Default Estimated Capacity for 3590 Tape Volumes

Recording Format	Estimated Capacity	Description
3590B	10GB	Specifies that ADSM writes data using the basic (uncompressed) recording format.
3590C	10GB	Specifies that ADSM writes data using the compressed recording format. This format uses the tape drive's hardware compression feature, which may allow for greater capacity depending on the effectiveness of the compression.
DRIVE	—	Lets the server select the highest recording format to be used based on the drive on which the volume is mounted.

CARTRIDGE Device Classes

Table 8. Recording Format and Default Estimated Capacity for Cartridge Tape Volumes

Recording Format	Estimated Capacity	Description
3480	180MB	18-track basic recording format
3480XF	180MB	18-track compacted recording format
3490B	360MB	36-track basic recording format
3490C	360MB	36-track compacted recording format
DRIVE	—	Lets the server select the recording format to be used based on the drive on which the volume is mounted.

Estimated Capacity Value

ADSM estimates the capacity of the volumes in a storage pool based on the parameters assigned to the device class that is associated with the storage pool. The estimated capacity value is also used by ADSM when making decisions about when to initiate a reclamation process for volumes in the storage pool. For more information on how ADSM uses the estimated capacity value, see “How ADSM Fills Volumes” on page 104.

You can either accept the default estimated capacity value for a given device class or explicitly specify an estimated capacity that you want the server to use instead of the default.

For tape device classes, the default values selected by the server depend on the recording format used to write data to the volume. These values are listed by device class in the following tables:

Device Class	**Reference**
4MM	Table 3 on page 96
8MM	Table 4 on page 96
DLT	Table 5 on page 97
3570	Table 6 on page 97
3590	Table 7 on page 98
CARTRIDGE	Table 8 on page 98

Library

Before the server can mount a volume, it must know which drives can be used to satisfy the mount request. This process is done by specifying the library when the device class is defined. The library must contain drives that can be used to mount the volume.

Note that only one library can be associated with a given device class. However, multiple device classes can reference the same library. In this case, you must ensure that the sum of the mount limit values for each such device class does not exceed the number of drives defined in the referenced library.

There is no default value for this parameter. It is required, and so must be specified when the device class is defined.

Defining and Updating Device Classes for Optical Devices

To use optical media, you must define a device class by issuing the DEFINE DEVCLASS command with a DEVTYPE parameter for one of the optical devices:

Parameter	**Description**
OPTICAL	5.25-inch rewritable optical media
WORM	5.25-inch write-once optical media
WORM12	12-inch write-once optical media
WORM14	14-inch write-once optical media

Other parameters specify how to manage data storage operations involving the new device class:

- MOUNTLIMIT
- MOUNTWAIT
- MOUNTRETENTION
- FORMAT
- ESTCAPACITY
- LIBRARY

You can update the device class information by issuing the UPDATE DEVCLASS command.

Library

The LIBRARY parameter for OPTICAL, WORM, WORM12, and WORM14 device classes is used as described in “Library” on page 99.

Mount Limit

The mount limit value for OPTICAL, WORM, WORM12, and WORM14 device classes is used as described in “Mount Limit” on page 93.

Mount Wait Period

The mount wait period for OPTICAL, WORM, WORM12, and WORM14 device classes is used as described in “Mount Wait Period” on page 94.

Mount Retention Period

The mount retention period for OPTICAL, WORM, WORM12, and WORM14 device classes is used as described in “Mount Retention Period” on page 94.

Estimated Capacity Value

ADSM estimates the capacity of the volumes in a storage pool based on the parameters assigned to the device class that is associated with the storage pool. The estimated capacity value is used by ADSM when making decisions about when to initiate a reclamation process for volumes in the storage pool. It is also used to generate storage pool and volume reports. As a volume is written and filled, the server determines the actual capacity of the volume and uses this instead of the estimated capacity.

You can either accept the default estimated capacity value for a given device class or explicitly specify an estimated capacity that you want the server to use instead of the default.

Recording Format

You can use the *FORMAT* parameter to specify the recording format used by ADSM when writing data to an optical cartridge. The following values are supported for OPTICAL, WORM, WORM12, and WORM14 device classes (Table 9):

OPTICAL Device Classes

Table 9 (Page 1 of 2). Recording Format for OPTICAL, WORM, WORM12, and WORM14 Drives

Recording Format	Description
650MB	Used with optical cartridges whose capacity is 650MB. **Note:** This value is not valid for WORM12 device types.

Table 9 (Page 2 of 2). Recording Format for OPTICAL, WORM, WORM12, and WORM14 Drives

Recording Format	**Description**
1300MB	Used with optical cartridges whose capacity is 1300MB. **Note:** This value is not valid for WORM12 device types.
2600MB	Used with optical cartridges whose capacity is 2600MB. **Note:** This value is not valid for WORM12 device types.
5600MB	Used only with WORM12 optical cartridges whose capacity is 5600MB.
10200MB	Used only with WORM14 optical cartridges whose capacity is 10200MB.
12000MB	Used only with WORM12 optical cartridges whose capacity is 12000MB.
14800MB	Used only with WORM14 optical cartridges whose capacity is 14800MB.
DRIVE	Allows the server to select the recording format to be used based on the capabilities of the drive used to access the optical disk.

Defining and Updating FILE Device Classes

The FILE device type is used for special device classes whose storage volumes are not physical units, such as tape or optical cartridges, but *simulated* storage volumes. Data is written sequentially into standard files in the file system of the server machine. You can define this device class by issuing a DEFINE DEVCLASS command with the DEVTYPE=FILE parameter.

Note: Do not use raw partitions with a device class type of FILE.

Because each volume in a FILE device class is actually a file, a volume name is a fully qualified file name string.

When you define the FILE device class, you can supply the following parameters to manage server storage operations for the new device class:

- MOUNTLIMIT
- MAXCAPACITY
- DIRECTORY

You can update the device class information by issuing the UPDATE DEVCLASS command.

Mount Limit

The mount limit value for FILE device classes is used to restrict the number of volumes (that is, files) that can be concurrently opened for access by server storage and retrieval operations. Any attempts to access more volumes than indicated by the mount limit causes the requester to wait.

For how to determine an appropriate mount limit value for the new device class, see “Mount Limit” on page 93.

Maximum Capacity Value

You can specify a maximum capacity value that restricts the size of volumes (that is, files) associated with a FILE device class. Use the MAXCAPACITY parameter of the DEFINE DEVCLASS command. When the server detects that a volume has reached a size equal to the maximum capacity, it treats the volume as full and stores any new data on a different volume.

The default MAXCAPACITY value for a FILE device class is 4MB.

Directory

You can specify the directory location of the files used in the FILES device class. The default is the current working directory of the server at the time the command is issued.

The directory name identifies the location where the server places the files that represent storage volumes for this device class. While processing the command, the server expands the specified directory name into its fully qualified form, starting from the root directory.

Later, if the server needs to allocate a scratch volume, it creates a new file in this directory. The following lists the file name extension created by the server for scratch volumes depending on the type of data that is stored.

For scratch volumes used to store this data:	The file extension is:
Client data	.BFS
Export	.EXP
Database backup	.DBB
Database dump	.DMP

Requesting Information about a Device Class

You can choose to view a standard or the default detailed report for a device class.

Task	Required Privilege Class
Request information about device classes	Any administrator

To display a standard report on device classes, enter:

```
query devclass
```

Figure 14 on page 103 is an example of a standard report for device classes.

```
Device       Device        Storage    Device     Format     Est/Max    Mount
Class        Access           Pool    Type                 Capacity    Limit
Name         Strategy        Count                             (MB)
---------    ----------    -------    -------    ------    --------    -----
DISK         Random              9
TAPE8MM      Sequential          1    8MM        8200       2,472.0        2
```

Figure 14. Example of a Standard Device Class Report

To view a detailed report for the TAPE8MM device class, enter:

```
query devclass tape8mm format=detailed
```

Figure 15 shows an example of a detailed report for a device class.

```
             Device Class Name: TAPE8MM
        Device Access Strategy: Sequential
            Storage Pool Count: 1
                   Device Type: 8MM
                        Format: 8200
         Est/Max Capacity (MB): 2,472.0
                   Mount Limit: 2
              Mount Wait (min): 10
         Mount Retention (min): 30
                  Label Prefix: ADSM
                       Library: TAPELIB
                     Directory:
Last Update by (administrator): ADSMADMIN
         Last Update Date/Time: 01/05/1996 16:02:13
```

Figure 15. Example of a Detailed Device Class Report

Deleting a Device Class

Task	Required Privilege Class
Delete a device classes	System or unrestricted storage

You can delete a device class with the DELETE DEVCLASS command when:

- No storage pools are assigned to the device class. For information on deleting storage pools, see “Deleting a Storage Pool” on page 154.
- The device class is not being used by an export or import process.

Note: You cannot delete the DISK device class from the server.

How ADSM Fills Volumes

The DEFINE DEVCLASS command has an optional ESTCAPACITY parameter that indicates the estimated capacity for sequential volumes associated with the device class. If the ESTCAPACITY parameter is not specified, ADSM uses a default value based on the DEVTYPE parameter of the device class.

If you specify an estimated capacity that exceeds the actual capacity of the volume in the device class, ADSM updates the estimated capacity of the volume when the volume becomes full. When ADSM reaches the end of the volume, it updates the capacity for the amount that is written to the volume.

You can either accept the default estimated capacity for a given device class, or explicitly specify an estimated capacity. An accurate estimated capacity value is not required, but is useful. ADSM uses the estimated capacity of volumes to determine the estimated capacity of a storage pool, and the estimated percent utilized. You may want to change the estimated capacity if:

- The default estimated capacity is inaccurate because data compression is being performed by the drives
- You have volumes of nonstandard size

Using Data Compression

Client files can be compressed to decrease the amount of data sent over networks and the space occupied by the data in ADSM storage. With ADSM, files can be compressed by the ADSM client before the data is sent to the ADSM server, or by the device where the file is finally stored.

Use either client compression or device compression, but not both. The following table summarizes the advantages and disadvantages of each type of compression.

Type of Compression	Advantages	Disadvantages
ADSM client compression	Reduced load on the network	Higher CPU usage by the client Longer elapsed time for client operations such as backup
Drive compression	Amount of compression can be better than ADSM client compression on some drives	Files that have already been compressed by the ADSM client can become larger

Either type of compression can affect tape drive performance, because compression affects data rate. When the rate of data going to a tape drive is slower than the drive can write, the drive starts and stops while data is written, meaning relatively poorer performance. When the rate of data is fast enough, the tape drive can reach streaming mode, meaning better performance. If tape drive performance is more important than the space savings that compression can mean, you may want to perform timed test backups using different approaches to determine what is best for your system.

For how to set up compression on the client, see “User Registration of Client Nodes” on page 293 and “Administrator Registration of Client Nodes” on page 294.

Tape Volume Capacity and Data Compression

How ADSM views the capacity of the volume where the data is stored depends on whether files are compressed by the ADSM client or by the storage device. It may wrongly appear that you are not getting the full use of the capacity of your tapes, for the following reasons:

- A tape device manufacturer often reports the capacity of a tape based on an assumption of compression by the device. If a client compresses a file before it is sent, however, the device may not be able to compress it any further before storing it.
- ADSM records the size of a file as it goes to a storage pool. If the client compresses the file, ADSM records this smaller size in the database. If the drive compresses the file, ADSM is not aware of this compression.

Figure 16 on page 106 compares what ADSM sees as the amount of data stored on tape when compression is done by the device and by the client. For this example, the tape has a physical capacity of 1.2GB; however, the manufacturer reports the capacity of the tape as 2.4GB by assuming the device compresses the data by a factor of two.

Suppose a client backs up a 2.4GB file:

- When the client does *not* compress the file, the server records the file size as 2.4GB, the file is compressed by the drive to 1.2GB, and the file fills up one tape.
- When the client compresses the file, the server records the file size as 1.2GB, the file cannot be compressed any further by the drive, and the file still fills one tape.

In both cases, ADSM considers the volume to be full. However, ADSM considers the capacity of the volume in the two cases to be different: 2.4GB when the drive compresses the file, and 1.2GB when the client compresses the file. Use the QUERY VOLUME command to see the capacity of volumes from ADSM’s viewpoint. See “Monitoring the Use of Storage Pool Volumes” on page 165.

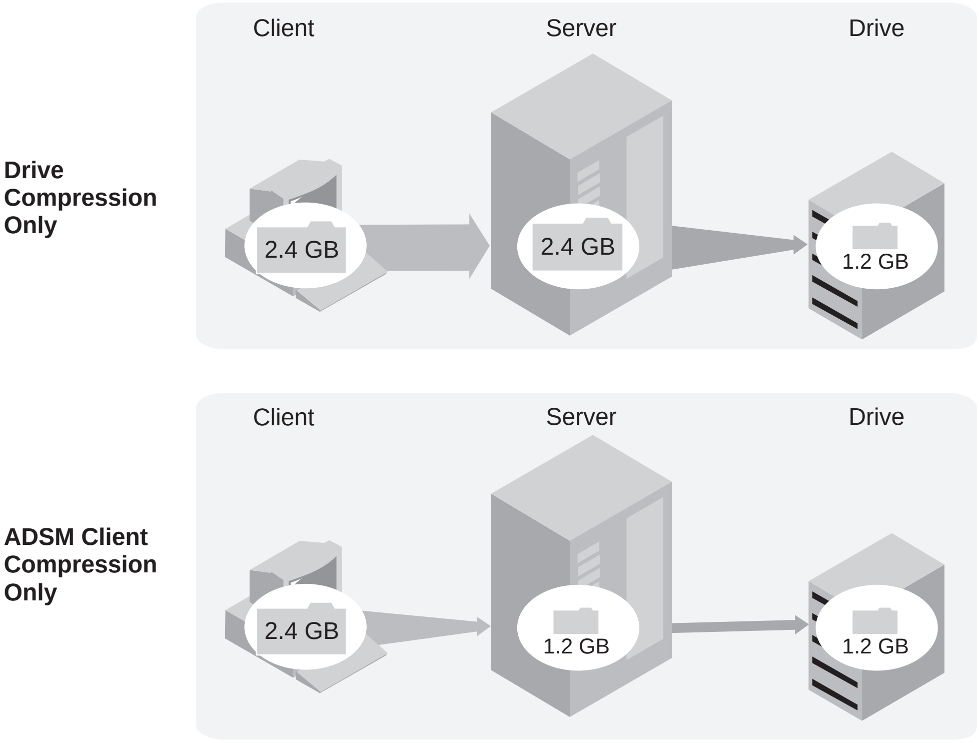


Figure 16. Comparing Compression at the Client and Compression at the Device

For how to set up compression on the client, see “User Registration of Client Nodes” on page 293 and “Administrator Registration of Client Nodes” on page 294.

Chapter 8. Managing Storage Pools

A storage pool is a collection of storage volumes belonging to the same device class. The storage volumes contain backed up, archived, or space-managed files. The group of storage pools you set up for ADSM to use is called ADSM's *server storage*.

The sections listed in the following table begin at the indicated pages.

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, "Interface Cross-Reference" on page 467 lists each command and shows if its function is also available on the administrative client GUI.

© Copyright IBM Corp. 1995, 1997

Storage Pools

ADSM has two types of storage pools:

Primary storage pool

When a client node backs up, archives, or migrates data, the data is stored in a primary storage pool.

When a user tries to restore, retrieve, recall, or export file data, the requested file is obtained from a primary storage pool if possible. Primary storage pool volumes are always located onsite.

A primary storage pool can use random access storage (DISK device class) or sequential access storage (for example, tape or FILE device classes).

ADSM has three default, random access, primary storage pools:

ARCHIVEPOOL	Contains files archived from client nodes
BACKUPPOOL	Contains files backed up from client nodes
SPACEMGPOOL	Contains files migrated from client nodes via the space management function (space-managed files)

ADSM does not require a separate storage pool for space-managed files, but a separate storage pool is recommended. Clients are likely to require fast access to their space-managed files, and therefore you may want to have those files stored in a separate storage pool that uses your fastest disk storage.

Copy storage pool

When an administrator backs up a primary storage pool, the data is stored in a copy storage pool. See “Backing Up Storage Pools” on page 142 for details.

The copy storage pool provides a means of recovering from disasters or media failures. For example, when a client attempts to retrieve a file and the server detects a data-integrity error in the file copy in the primary storage pool, the server marks the file as damaged. At the next attempt to access the file, the server obtains the file from a copy storage pool.

ADSM attempts to access the file from a copy storage pool if the primary copy of the file cannot be obtained for one of the following reasons:

- The primary file copy has been previously marked damaged (for information about damaged files, see “Correcting Damaged Files” on page 350)
- The primary file is stored on a volume that is UNAVAILABLE or DESTROYED
- The primary file is stored on an offline volume
- The primary file is located in a storage pool that is UNAVAILABLE, and the operation is for restore, retrieve, or recall of files to a user, or export of file data

For details, see “Restoring Storage Pools” on page 154, “Using Copy Storage Pools to Improve Data Availability” on page 143, “Recovering a Lost or Damaged Storage Pool Volume” on page 356, and “Maintaining the Integrity of Files” on page 350.

A copy storage pool can use only sequential access storage (for example, a tape or FILE device class).

Copy storage pool volumes can be moved offsite and still be tracked by ADSM. Moving copy storage pool volumes offsite provides a means of recovering from an onsite disaster.

An Example of Server Storage

Figure 17 shows one way to set up ADSM server storage. In this example, the storage defined for the server includes:

- The three default disk storage pools, all primary storage pools
- One primary storage pool consisting of tape cartridges
- One copy storage pool consisting of tape cartridges

For each of the three disk storage pools, the tape primary storage pool is next in the hierarchy. For more information about setting up a storage hierarchy, see “Storage Pool Hierarchy” on page 111.

All four of the primary storage pools can be backed up to the one copy storage pool. For more information on backing up primary storage pools, see “Backing Up Storage Pools” on page 142.

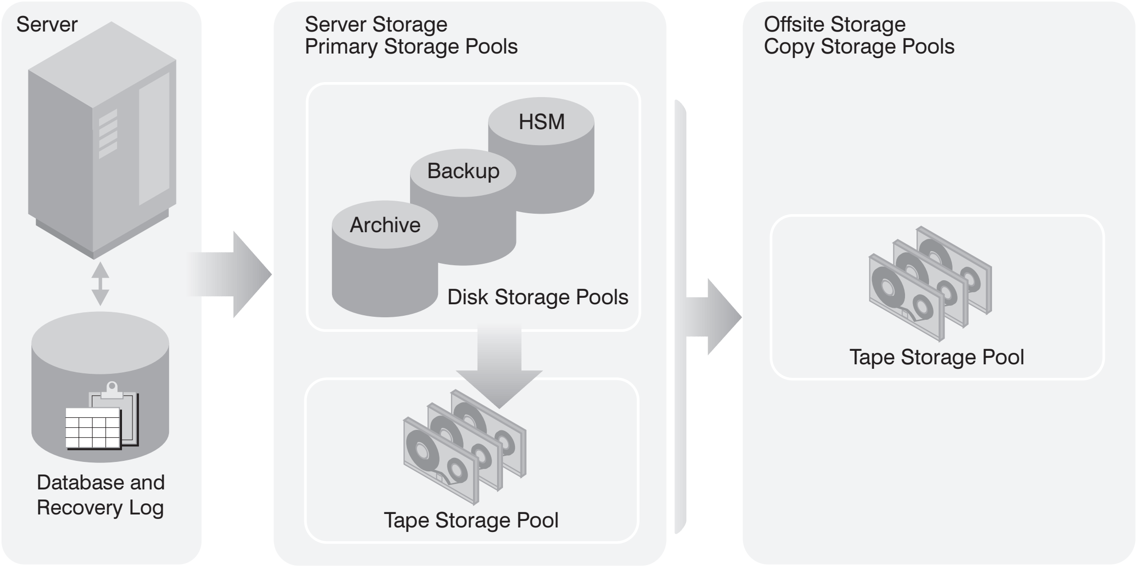


Figure 17. Example of Server Storage

Comparing Primary and Copy Storage Pools

Table 10 compares the characteristics of primary and copy storage pools.

Table 10. Comparing Primary and Copy Storage Pools

Characteristic	Primary storage pool	Copy storage pool
Destination for backed up or archived files (specified in backup or archive copy groups)	Yes	No
Destination for space-managed files (specified in the management class)	Yes	No
Offsite access mode for volumes	No	Yes
Destroyed access mode for volumes	Yes	No
Random access storage volumes	Yes	No
Sequential access storage volumes	Yes	Yes
Contents	Client files (backup versions, archived files, space-managed files)	Copies of files that are stored in primary storage pools
Moving data allowed	Within the same primary storage pool, or to any primary storage pool	Within the same pool only. If volumes are offsite, data is copied from the original files in primary storage pools.
Collocation	Yes (sequential access storage pools only)	Yes
Reclamation	Yes (sequential access storage pools only)	Yes Offsite volumes are handled differently. For details, see "Reclamation of Offsite Volumes" on page 129.
File deletion	Files are deleted: • During inventory expiration processing, if the files have expired • When a file space is deleted • When a volume is deleted with the option to discard the data • When a primary storage pool volume is audited with the FIX=YES option, if the files on the volume are damaged and no other copies of the file exist	Files are deleted: • Whenever the primary copy of the file is deleted from the primary storage pool (because of expiration, file space deletion, or volume deletion) • When a volume is deleted with the option to discard the data • When a copy storage pool volume is audited with the FIX=YES option, if the files on the volume are damaged

Assigning Volumes to Storage Pools

Before a storage pool can be used to store data, volumes must be assigned to the pool. Volumes are assigned differently depending on whether the pool is a random access storage pool or a sequential access storage pool.

Assigning Random Access Storage Pool Volumes

Volumes in random access storage pools must be prepared for use (formatted) and then defined. See Chapter 9, “Managing Storage Pool Volumes” on page 159 for information about formatting and defining volumes.

Assigning Sequential Access Storage Pool Volumes

You can define volumes in a sequential access storage pool or you can specify that ADSM dynamically acquire scratch volumes. You can also use a combination of defined and scratch volumes.

Use defined volumes when you want to control precisely which volumes are used in the storage pool. Using defined volumes may be useful when you want to establish a volume naming scheme for ADSM volumes. See Chapter 9, “Managing Storage Pool Volumes” on page 159 for information about defining volumes.

Use scratch volumes when you want to allow ADSM to dynamically acquire a volume when needed and dynamically delete the volume when it becomes empty. For example, you might want to use scratch volumes to avoid the burden of explicitly defining all of the volumes in a given storage pool.

ADSM tracks whether a volume being used was originally a scratch volume. Scratch volumes that ADSM acquired for a primary storage pool are deleted from the ADSM database when they become empty. The volumes are then available for reuse by ADSM or other applications. For scratch volumes that were acquired in a FILE device class, the space that the volumes occupied is freed by ADSM and returned to the file system.

Scratch volumes in a copy storage pool are handled in the same way as scratch volumes in a primary storage pool, except for volumes with the access value of offsite. If an offsite volume becomes empty, it is not immediately returned to the scratch pool. The delay prevents the empty volumes from being deleted from the database and makes it easier to determine which volumes should be returned to the onsite location. The administrator can query ADSM for empty offsite copy storage pool volumes and return them to the onsite location. The volume is returned to the scratch pool only when the access value is changed to READWRITE, READONLY, or UNAVAILABLE.

Storage Pool Hierarchy

Consider using multiple levels of primary storage pools to form a storage hierarchy. For example, assume that your fastest devices are disks, but space on these devices is scarce. You also have tape drives, which are slower to access, but have much greater capacity. You can define a hierarchy so that files are initially stored on the fast disk volumes in one storage pool, to provide clients with quick response to backup and recall requests. Then, as the disk storage pool becomes full, ADSM migrates, or moves, data to tape volumes in a different storage pool. Migrating files to sequential storage pool volumes is particularly useful because all the files for a node are migrated together and organized in a more orderly way. This is especially helpful if collocation is not enabled.

When defining or updating a storage pool, you establish a hierarchy by identifying the storage pool to which data will be migrated, or moved, if the original storage pool is full or otherwise unavailable.

Restrictions:

1. You cannot establish a chain of storage pools that leads to an endless loop. For example, you cannot define StorageB as the *next* storage pool for StorageA, and then define StorageA as the *next* storage pool for StorageB.
2. The storage pool hierarchy includes only primary storage pools, not copy storage pools.

How ADSM Stores Files in a Storage Pool Hierarchy

Understanding how the server selects and accesses a primary storage pool can help you estimate the amount of space required for each storage pool in the hierarchy.

When a user backs up, archives, or migrates a file from a client node to the server, the server looks at the management class that is bound to the file to determine in which storage pool to store the file. The server then checks the storage pool to determine the following:

- If it is possible to write file data to the storage pool (access mode)
- What maximum file size is allowed in the storage pool
- Whether sufficient space is available on the available volumes in the storage pool
- What the next storage pool used is, if any of the previous conditions prevent the file from being stored in the storage pool being checked

Based on these factors, the server determines if the file can be written to that storage pool or the next storage pool in the hierarchy. As an example of how this might work, assume a company has a storage pool hierarchy as shown in Figure 18 on page 113.

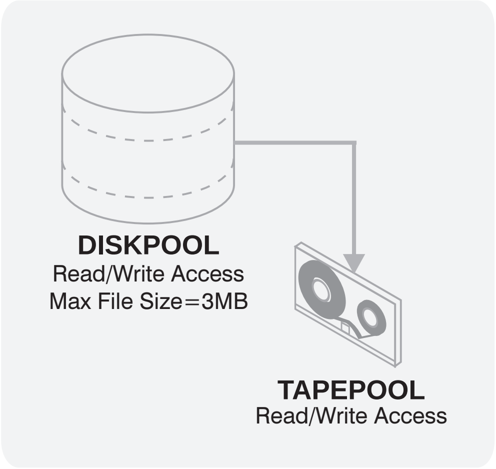


Figure 18. Storage Hierarchy, Read/Write Access, and Maximum File Size

The storage pool hierarchy consists of two storage pools:

DISKPOOL The top of the storage hierarchy. It contains fast disk volumes for storing data.

TAPEPOOL The next storage pool in the hierarchy. It contains tape volumes accessed by high-performance tape drives.

Assume a user wants to archive a 5MB file named *FileX*. FileX is bound to a management class that contains an archive copy group whose storage destination is DISKPOOL, see Figure 18.

When the user archives the file, the server determines where to store the file based on the following process:

1. The server selects DISKPOOL because it is the storage destination specified in the archive copy group.
2. Because the access mode for DISKPOOL is read/write, the server checks the maximum file size allowed in the storage pool.
3. The maximum file size allowed in DISKPOOL is 3MB. FileX is a 5MB file and therefore cannot be stored in DISKPOOL. The server searches for the next storage pool in the storage hierarchy.
4. The server checks the access mode of TAPEPOOL, which is the next storage pool in the storage hierarchy.
5. The access mode for TAPEPOOL is read/write. The server then checks the maximum file size allowed in the storage pool.
6. Because TAPEPOOL is the last storage pool in the storage hierarchy, no maximum file size is specified. Therefore, if there is available space in TAPEPOOL, FileX can be stored in it.

How the Storage Hierarchy Affects Planning for Copy Storage Pools

It is strongly recommended that all primary storage pools that are linked to form a storage hierarchy use the same copy pool for backup. If this is done, then a file that is copied does not need to be recopied when it migrates to another primary storage pool.

For most cases, a single copy storage pool can be used for backup of all primary storage pools. The number of copy storage pools you need depends on the hierarchies you have set up with your primary storage pools and what type of disaster recovery protection you wish to implement.

Multiple copy storage pools may be needed to handle particular situations, including:

- Special processing of certain primary storage hierarchies (for example, archive storage pools or storage pools dedicated to priority clients)
- Creation of multiple copies for multiple locations (for example, to keep one copy onsite and one copy offsite)
- Rotation of full storage pool backups (See “Backing Up Storage Pools” on page 332 for more information.)

Using the Hierarchy to Stage Client Data from Disk to Tape

A common way to use the storage hierarchy is for initially storing client data on disk, then letting ADSM migrate the data to tape. A guideline for how much primary disk storage should be dedicated for this staging of client data is enough storage to handle one night’s worth of the clients’ incremental backups. While not always feasible, this guideline has even more value when considering storage pool backups.

For example, if you have enough disk space for nightly incremental backups for clients and have tape devices, you can set up the following pools:

- A primary storage pool on disk, with enough volumes assigned to contain the nightly incremental backups for clients
- A primary storage pool on tape, which is identified as the next storage pool in the hierarchy for the disk storage pool
- A copy storage pool on tape

Then you can schedule these steps every night:

1 Perform an incremental backup of the clients to the disk storage pool.

2 After clients complete their backups, back up the disk primary storage pool (now containing the incremental backups) to the copy storage pool.

Backing up disk storage pools before migration processing allows you to copy as many files as possible while they are still on disk. This saves mount requests while performing your storage pool backups.

3 Start the migration of the files in the disk primary storage pool to the tape primary storage pool (the next pool in the hierarchy) by lowering the high migration threshold. For example, lower the threshold to 40%.

When this migration completes, raise the high migration threshold back to 100%.

4 Back up the tape primary storage pool to the copy storage pool to ensure that all files have been backed up.

The tape primary storage pool must still be backed up to catch any files that might have been missed in the backup of the disk storage pools (for example, large files that went directly to sequential media).

See “Estimating Space Needs for Storage Pools” on page 135 for more information about storage pool space.

Server Migration of Files

ADSM provides automatic migration to maintain free space in a primary storage pool. For example, ADSM can migrate data stored on a random access disk storage pool to a less expensive sequential access storage pool when the migration threshold parameter you set is exceeded.

Migration Thresholds for Disk Storage Pools

When you define or update a storage pool, set migration thresholds to specify when the server should begin migrating, or moving, data to the next storage pool in the storage hierarchy. This process helps to ensure that there is sufficient free space in the storage pools at the top of the hierarchy, where faster devices can provide the most benefit to clients.

You can use the defaults for the migration thresholds, or you can change the threshold values to identify the maximum and minimum amount of space for a storage pool. See “Defining a Primary Storage Pool” on page 137 for more information about migration thresholds.

Before you define migration thresholds, you should understand how the server determines when to migrate files, and how it chooses which files to migrate. Then you can determine migration thresholds for both disk and sequential access storage pools.

For disk storage pools, migration thresholds can be set lower when cache is enabled. See “The Use of Cache on Disk Storage Pools” on page 120 for information about setting the CACHE parameter.

When Files Are Migrated

When the high migration threshold is reached in a storage pool, ADSM migrates files from the pool to the next storage pool. ADSM first identifies which client node has backed up or migrated the largest single file space or has archived files that occupy the most space. When the server identifies the client node based on these criteria, the server migrates *all* files from *every* file space belonging to that client.

After the files for the first client node are migrated to the next storage pool, the server checks the low migration threshold for the storage pool to determine if the migration process should be stopped. If the amount of space used in the storage pool is now below the low migration threshold, migration ends. If not, another client node is chosen by using the same criteria as described above, and the migration process continues.

For example, Table 11 displays information contained in the database that is used by the server to determine which files to migrate. This example assumes no space-managed files are stored in the storage pool.

Table 11. Database Information on Files Stored in DISKPOOL

Client Node	Backed-Up File Spaces	Archived Files (All Client File Spaces)
TOMC	TOMC/C = 200MB TOMC/D = 100MB	55MB
HTANG	HTANG = 50MB	5MB
PEASE	PEASE/home = 150MB PEASE/temp = 175MB	40MB

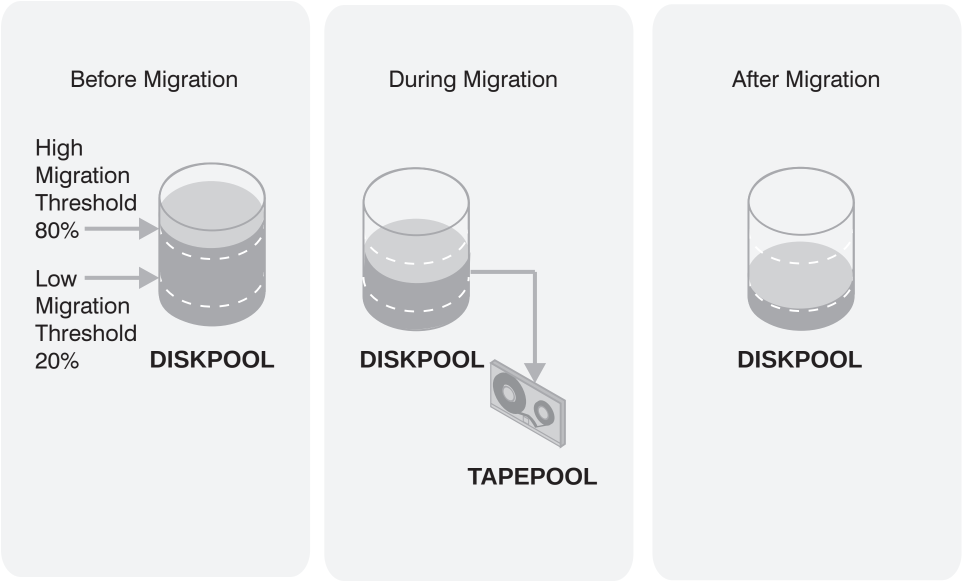


Figure 19. The Migration Process and Migration Thresholds

Figure 19 shows what happens when the high migration threshold defined for the disk storage pool *DISKPOOL* is exceeded. When the amount of migratable data in DISKPOOL reaches 80%, the server performs the following tasks:

1. Determines that the TOMC/C file space is taking up the most space in the DISKPOOL storage pool, more than any other single backed-up or space-managed file space and more than any client node's archived files.
2. Locates all data belonging to node TOMC stored in DISKPOOL. In this example, node TOMC has backed up or archived files from file spaces TOMC/C and TOMC/D stored in the DISKPOOL storage pool.
3. Migrates all data from TOMC/C and TOMC/D to the next available storage pool. In this example, the data is migrated to the tape storage pool, TAPEPOOL.

   The server migrates all of the data from both file spaces belonging to node TOMC, even if the occupancy of the storage pool drops below the low migration threshold before the second file space has been migrated.

   If the cache option is enabled, files that are migrated remain on disk storage (that is, the files are *cached*) until space is needed for new files. For more information about using cache, see "The Use of Cache on Disk Storage Pools" on page 120.
4. After all files that belong to TOMC are migrated to the next storage pool, the server checks the low migration threshold. If the low migration threshold has not been reached, then the server again determines which client node has backed up or migrated the largest single file space or has archived files that occupy the most space. The server begins migrating files belonging to that node.

   In this example, the server migrates *all* files that belong to the client node named PEASE to the TAPEPOOL storage pool.
5. After all the files that belong to PEASE are migrated to the next storage pool, the server checks the low migration threshold again. If the low migration threshold has been reached or passed, then migration ends.

Appropriate Migration Threshold Values

Setting migration thresholds for disk storage pools ensures sufficient free space on faster speed devices, which can lead to better ADSM performance. Choosing thresholds appropriate for your situation takes some experimenting, and you can start by using the default values. You need to ensure that migration occurs frequently enough to maintain some free space but not so frequently that the device is unavailable for other use.

To choose the high-migration threshold, consider:

- The amount of storage capacity provided for each storage pool
- The amount of free storage needed for users to store additional files, without having migration occur

If you set the high-migration threshold too high, the pool may be just under the high threshold, but not have enough space to store an additional, typical client file. Or, with a high threshold of 100%, the pool may become full and a migration process must start before clients can back up any additional data to the disk storage pool. In either case, client files must be stored directly to tape until migration completes, resulting in slower performance.

If you set the high-migration threshold too low, migration runs more frequently and can interfere with other operations.

Keeping the high-migration threshold at a single value means that migration processing could start at any time of day, whenever that threshold is exceeded. You can control when migration occurs by using administrative command schedules to change the threshold. For example, set the high-migration threshold to 95% during the night when clients run their backup operations, then lower the high-migration threshold to 50% during the time of day when you want migration to occur. By scheduling when migration occurs, you can choose a time when your tape drives and mount operators are available for the operation.

To choose the low-migration threshold, consider:

- The amount of free disk storage space needed for normal daily processing. If you have disk space to spare, you can keep more data on the disk (a larger low threshold). If clients' daily backups are enough to fill the disk space every day, you may need to empty the disk (a smaller low threshold).

  If your disk space is limited, try setting the threshold so that migration frees enough space for the pool to handle the amount of client data that is typically stored every day. Migration then runs about every day, or you can force it to run every day by lowering the high-migration threshold at a time you choose.

- Whether you use cache on disk storage pools to improve the retrievability of data. If you use cache, you can set the low threshold lower, yet still maintain faster retrieval for some data. Migrated data remains cached on the disk until new client data pushes the data off the disk. Using cache requires more disk space for the database, however.

  If you do not use cache, you may want to keep the low threshold at a higher number so that more data stays on the disk.

- How frequently you want migration to occur, based on the availability of sequential access storage devices and mount operators. The larger the low threshold, the shorter time that a migration process runs (because there is less data to migrate). But if the pool refills quickly, then migration occurs more frequently. The smaller the low threshold, the longer time that a migration process runs, but the process runs less frequently.

  You may need to balance the costs of larger disk storage pools with the costs of running migration (drives, tapes, and either operators or automated libraries).

- Whether you are using collocation on the next storage pool. When you use collocation, ADSM attempts to store data for different clients or client file spaces on separate tapes, even for clients with small amounts of data. You may want to set the low threshold to keep more data on disk, to avoid having lots of tapes used by clients with only small amounts of data.

Immediate User Access to Files on Disk Storage

Caching is a good method of providing immediate access to files on disk storage, even if the files have been migrated to a tape storage pool. However, cached files are removed from disk when the space they occupy is required. The file then must be obtained from the storage pool to which it was migrated.

To ensure that files remain on disk storage and do not migrate to other storage pools, use one of the following methods:

- Do not define the *next* storage pool.

  A disadvantage of using this method is that if the file exceeds the space available in the storage pool, the operation to store the file fails.

- Set the high-migration threshold to 100%.

  When you set the high migration threshold to 100%, files will not migrate at all. You can still define the *next* storage pool in the storage hierarchy, and set the maximum file size so that large files are stored in the next storage pool in the hierarchy.

  A disadvantage of setting the high threshold to 100% is that once the pool becomes full, client files are stored directly to tape instead of to disk. Performance may be affected as a result.

Migration Thresholds for Sequential Access Storage Pools

Migration from sequential storage pools is performed by volume, to minimize the number of mounts for source volumes. Sequential volumes selected for migration are those that were least recently referenced.

While you can define or update migration thresholds for sequential access storage pools, you probably will not perform this type of migration on a regular basis. This type of operation, such as tape-to-tape migration, has limited benefits compared to disk-to-tape migration and requires at least two tape drives.

However, you may find it necessary to migrate data from one sequential access storage pool to another. For example, if you install a different tape drive or you want to move tape volumes from an automatic tape library to shelf volumes, then migration from a sequential access storage pool may be appropriate.

When defining migration criteria for sequential access storage pools, consider:

- The capacity of the volumes in the storage pool
- The time required to migrate data to the next storage pool
- The speed of the devices that the storage pool uses
- The time required to mount media, such as tape volumes, into drives
- Whether operator presence is required

If you decide to migrate data from one sequential access storage pool to another, ensure that:

- Two drives (mount points) are available, one in each storage pool.

- The next storage pool in the storage hierarchy has read/write access.

  For information about setting an access mode for sequential access storage pools, see “Defining a Primary Storage Pool” on page 137.

- Collocation is set the same in both storage pools. For example, if collocation is set to *yes* in the first storage pool, then collocation should be set to *yes* in the subordinate storage pool.

  When you enable collocation for a storage pool, ADSM attempts to keep all files belonging to a client node or a client file space on a minimal number of volumes. For information about collocation for sequential access storage pools, see "Collocation on Sequential Access Storage Pools" on page 121.

- You have sufficient staff available to handle any necessary media mount and dismount operations, because the server attempts to reclaim space from sequential access storage pool volumes before it migrates files to the next storage pool.

  If you want to limit migration from a sequential access storage pool to another storage pool, set the high-migration threshold to a high percentage, such as 95%.

  For information about setting a reclamation threshold for tape storage pools, see "Space Reclamation for Sequential Access Storage Pools" on page 127.

There is no straightforward way to selectively migrate data for a specific node from one sequential storage pool to another. If you know the volumes on which a particular node's data is stored, you can use the MOVE DATA command to move all files from selected volumes to the new storage pool.

Migration and Copy Storage Pools

Copy storage pools are not part of the storage migration hierarchy. Files are not migrated to or from copy storage pools. The only way to store files in copy storage pools is by using the BACKUP STGPOOL command.

Migration of files between primary storage pools does not affect copy storage pool files. Copy storage pool files do not move when primary storage pool files move.

For example, suppose a copy of a file is made while it is in a disk storage pool. The file then migrates to a primary tape storage pool. If you then back up the primary tape storage pool to the same copy storage pool, a new copy of the file is not needed. ADSM knows it already has a valid copy of the file.

The Use of Cache on Disk Storage Pools

When defining or updating disk storage pools, you can enable or disable cache. When cache is enabled, the migration process leaves behind duplicate copies of files on disk after the server migrates these files to subordinate storage pools in the storage hierarchy. The copies remain in the disk storage pool, but in a *cached* state, so that subsequent retrieval requests can be satisfied quickly. However, if space is needed to store new data in the disk storage pool, the space occupied by cached files can be immediately reused for the new data.

When cache is not used and migration occurs, the server migrates the files to the next storage pool and erases the files from the disk storage pool.

By default, the system enables caching for each disk storage pool. You can change this option by specifying CACHE=NO when you define or update a storage pool.

Why Use Cache?

Using cache improves the retrievability of files, because a copy of the file remains on fast disk storage after the primary file is migrated.

When cache is used and migration occurs for the disk storage pool, the server migrates files to the next storage pool, but leaves cached copies of the migrated files in the disk storage pool. The cached copies remain in the disk storage pool until space is needed for new files.

When space is needed, the server reclaims space by writing over the cached files. Files that have the oldest retrieval date and occupy the largest amount of disk space are overwritten first. For example, if File A was last retrieved on 04/16/95 and File B was last retrieved on 06/19/95, then File A is deleted to reclaim space before File B.

Effect of Caching on Storage Pool Statistics: The space utilization statistic for the pool (%Util) includes the space used by any cached copies of files in the storage pool. The migratable data statistic (%Migr) does *not* include space occupied by cached copies of files. ADSM uses the migratable data statistic (%Migr) to compare with migration threshold parameters to determine when migration should begin or end. For more information on storage pool statistics, see “Monitoring the Use of Storage Pools” on page 144.

When Not to Use Cache

Do not use cache if you have limited space for the ADSM database. When you use cache, more database space is needed because the server has to keep track of both the cached copy of the file and the new copy in the subordinate storage pool.

If you disable cache, you may want to set higher migration thresholds for the disk storage pool. A higher migration threshold keeps files on disk longer because migration occurs less frequently.

Collocation on Sequential Access Storage Pools

Collocation is a process in which the server attempts to keep files belonging to a single client node or to a single file space of a client node on a minimal number of sequential access storage volumes. You can set collocation for each sequential access storage pool when you define or update the pool.

To have ADSM collocate data in a storage pool by client node, set collocation to *yes*. To have ADSM collocate data in a storage pool by client file space, set collocation to *filespace*. By using collocation, you reduce the number of volume mount operations required when users restore, retrieve, or recall many files from the storage pool. Collocation thus improves access time for these operations. Figure 20 on page 122 shows an example of collocation by client node with three clients, each having a separate volume containing that client’s data.

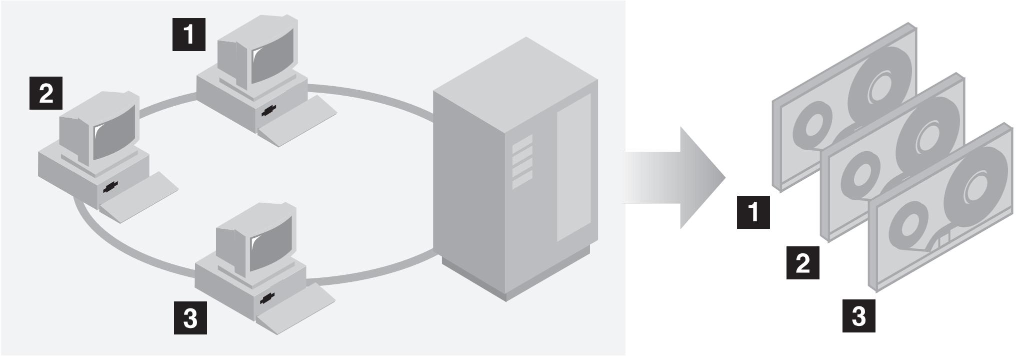


Figure 20. Example of Collocation Enabled

When collocation is disabled, the server attempts to use all available space on each volume before selecting a new volume. While this process provides better utilization of individual volumes, user files can become scattered across many volumes. Figure 21 shows an example of collocation disabled, with three clients sharing space on a volume.

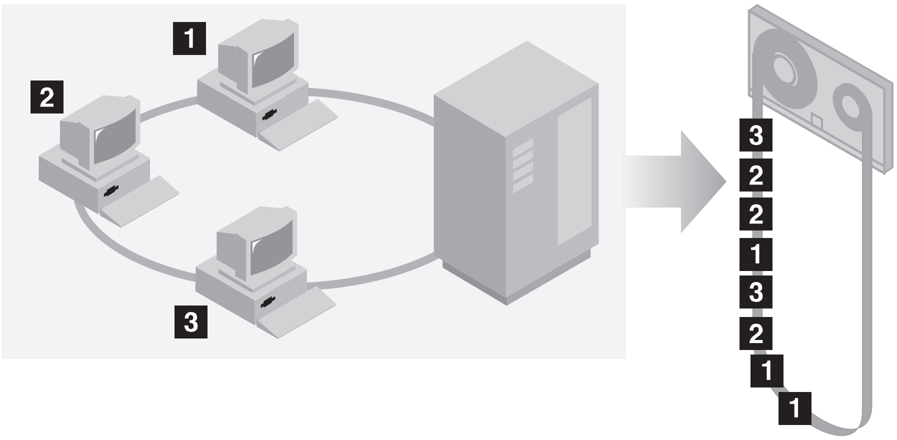


Figure 21. Example of Collocation Disabled

When users want to restore, retrieve, or recall a large number of files, media mount operators may be required to mount more volumes to recover user data. The system default is to not use collocation.

To determine whether to use collocation, consider:

- The amount of time available for backup processing

  If you have limited time for backup, disable collocation because with collocation you have more media mounts.

- The amount of time required to access a particular sequential access storage volume

  The access time depends mostly on the type of media involved in the operation. For example, if the storage pool is associated with a tape device class, the access time is relatively long, because tape volumes must be mounted into the appropriate type of drive by either an operator or robotics. However, if the device type of the device class associated with the storage pool is FILE, then the storage volumes can typically be accessed very quickly, and without manual intervention.

- Whether users need to be able to restore or retrieve a large number of files within a short period of time

  When users may need to restore or retrieve a large number of files and need fast response, enable collocation. Without collocation, your ability to recover files for users might be delayed because:

  – More than one user's files can be stored on the same sequential access storage volume.

    For example, if two users attempt to recover a file that resides on the same volume, the second user will be forced to wait until the first user's files are recovered.

  – A user's files can be spread across multiple volumes, requiring additional media mounts and dismounts by operators.

- How you want the server to utilize storage space

  When collocation is enabled, the server attempts to use all available tape volumes to separate user files before it uses all available space on every tape volume.

  When collocation is disabled, the server attempts to use all available space on each tape volume before it selects the next tape volume.

- Whether you have sufficient personnel to manage media mounts during backup, archive, or client migration operations

  While collocation helps to reduce the number of mount operations during recovery, operators may experience:

  – More mounts when user files are backed up, archived, or migrated from client nodes directly to sequential access volumes

  – More mounts during reclamation or migration

  – Additional handling of sequential access volumes because the volumes might not be fully used

  To reduce the number of media mounts and to use space on sequential volumes more efficiently, you can:

  – Define a storage pool hierarchy that requires backed up, archived, or space-managed files to be stored initially in disk storage pools.

    When files are migrated from a disk storage pool, the server attempts to migrate all files belonging to the client node that is using the most disk space in the storage pool. This process works well with the collocation option because the server tries to place all of the files from a given client on the same sequential access storage volume.

– Use scratch volumes for sequential access storage pools to allow the server to select new volumes for collocation.

How the Server Selects Volumes with Collocation Enabled

When collocation is enabled at the client node level (COLLOCATION=YES) and a client node backs up, archives, or migrates files to sequential access storage, the server attempts to select a volume that already contains files from the same client node. If no such volume exists, the server selects a volume using the following selection order:

1. An empty predefined volume
2. An empty scratch volume
3. A volume with the most available free space among volumes that already contain data

When collocation is enabled at the file space level (COLLOCATION=FILESPACE) and a client node backs up, archives, or migrates files to sequential access storage, the server attempts to select a volume that already contains files from the same file space of that client node. If no such volume exists, the server selects a volume using the following selection order:

1. A volume containing data from the same client node
2. An empty predefined volume
3. An empty scratch volume
4. A volume with the most available free space among volumes that already contain data

When the server needs to continue to store data on a second volume, it uses the following selection order to acquire additional space:

1. An empty predefined volume
2. An empty scratch volume
3. A volume on which other user files are already stored
4. A volume that has the most available free space
5. Any available volume in the storage pool

Through this selection process, the server attempts to provide the best use of individual volumes while minimizing the mixing of files from different clients or file spaces on volumes. For example, Figure 22 on page 125 shows that volume selection is *horizontal*, where all available volumes are used before all available space on each volume is used. A, B, C, and D represent files from four different client nodes.

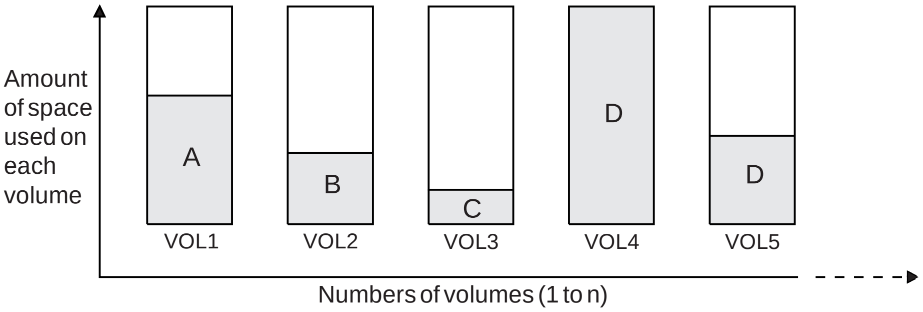

Amount of space used on each volume
A
B
C
D
D
VOL1
VOL2
VOL3
VOL4
VOL5
Numbers of volumes (1 to n)

Figure 22. Using All Available Sequential Access Storage Volumes with Collocation Enabled

How the Server Selects Volumes with Collocation Disabled

When collocation is disabled, the server attempts to use all available space in a storage volume before it accesses another volume. When storing client files in a sequential access storage pool where collocation is disabled, the server selects a volume using the following selection order:

1. A previously used sequential volume with available space (a volume with the most amount of data is selected first)
2. An empty volume

When the server needs to continue to store data on a second volume, it attempts to select an empty volume. If none exists, the server attempts to select any remaining available volume in the storage pool.

Figure 23 shows that volume utilization is *vertical* when collocation is disabled. In this example, fewer volumes are used because the server attempts to use all available space by mixing client files on individual volumes.

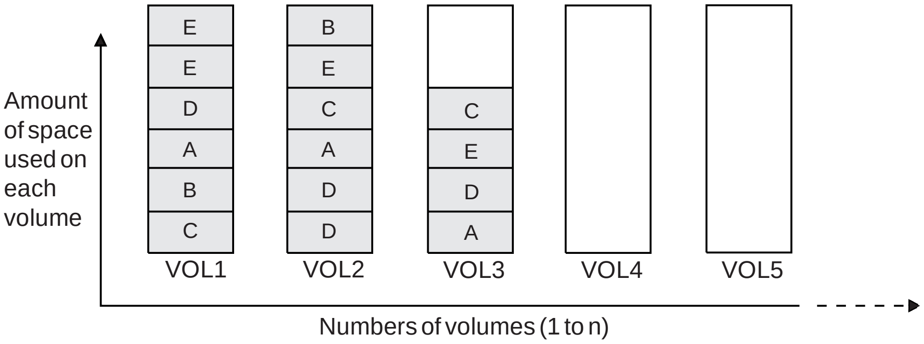

Amount of space used on each volume
E
E
D
A
B
C
B
E
C
A
D
D
C
E
D
A
VOL1
VOL2
VOL3
VOL4
VOL5
Numbers of volumes (1 to n)

Figure 23. Using All Available Space on Sequential Volumes with Collocation Disabled

Turning Collocation On or Off

After you define a storage pool, you can change the collocation setting by updating the storage pool. The change in collocation for the pool does not affect files that are already stored in the pool.

For example, if collocation had been off for a storage pool and you turn it on, *from then on* client files stored in the pool are collocated. Files that had previously been stored in the pool are *not* moved to collocate them. As volumes are reclaimed, however, the data in the pool tends to become more collocated. You can also use the MOVE DATA command to move data to new volumes to increase collocation, if you are able to afford the processing time and volume mount activity this would cause.

Collocation on Copy Storage Pools

There are special considerations when using collocation on copy storage pools. Primary and copy storage pools perform different recovery roles. Normally you use primary pools to recover data to clients directly, and you use copy storage pools to recover data to the primary pools. In a disaster where both clients and the server are lost, the copy storage pool volumes will probably be used directly to recover clients. The types of recovery scenarios that are of most concern to you will help to determine whether to use collocation on your copy storage pools.

Another consideration is that collocation on copy storage pools will result in more partially filled volumes and potentially unnecessary offsite reclamation activity.

Collocation typically results in a partially filled sequential volume for each client or client file space. This may be acceptable for primary storage pools because these partially filled volumes remain available and can be filled during the next migration process. However, for copy storage pools this may be unacceptable because the storage pool backups are usually made to be taken offsite immediately. If you use collocation for copy storage pools, you will have to decide between:

- Taking more partially filled volumes offsite, thereby increasing the reclamation activity when the reclamation threshold is lowered or reached.

or

- Leaving these partially filled volumes onsite until they fill and risk not having an offsite copy of the data on these volumes.

With collocation disabled for a copy storage pool, typically there will be only a single partially filled volume after storage pool backups to this copy storage pool are complete.

Consider carefully before using collocation for copy storage pools. Even if you use collocation for your primary storage pools, you may wish to disable collocation for copy storage pools. One example of when collocation on copy storage pools may be desirable is when you have few clients, but each of them has large amounts of incremental backup data each day.

See “Collocation on Sequential Access Storage Pools” on page 121 for more information about collocation.

Space Reclamation for Sequential Access Storage Pools

Space on a sequential volume becomes reclaimable as files expire or are deleted from the volume. For example, files become obsolete because of aging or version limits. When the percentage of reclaimable space exceeds a specified level, the *reclamation threshold*, the server begins space reclamation for the volume. You can set a reclamation threshold for each sequential access storage pool when you define or update the pool.

During space reclamation, the server copies active files from the candidate volume to other volumes in the storage pool. For example, Figure 24 on page 128 shows the active files from tapes 1, 2, and 3, being consolidated on tape 4.

WORM Optical Media: Because WORM optical media cannot be reused, storage pools that are assigned to device classes with a device type of WORM, WORM12, or WORM14 have a default reclamation value of 100. This prevents reclamation of WORM optical media.

To allow reclamation, you can set the reclamation value to something lower when defining or updating the storage pool. Reclaiming WORM volumes does not allow reuse, but does allow you to free library space by consolidating data on almost empty volumes to other volumes, and then ejecting the empty WORM volumes.

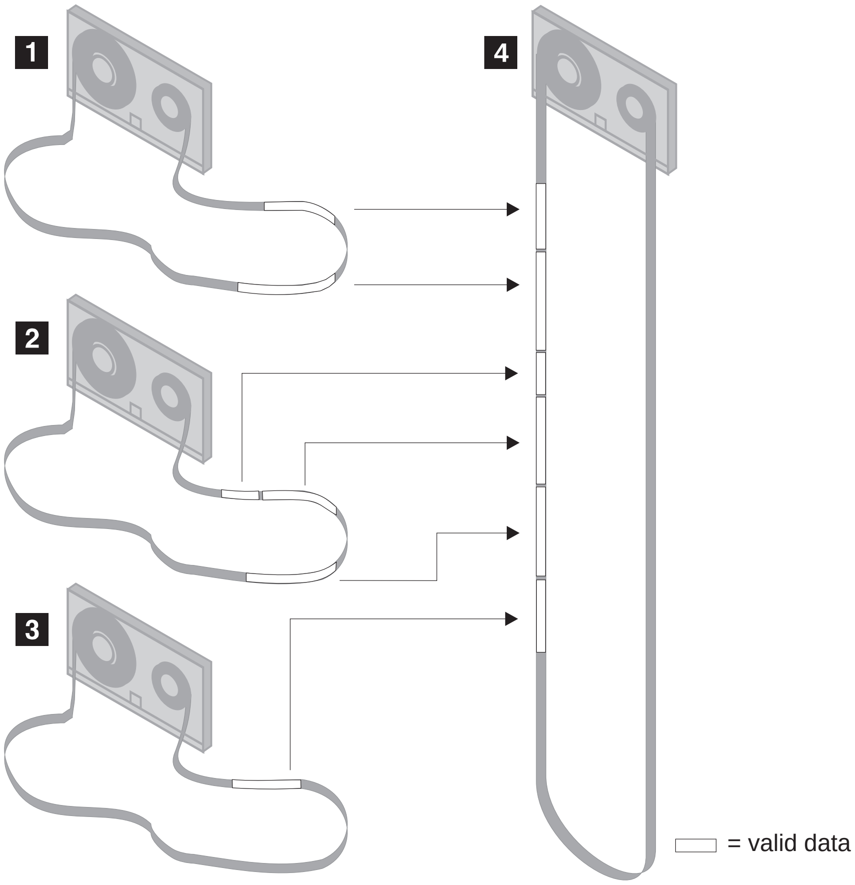


Figure 24. Tape Reclamation

After all readable files have been moved to other volumes, one of the following actions is taken for the candidate volume:

- If the volume has been defined to the storage pool, it becomes available for reuse by ADSM
- If the volume has been acquired as a scratch volume, it is deleted from the ADSM database

Volumes in a copy storage pool are reclaimed in the same manner as a primary storage pool with the exception of *offsite* volumes.

Choosing a Reclamation Threshold

The reclamation threshold indicates how much reclaimable space a volume must have before ADSM reclaims the volume. Space is reclaimable because it is occupied by files that have been expired or deleted from the ADSM database, or because it has not been used.

The lower the reclamation threshold, the more frequently the server tries to reclaim space. Frequent reclamation optimizes the use of a sequential access storage pool's space, but can interfere with other processes, such as backups from clients.

Each reclamation process requires *at least* two simultaneous volume mounts, that is, at least two mount points (drives) in the same device class. There must be a sufficient number of volumes, drives (if appropriate), and mount operators (if appropriate) to handle frequent reclamation requests. For more information about mount limit, see "Mount Limit" on page 93.

If you set the reclamation threshold to 50% or greater, ADSM can combine the usable files from two or more volumes onto a single new volume.

If the reclamation threshold is high, reclamation occurs less frequently. A high reclamation threshold is useful if mounting a volume is a manual operation and the operations staff is at a minimum.

Setting the reclamation threshold to 100% prevents reclamation from occurring at all. You might want to do this to control when reclamation occurs, to prevent interfering with other server processes. When convenient for you and your users, you can lower the reclamation threshold to cause reclamation to begin.

Reclamation for Copy Storage Pools

Reclamation of primary storage pool volumes does not affect copy storage pool files.

Reclamation of volumes in copy storage pools is similar to that of primary storage pools. One difference, however, is that most volumes in copy storage pools may be set to an access mode of offsite, making them ineligible to be mounted. During reclamation, valid files on offsite volumes are copied from the original files in the primary storage pools. In this way, valid files on offsite volumes are copied without having to mount these volumes. For more information, see "Reclamation of Offsite Volumes."

Reclamation of copy storage pool volumes should be done periodically to allow reuse of partially filled volumes that are offsite. Reclamation can be done automatically by setting the reclamation threshold for the copy storage pool to less than 100%. However, you need to consider controlling when reclamation occurs because of how offsite volumes are treated. For more information, see "Controlling When Reclamation Occurs for Offsite Volumes" on page 130.

Reclamation of Offsite Volumes

As for other volumes, volumes with the access value of offsite are eligible for reclamation if the amount of empty space on a volume exceeds the reclamation threshold for the copy storage pool. The default reclamation threshold for copy storage pools is 100%, which means that reclamation is not performed.

When an offsite volume is reclaimed, the files on the volume are rewritten to a *read/write* volume. Effectively these files are moved back to the onsite location, but may be obtained from the offsite volume after a disaster if a database backup is used that references the files on the offsite volume.

The ADSM server reclaims offsite volumes as follows:

1. The server determines which files are still active on the volume to be reclaimed.
2. These active files are obtained from a primary storage pool (or from an onsite volume of a copy storage pool, if necessary).
3. The active files are written to one or more new volumes in the copy storage pool and the database is updated.
4. A message is issued indicating that the offsite volume was reclaimed.

If you have the Disaster Recovery Manager feature, see “Moving Reclaimed or Expired Volumes Back Onsite” on page 367.

Controlling When Reclamation Occurs for Offsite Volumes

Suppose you plan to make daily storage pool backups to a copy storage pool, then mark all new volumes in the copy storage pool as *offsite* and send them to the offsite storage location. This strategy works well with one consideration if you are using automatic reclamation (reclamation threshold less than 100%).

Each day’s storage pool backups will create some number of new copy storage pool volumes, the last one being only partially filled. If this partially filled volume is emptier than the reclaim percentage, this volume becomes eligible for reclamation as soon as you mark it offsite. The reclamation process would cause a new volume to be created with the same files on it. The volume you take offsite would then be empty according to the ADSM database. If you do not recognize what is happening, you could perpetuate this process by marking the new partially filled volume offsite.

One way to resolve this situation is to keep partially filled volumes onsite until they fill up. However, this would mean a small amount of your data would be without an offsite copy for another day.

For this reason, it is recommended you control copy storage pool reclamation by using the default value of 100. This turns reclamation off for the copy storage pool. You can start reclamation processing at desired times by changing the reclamation threshold for the storage pool. To monitor offsite volume utilization and help you decide what reclamation threshold to use, enter the following:

```
query volume * access=offsite format=detailed
```

Depending on your data expiration patterns, you may not need to do reclamation of offsite volumes each day. You may choose to perform offsite reclamation on a less frequent basis. For example, suppose you ship copy storage pool volumes to and from your offsite storage location once a week. You can run reclamation for the copy storage pool weekly, so that as offsite volumes become empty they are sent back for reuse.

When you do perform reclamation for offsite volumes, the following sequence is recommended:

1 Back up your primary storage pools to copy storage pools

2 Turn on reclamation for copy storage pools by lowering the reclamation threshold

3 When reclamation processing completes, turn off reclamation for copy storage pools by raising the reclamation threshold to 100%

4 Mark any newly created, copy storage pool volumes as offsite and then move them to the offsite location

This sequence ensures that the files on the new copy storage pool volumes are sent offsite, and are not inadvertently kept onsite because of reclamation.

Delaying Reuse of Reclaimed Volumes

You should delay the reuse of any reclaimed volumes in copy storage pools for as long as you keep your oldest database backup. Delaying reuse may help you to recover data under certain conditions during recovery from a disaster. For more information on delaying volume reuse, see "Delaying Reuse of Sequential Access Volumes" on page 133.

How Collocation Affects Reclamation

If collocation is enabled and reclamation occurs, the server tries to reclaim the files for each client node or client file space onto a minimal number of volumes. Therefore, if the volumes are manually mounted, the mount operators must:

- Be aware that a tape volume may be rewound more than once if the server completes a separate pass to move the data for each client node or client file space.
- Mount and dismount multiple volumes to allow the server to select the most appropriate volume on which to move data for each client node or client file space. The server tries to select a volume in the following order:
  1. A volume that already contains files belonging to the client file space or client node
  2. An empty volume
  3. The volume with the most available space
  4. Any available volume

If collocation is disabled and reclamation occurs, the server tries to move usable data to new volumes by using the following volume selection criteria:

1. The volume that contains the most data
2. Any partially full volume

3. An empty predefined volume
4. An empty scratch volume

Reclamation in a Single-Drive Library

If a library defined to ADSM has only a single drive, ADSM cannot perform automatic reclamation for volumes in that library. To reclaim volumes in a single-drive library, use the MOVE DATA command. If the target storage pool is higher in the storage pool hierarchy than the original storage pool, the moved data will migrate back into the original storage pool and be written to a new volume. The original storage pool volume is then reclaimed.

Here is an example of how you can do this:

1 Define a device class with device type FILE.

2 Define a storage pool using the file device class. As the next storage pool, specify the tape storage pool associated with the single-drive library.

3 Move data from tape volumes that need to be reclaimed to the file storage pool.

4 Lower the high migration threshold for the file storage pool so that data migrates back to the tape storage pool. When the data migrates, it will be written to new volumes there.

Expiration Processing

When file spaces are deleted, backup files are versioned off, or archive files pass their archive retention period, these files are expired from the ADSM database. Later, when expiration processing runs, information about these files and also any copies of these files made in copy storage pools is removed from the database.

If backup policies are set up appropriately, the need to recover an expired file should be a rare occurrence. If this need occurs, expired files can be recovered by:

1. Restoring the database to a point in time prior to file expiration.
2. Using a primary or copy storage pool volume that has not been rewritten and contains the expired file data at the time of database backup.

You should delay the reuse of copy storage pool volumes that have no active files for as long as you keep your oldest database backup. Delaying reuse may help you to recover data under certain conditions during recovery from a disaster. For more information on delaying volume reuse, see “Delaying Reuse of Sequential Access Volumes” on page 133.

Delaying Reuse of Sequential Access Volumes

When you define or update a sequential access storage pool, you can use a parameter called REUSEDELAY. This parameter specifies the number of days that must elapse before a volume can be reused or returned to scratch status, after all files have been expired, deleted, or moved from the volume. When you delay reuse of such volumes, volumes enter the *pending* state once they no longer contain any files. Volumes remain in the pending state for as long as specified with the REUSEDELAY parameter for the storage pool to which the volume belongs.

Delaying reuse of volumes can be helpful under certain conditions for disaster recovery. When ADSM expires, deletes, or moves files from a volume, the files are not actually erased from the volumes: the database references to these files are removed. Thus the file data may still exist on sequential volumes if the volumes are not immediately reused.

If a disaster forces you to restore the ADSM database using a database backup that is old or is not the most recent backup, some files may not be recoverable because ADSM cannot find them on current volumes. Some of this data may exist on volumes that are in pending state, and you may be able to use them to recover data.

If you back up your primary storage pools, the REUSEDELAY parameter for the primary storage pools should be set to 0, to efficiently reuse primary scratch volumes. For your copy storage pools, you should delay reuse of volumes for as long as you keep your oldest database backup.

For an example of using database backup and delaying volume reuse, see “Protecting Your Database and Storage Pool” on page 352. For more information about expiration, see “Expiration Processing” on page 132.

How Restore Processing Works

ADSM provides two commands that allow an administrator to recreate files in a primary storage pool using copies in a copy storage pool:

RESTORE STGPOOL
Restores all files in a storage pool that have been previously identified as having data-integrity errors. These files are also known as *damaged* files. This command also restores all files on any volumes that have been designated as *destroyed* using the UPDATE VOLUME command. See “Restoring Storage Pools” on page 154 for more detailed information.

RESTORE VOLUME
Recreates files that reside on a volume or volumes in the same primary storage pool. This command can be used to recreate files for one or more volumes that have been lost or damaged. See “Restoring Storage Pool Volumes” on page 180 for more detailed information.

ADSM uses database information to determine which files should be restored for a volume or storage pool, so restore processing does not require that the original volumes be accessed. For example, if a primary storage pool volume becomes damaged, the RESTORE VOLUME command could be used to recreate files that were stored on that volume, even though the volume itself is not readable. However, if the administrator were to delete the damaged files with DISCARDDATA=YES, the database reference to the files on the primary storage pool volume and all references to copies of the files on copy storage pool volumes, would be removed from the database. It would not be possible to restore those files.

Restore processing obtains files from a copy storage pool and stores these files on new primary storage pool volumes. Database references to files on the original primary storage pool volumes are then deleted. If a primary storage pool volume becomes empty because all files that were stored on that volume have been restored to other volumes, the empty volume is automatically deleted from the database.

To facilitate restore processing of entire volumes, ADSM has a *destroyed* volume access mode. This mode is used to designate primary volumes for which files are to be restored. If a volume has an access mode of destroyed, ADSM does not mount that volume for either read or write access. You can change the access mode of a volume to destroyed in one of two ways:

- By using the RESTORE VOLUME command. The RESTORE VOLUME command automatically changes the access mode of specified volumes to destroyed using a volume list provided as part of the command.
- By using the UPDATE VOLUME command. Before using the RESTORE STGPOOL command to restore volumes in a storage pool, the administrator must update the access mode of the volumes to destroyed.

The destroyed designation for volumes is important during restore processing, particularly when the RESTORE STGPOOL command is used to restore a large number of primary storage pool volumes after a major disaster:

- You can designate as destroyed only those volumes that need to be restored. If some volumes are known to be usable after a disaster, the access state of the usable volumes should not be set to destroyed, so they will not be restored.
- Once the administrator has identified the primary volumes to be restored, and has changed the access mode of these volumes to destroyed, new volumes can be added to the storage pool. The new volumes are used to contain the files as they are restored from the copy storage pool volumes, and can also be used for storage of new files that may be backed up, archived, or migrated by the end users.
- The designation of destroyed volumes allows ADSM to keep track of the files that still need to be restored from copy storage pools. If restore processing is ended before completion for any reason, you can start the restore again. Processing would be resumed and only the files that still reside on destroyed volumes would need to be restored.

Estimating Space Needs for Storage Pools

This section provides guidelines for estimating the initial storage space required for your installation. It assumes the use of the following default random access (disk) storage pools provided by ADSM:

- BACKUPPOOL for backed up files
- ARCHIVEPOOL for archived files
- SPACEMGPOOL for files migrated from client nodes (space-managed files)

As your storage environment grows, you may want to consider how policy and storage pool definitions affect where workstation files are stored. Then you can define and maintain multiple storage pools in a hierarchy that allows you to contain storage costs by using sequential access storage pools in addition to disk storage pools, and still provide appropriate levels of service to users.

To help you determine how to adjust your policies and storage pools, get information about how much storage is being used (by client node) and for what purposes in your existing storage pools. For more information on how to do this, see "Requesting Information on Storage Occupancy" on page 151.

Estimating Space Needs in Random Access Storage Pools

To estimate the amount of storage space required for each random access (disk) storage pool:

- Determine the amount of disk space needed for different purposes:
  - For backup storage pools, provide enough disk space to support efficient daily incremental backups.
  - For archive storage pools, provide sufficient space for a user to archive a moderate size file system without causing migration from the disk pool to occur.
  - For storage pools for space-managed files, provide enough disk space to support the daily space-management load from HSM clients, without causing migration from the disk pool to occur.
- Decide what percentage of this data you want to keep on disk storage space and establish migration thresholds to have the server migrate the remainder of the data to less expensive storage media in sequential access storage pools.

  See "Appropriate Migration Threshold Values" on page 117 for recommendations on setting migration thresholds.

Estimating Space for Backed Up Files in a Random Access Storage Pool

To compute the total amount of space needed for all backed up files stored in a single random access (disk) storage pool, such as BACKUPPOOL, use the following formula:

```
Backup space = AvgWkstSize * Utilization * VersionExpansion * NumWkst
```

Backup Space
The total amount of storage pool disk space needed.

AvgWkstSize
The average data storage capacity of a workstation, in MB. For example, if the typical workstation at your installation has two 70MB hard drives, then the average workstation storage capacity is 140MB.

Utilization
An estimate of the fraction of each workstation disk space used, in the range 0 to 1. For example, if you expect that workstations are 75% full, then use 0.75.

VersionExpansion
An expansion factor (greater than 1) that takes into account the additional backup versions, as defined in the copy group. A rough estimate allows 5% additional files for each backup copy. For example, for a version limit of 2, use 1.05, and for a version limit of 3, use 1.10.

NumWkst
The estimated total number of workstations ADSM supports.

If clients use compression, the amount of space required may be less than the amount calculated, depending on whether the data is compressible.

Estimating Space for Archived Files in a Random Access Storage Pool

Computing the amount of storage space for archived files is more difficult, because the number of archived files generated by users is not necessarily proportional to the amount of data stored on their workstations.

To estimate the total amount of space needed for all archived files in a single random access (disk) storage pool, such as ARCHIVEPOOL, determine what percentage of user files are typically archived.

Work with policy administrators to calculate this percentage based on the number and type of archive copy groups defined. For example, if policy administrators have defined archive copy groups for only half of the policy domains in your enterprise, then you can estimate that you will need less then 50% of the amount of space you have defined for backed up files.

Because additional storage space can be added at any time, you can start with a modest amount of storage space and increase the space by adding storage volumes to the archive storage pool, as required.

Estimating Space Needs in Sequential Access Storage Pools

To estimate the amount of space required for sequential access storage pools, consider:

- The amount of data being migrated from disk storage pools
- The length of time backed up files are retained, as defined in backup copy groups
- The length of time archived files are retained, as defined in archive copy groups
- How frequently you reclaim unused space on sequential volumes

  See "Space Reclamation for Sequential Access Storage Pools" on page 127 for information about setting a reclamation threshold.
- Whether or not you use collocation to reduce the number of volume mounts required when restoring or retrieving large numbers of files from sequential volumes

  If you use collocation, you may need additional tape drives and volumes.

  See "Collocation on Sequential Access Storage Pools" on page 121 for information about using collocation for your storage pools.
- The type of storage devices and sequential volumes supported at your installation

Defining or Updating Storage Pools

This section provides examples of how you can set up a storage pool hierarchy for an organization in your installation.

Task	Required Privilege Class
Define storage pools	System
Update storage pool information	System or unrestricted storage

Defining a Primary Storage Pool

When you define a primary storage pool, be prepared to provide some or all of the information shown in Table 12. Some information applies only to random access storage pools or only to sequential access storage pools.

Table 12 (Page 1 of 3). Information for Defining a Storage Pool

Information	Explanation	Applies to Random Access	Applies to Sequential Access
Device class	Specifies the name of the device class assigned for the storage pool. This is a required parameter.	Yes	Yes

Table 12 (Page 2 of 3). Information for Defining a Storage Pool

Information	**Explanation**	**Applies to Random Access**	**Applies to Sequential Access**
Pool type	Specifies that you want to define a primary storage pool (this is the default). Updating a storage pool cannot change whether it is a primary or a copy storage pool.	Yes	Yes
Access mode	Defines access to volumes in the storage pool for user operations (such as back up and restore) and system operations (such as reclamation and server migration). Possible values are: **Read/Write** User and system operations can read from or write to the volumes. **Read-Only** User operations can read from the volumes, but not write. However, system processes can move files within the volumes in the storage pool. **Unavailable** No new writes are permitted to volumes in the storage pool from other volumes outside the storage pool. However, system processes (like reclamation) are permitted to move files within the volumes in the storage pool.	Yes	Yes
Maximum file size	To exclude large files from a storage pool, set a maximum file size. Do not set a maximum file size for the last storage pool in the hierarchy unless you want to exclude very large files from being stored in server storage.	Yes	Yes
Name of the next storage pool	Specifies the name of the next storage pool where files can be migrated or stored.	Yes	Yes
Migration thresholds	Specifies a percentage of storage pool occupancy at which ADSM begins migrating files to the next storage pool (high) threshold and the percentage when migration stops (low threshold).	Yes	Yes
Migration process	Specifies the number of processes that are used for migrating files from this storage pool.	Yes	—
Cache	Enables or disables cache. When cache is enabled, copies of files migrated by the server to the next storage pool are left on disk after the migration. In this way, a retrieval request can be satisfied quickly.	Yes	—
Maximum number of scratch volumes	By providing a nonzero value, you specify that ADSM dynamically acquires scratch volumes.	—	Yes
Collocation	*Collocation* is a process in which the server attempts to keep all files belonging to a client node or a client file space on a minimal number of sequential access storage volumes.	—	Yes

Table 12 (Page 3 of 3). Information for Defining a Storage Pool

Information	Explanation	Applies to Random Access	Applies to Sequential Access
Reclamation threshold	Specifies what percentage of reclaimable space can accumulate on a volume before the server initiates a space reclamation process for the volume.	—	Yes
Reuse delay period	Specifies an integer that defines the number of days that must elapse after all of the files have been deleted from a volume, before the volume can be rewritten or returned to the scratch pool.	—	Yes

Example: Defining a Storage Pool Hierarchy

For this example, suppose you have determined that an engineering department requires a separate storage hierarchy. You want the department's backed up files to go to a disk storage pool. When that pool fills, you want the files to migrate to a tape storage pool. You want the pools to have the following characteristics:

- Disk primary storage pool
  – The pool named ENGBACK1 is the storage pool for the engineering department.
  – The size of the largest file that can be stored is 5MB. Files larger than 5MB are stored in the tape storage pool.
  – Files migrate from the disk storage pool to the tape storage pool when the disk pool is 85% full. File migration to the tape storage pool stops when the disk pool is down to 40% full.
  – The access mode is the default, read/write.
  – Cache is used.
- Tape primary storage pool
  – The name of the pool is BACKTAPE.
  – The pool uses the device class TAPE, which has already been defined.
  – No limit is set for the maximum file size, because this is the last storage pool in the hierarchy.
  – To group files from the same client on a small number of volumes, use collocation at the client node level.
  – Use scratch volumes for this pool, with a maximum number of 100 volumes.
  – The access mode is the default, read/write.
  – Use the default for reclamation: Reclaim a partially full volume (to allow reuse) when 60% of the volume's space can be reclaimed.

There are two ways to define the storage pools in a storage pool hierarchy: from the top down or from the bottom up. Defining the hierarchy from the bottom up requires fewer steps. To define the hierarchy from the bottom up, perform the following steps:

1. Define the storage pool named BACKTAPE with the following command:

```
define stgpool backtape tape
description='tape storage pool for engineering backups'
maxsize=nolimit collocate=yes maxscratch=100
```

2. Define the storage pool named ENGBACK1 with the following command:

```
define stgpool engback1 disk
description='disk storage pool for engineering backups'
maxsize=5M nextstgpool=backtape highmig=85 lowmig=40
```

Example: Updating a Storage Pool Hierarchy

If you have already defined the storage pool at the top of the hierarchy, you can update the storage hierarchy to include a new storage pool.

For example, suppose you had already defined the ENGBACK1 disk storage pool. Now you have decided to set up a tape storage pool to which files from ENGBACK1 can migrate. Perform the following steps to define the new tape storage pool and update the hierarchy:

1. Define the storage pool named BACKTAPE with the following command:

```
define stgpool backtape tape
description='tape storage pool for engineering backups'
maxsize=nolimit collocate=yes maxscratch=100
```

2. Specify that BACKTAPE is the next storage pool defined in the storage hierarchy for ENGBACK1. To update ENGBACK1, enter:

```
update stgpool engback1 nextstgpool=backtape
```

Defining a Copy Storage Pool

When you define a copy storage pool, be prepared to provide some or all of the following information:

Device class
Specifies the name of the device class assigned for the storage pool. This is a required parameter.

Pool type
Specifies that you want to define a copy storage pool. This is a required parameter. Updating a storage pool cannot change whether the pool is a primary or copy storage pool.

Access mode
Defines access to volumes in the storage pool for user operations (such as backup and restore) and system operations (such as reclamation). Possible values are:

Read/Write User and system operations can read from or write to the volumes.

Read-Only User operations can read from the volumes, but not write. However, system processes can move files within the volumes in the storage pool.

Unavailable Specifies that users cannot access files stored on volumes in the copy storage pool. Files can be moved within the volumes of the copy storage pool, but no new writes are permitted to the volumes in the storage pool from volumes outside the storage pool.

Maximum number of scratch volumes
By providing a nonzero value, you specify that ADSM dynamically acquires scratch volumes.

Collocation
Collocation is a process in which the server attempts to keep all files belonging to a client node or a client file space on a minimal number of sequential access storage volumes.

Reclamation threshold
Specifies when to initiate reclamation of volumes in the copy storage pool. Reclamation is a process that moves any remaining active, fragmented files from one volume to another volume, thus making the original volume available for reuse. A volume is eligible for reclamation when the percentage of unused space on the volume is greater than the reclaim parameter value.

Reclamation processing works differently for offsite storage pool volumes compared to other volumes. When a copy storage pool volume that is offsite becomes eligible for reclamation, the reclamation process attempts to retrieve the active files on the reclaimable volume from a primary or copy storage pool volume that is onsite, and then write these files to an available volume in the original copy storage pool.

Reuse delay period

Specifies an integer that defines the number of days that must elapse after all of the files have been deleted from a volume before the volume can be rewritten or returned to the scratch pool.

Example: Defining a Copy Storage Pool

Assume you need to have copies of the files stored in BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL (default disk storage pools) for disaster recovery purposes. An ADSM administrator uses the DEFINE STGPOOL command to create a copy storage pool named DISASTER-RECOVERY. It was decided to use only scratch tapes so the maximum number of scratch volumes is set to an appropriate value.

```
define stgpool disaster-recovery tape8mm pooltype=copy
maxscratch=100
```

To store data in the new storage pool, you must back up the primary storage pools (BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL) to the DISASTER-RECOVERY pool. See “Backing Up Storage Pools.”

Backing Up Storage Pools

Administrators can back up primary storage pools into copy storage pools.

Multiple primary storage pools can be backed up to one copy storage pool. A primary storage pool can be backed up to multiple copy storage pools if multiple copies are necessary. However, it is recommended that the entire primary storage pool hierarchy be backed up to the same copy storage pool for easier management of storage volumes.

Task	Required Privilege Class
Back up storage pools	System, unrestricted storage, or restricted storage for the copy storage pool

The BACKUP STGPOOL command is used to copy files into a copy storage pool. Because the copies are made incrementally, the backup process may be canceled if desired. Reissuing the BACKUP STGPOOL command allows the backup to continue from the spot the backup was canceled. For example, to back up the ARCHIVEPOOL primary pool to the DISASTER-RECOVERY copy pool, enter:

```
backup stgpool archivepool disaster-recovery
```

The BACKUP STGPOOL command can also be scheduled. The administrator can define schedules to initiate incremental backups of files in the primary storage pools.

For example, to back up the BACKUPPOOL, ARCHIVEPOOL, and the TAPEPOOL every night, the following commands are scheduled:

```
backup stgpool backuppool disaster-recovery maxprocess=4

backup stgpool archivepool disaster-recovery maxprocess=4

backup stgpool tapepool disaster-recovery maxprocess=4
```

These commands use four parallel processes to perform an incremental backup of each primary storage pool to the copy pool. The only files backed up to the DISASTER-RECOVERY pool are files for which a copy does not already exist in the copy storage pool. See Chapter 11, “Automating Operations” on page 223 for information about scheduling commands.

Notes:

1. Backing up storage pools places additional space requirements on the ADSM database.
2. If a copy is to be generated in a specific copy storage pool and a copy already exists with the same insertion date, no action is taken.
3. File copies stored in a copy storage pool do not migrate from that copy storage pool to any other.
4. Copies of files that remain on disk after being migrated to the next storage pool (cached files) are not backed up when the disk storage pool is backed up.
5. Set the MAXPROCESS parameter to the number of mount points or drives that can be dedicated to this operation.

See “Backing Up Storage Pools” on page 332 for more information about using storage pool backup in your disaster recovery strategy.

Using Copy Storage Pools to Improve Data Availability

Copy storage pools enable multiple copies of files to be maintained, thus reducing the potential for data integrity loss due to media failure. If the primary file is not available or becomes corrupted, ADSM accesses and uses the duplicate file from a copy storage pool.

Example: Simple Hierarchy with One Copy Storage Pool

A company has a storage hierarchy consisting of two primary storage pools: one random access storage pool (DISK-POOL) and one tape storage pool (8MM-POOL, with device class TAPE8MM). The files stored in the random access storage pool are migrated to the tape storage pool. Because the files are important to the function of the company, the company wants to back up the files in both primary storage pools to a copy storage pool.

The administrator decides to schedule daily incremental backups of the files in the primary storage pools. The administrator performs the following steps:

1. Create a copy storage pool called 8MM-COPYPOOL, with the same device class as the 8MM-POOL primary storage pool, by issuing the following command:

```
define stgpool 8mm-copypool tape8mm pooltype=copy
maxscratch=50
```

   Notes:

   a. Because scratch volumes are allowed in this copy storage pool, you do not need to define volumes for the pool.

   b. All of the storage volumes in the copy storage pool 8MM-COPYPOOL are located onsite.

2. Perform the initial backup of the primary storage pools to the new copy storage pool. Copy the files in the primary storage pools to the copy storage pool 8MM-COPYPOOL by issuing the following commands:

```
backup stgpool disk-pool 8mm-copypool

backup stgpool 8mm-pool 8mm-copypool
```

3. Define schedules to automatically run the commands for backing up the primary storage pools to the copy storage pool. The commands to schedule are those that you issued in step 2.

   To minimize tape mounts, back up the disk storage pool first, then the tape storage pool.

   For more information about scheduling, see Chapter 11, “Automating Operations” on page 223.

Monitoring the Use of Storage Pools

Any administrator can query for information about a storage pool by viewing a standard or a detailed report. Use these reports to monitor storage pool usage, including:

- Whether you need to add space to your disk and sequential access storage pools
- The status of the process of migrating data from one to storage pool to the next storage pool in the storage hierarchy
- The use of disk space by cached copies of files that have been migrated to the next storage pool

Monitoring the Use of Storage Pool Space

To query the server to view a standard report for all storage pools defined to the system, enter:

```
query stgpool
```

Figure 25 shows a standard report with all storage pools defined to the system. To monitor the use of storage pool space, review the *Estimated Capacity* and *%Util* columns.

```
Storage      Device       Estimated  %Util  %Migr  High   Low  Next
Pool Name    Class Name    Capacity               Mig%  Mig%  Storage
                               (MB)                           Pool
-----------  ----------  ----------  -----  -----  ----  ----  -----------
ARCHIVEPOOL  DISK               0.0    0.0    0.0    90    70
BACKTAPE     TAPE             180.0   85.0  100.0    90    70
BACKUPPOOL   DISK              80.0   51.6   51.6    50    30  BACKTAPE
COPYPOOL     TAPE             300.0   42.0
ENGBACK1     DISK               0.0    0.0    0.0    85    40  BACKTAPE
```

Figure 25. Information about Storage Pools

Estimated Capacity

Specifies the space available in the storage pool in megabytes.

For disk storage pools, this value reflects the total amount of available space in the storage pool, including any volumes that are varied offline.

For sequential access storage pools, this value is an estimate of the total amount of available space on all volumes in the storage pool, including volumes that have *unavailable*, *read-only*, *offsite*, or *destroyed* access mode, and all scratch volumes that can be acquired in this storage pool. Volumes in a sequential access storage pool, unlike those in a disk storage pool, do not contain preallocated space. Data is written to a volume as necessary until the end of the volume is reached. For this reason, the estimated capacity is truly an *estimate* of the amount of available space in a sequential access storage pool.

%Util

Specifies, as a percentage, the space used in each storage pool.

For disk storage pools, this value reflects the total number of disk blocks currently allocated by ADSM. Space is allocated for backed up, archived, or space-managed files that are eligible for server migration, cached files that are copies of server-migrated files, and files that reside on any volumes that are varied offline.

Note: The value for %Util can be higher than the value for %Migr if you query for storage pool information while a backup or archive transaction is in progress. The value for %Util is determined by the amount of space actually allocated (while the transaction is in progress), while the value for %Migr only represents the space occupied by *committed* files. At the end of the transaction, %Util and %Migr become synchronized.

For sequential access storage pools, this value is the percentage of the total bytes of storage available that are currently being used to store active (non-expired) data. Because the server can only estimate the available capacity of a sequential access storage pool, this percentage also reflects an estimate of the actual utilization of the storage pool.

Example: Monitoring the Capacity of a Backup Storage Pool

Figure 25 on page 145 shows that the estimated capacity for a disk storage pool named BACKUPPOOL is 80MB, which is the amount of available space on disk storage. More than half (51.6%) of the available space is occupied by either backup files or cached copies of backup files.

The estimated capacity for the tape storage pool named BACKTAPE is 180MB, which is the total estimated space available on all tape volumes in the storage pool. This report shows that 85% of the estimated space is currently being used to store workstation files.

Note: This report also shows that volumes have not yet been defined to the ENGBACK1 storage pool, because the storage pool shows an estimated capacity of 0.0MB.

Monitoring Migration Processes

Four fields on the standard storage pool report provide you with information about the migration process. They include:

%Migr

Specifies the percentage of data in each storage pool that can be migrated. This value is used to determine when to start or stop migration.

For disk storage pools, this value represents the amount of disk space occupied by backed up, archived, or space-managed files that can be migrated to another storage pool, including files on volumes that are varied offline. Cached data are excluded in the %Migr value.

For sequential access storage pools, this value is the percentage of the total volumes in the storage pool that actually contain data at the moment. For example, assume a storage pool has four explicitly defined volumes, and a maximum scratch value of six volumes. If only two volumes actually contain data at the moment, then %Migr will be 20%.

This field is blank for copy storage pools.

High Mig%

Specifies when ADSM can begin migrating data from this storage pool. Migration can begin when the percentage of data that can be migrated reaches this threshold. (This field is blank for copy storage pools.)

Low Mig%

Specifies when ADSM can stop migrating data from this storage pool. Migration can end when the percentage of data that can be migrated falls below this threshold. (This field is blank for copy storage pools.)

Next Storage Pool

Specifies the primary storage pool destination to which data is migrated. (This field is blank for copy storage pools.)

Example: Monitoring the Migration of Data Between Storage Pools

ADSM sets a default of 90% for the high migration threshold and 70% for the low migration threshold for each primary storage pool.

Figure 25 on page 145 shows that the predefined migration thresholds for BACKUPPOOL storage pool have been updated to 50% for the *high migration threshold* and 30% for the *low migration threshold*.

When the amount of migratable data stored in the storage pool reaches 50%, the server can begin to migrate files to BACKTAPE.

To monitor the migration of files from BACKUPPOOL to BACKTAPE, enter:

```
query stgpool back*
```

See Figure 26 for an example of the results of this command.

If caching is on for a disk storage pool and files are migrated, the %Util value does not change because the cached files still occupy space in the disk pool. However, the %Migr value decreases because the space occupied by cached files is no longer migratable.

```
Storage      Device       Estimated  %Util  %Migr  High   Low  Next
Pool Name    Class Name    Capacity               Mig%  Mig%  Storage
                               (MB)                           Pool
-----------  ----------  ----------  -----  -----  ----  ----  -----------
BACKTAPE     TAPE             180.0   95.2  100.0    90    70
BACKUPPOOL   DISK              80.0   51.6   28.8    50    30  BACKTAPE
```

Figure 26. Information on Backup Storage Pools

You can query the server to monitor the migration process by entering:

```
query process
```

A message similar to Figure 27 on page 148 is displayed:

```
 Process Process Description      Status
  Number
-------- ------------------------ ---------------------------------------------
       2 Migration                Disk Storage Pool BACKUPPOOL, Moved Files:
                                   1086, Moved Bytes: 25555579, Unreadable
                                   Files: 0, Unreadable Bytes: 0
```

Figure 27. Information on the Migration Process

When migration is finished, the server displays the following message:

```
ANR1101I Migration ended for storage pool BACKUPPOOL.
```

Handling Problems during the Migration Process

A problem can occur during the migration process that causes the migration process to be suspended. For example, there may not be sufficient space in the storage pool to which data is being migrated. When migration is suspended, the process might be retried.

At this point, a system administrator can:

- Cancel the migration process. See “Canceling the Migration Process” for additional information.
- End the migration process by changing the attributes of the storage pool from which data is being migrated. See “Ending the Migration Process by Changing Storage Pool Characteristics” on page 149 for additional information.
- Provide additional space. See “Providing Additional Space for the Migration Process” on page 149 for additional information.

The server attempts to restart the migration process every 60 seconds for several minutes and then will terminate the migration process.

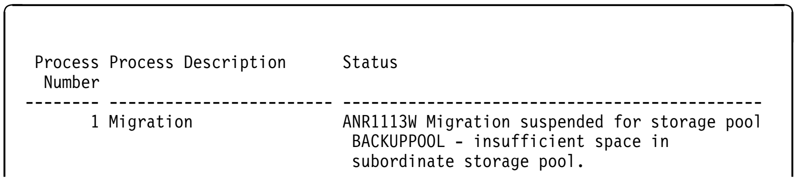
Canceling the Migration Process

To stop server migration when a problem occurs or when you need the resources the process is using, you can cancel the migration.

First determine the identification number of the migration process by entering:

```
query process
```

A message similar to Figure 28 on page 149 is displayed:



```
 Process Process Description      Status
  Number
-------- ------------------------ ---------------------------------------------
       1 Migration                ANR1113W Migration suspended for storage pool
                                   BACKUPPOOL - insufficient space in
                                   subordinate storage pool.
```

Figure 28. Getting the Identification Number of the Migration Process

Then you can cancel the migration process by entering:

```
cancel process 1
```

Ending the Migration Process by Changing Storage Pool Characteristics

Some errors cause the server to continue attempting to restart the migration process after 60 seconds. (If the problem still exists after several minutes, the migration process will end.) To stop the repeated attempts at restart, you can change some characteristics of the storage pool from which data is being migrated. Depending on your environment, you can:

- Set higher migration thresholds for the storage pool from which data is being migrated. The higher threshold means the storage pool must have more migratable data before migration starts. This change delays migration.

  In the example in “Example: Monitoring the Migration of Data Between Storage Pools” on page 147, you could update the disk storage pool BACKUPPOOL.

- Add volumes to the pool from which data is being migrated. Adding volumes decreases the percentage of data that is migratable (%Migr).

  In the example in “Example: Monitoring the Migration of Data Between Storage Pools” on page 147, you could add volumes to the disk storage pool BACKUPPOOL to increase its storage capacity.

  Note: Do this only if you received an out-of-space message for the storage pool to which data is being migrated.

Providing Additional Space for the Migration Process

A migration process can be suspended because of insufficient space in the storage pool to which data is being migrated. To allow the migration process to complete, you can provide additional storage volumes for that storage pool.

In the example in “Example: Monitoring the Migration of Data Between Storage Pools” on page 147, you could add volumes to the BACKTAPE storage pool or increase the maximum number of scratch tapes allowed for it. Either way, you increase the storage capacity of BACKTAPE.

Monitoring the Use of Cache Space on Disk Storage

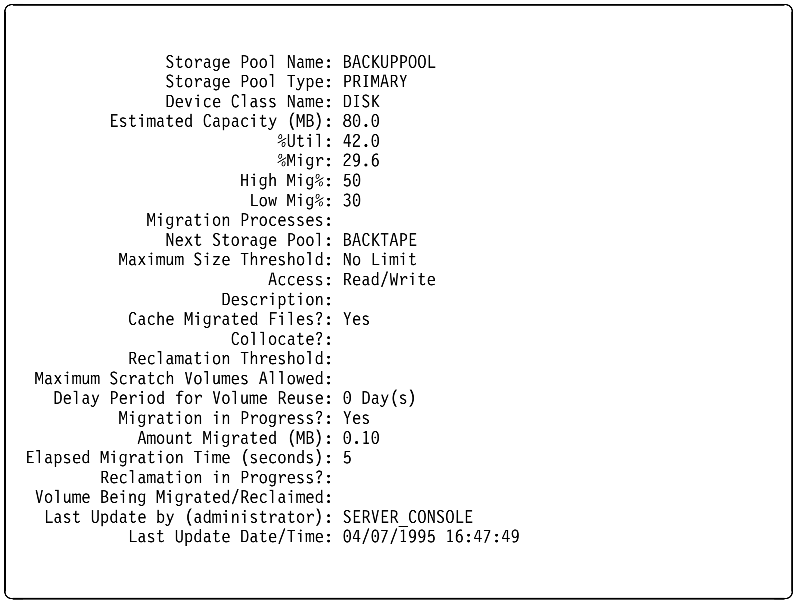
The %Util value includes cached data on a volume (when cache is enabled) and the %Migr value excludes cached data. Therefore, when cache is enabled and migration occurs, the %Migr value decreases while the %Util value remains the same. The %Util value remains the same because the migrated data remains on the volume as cached data. In this case, the %Util value only decreases when the cached data expires.

If you update a storage pool from CACHE=YES to CACHE=NO, the cached files will not disappear immediately. The %Util value will be unchanged. The cache space will be reclaimed over time as the server needs the space, and no additional cached files will be created.

To determine whether cache is being used on disk storage and to monitor how much space is being used by cached copies, query the server for a detailed storage pool report. For example, to request a detailed report for BACKUPPOOL, enter:

```
query stgpool backuppool format=detailed
```

Figure 29 displays a detailed report for the storage pool.



```
               Storage Pool Name: BACKUPPOOL
               Storage Pool Type: PRIMARY
               Device Class Name: DISK
         Estimated Capacity (MB): 80.0
                           %Util: 42.0
                           %Migr: 29.6
                       High Mig%: 50
                        Low Mig%: 30
             Migration Processes:
               Next Storage Pool: BACKTAPE
          Maximum Size Threshold: No Limit
                          Access: Read/Write
                     Description:
           Cache Migrated Files?: Yes
                      Collocate?:
           Reclamation Threshold:
 Maximum Scratch Volumes Allowed:
   Delay Period for Volume Reuse: 0 Day(s)
          Migration in Progress?: Yes
            Amount Migrated (MB): 0.10
Elapsed Migration Time (seconds): 5
        Reclamation in Progress?:
 Volume Being Migrated/Reclaimed:
  Last Update by (administrator): SERVER_CONSOLE
           Last Update Date/Time: 04/07/1995 16:47:49
```

Figure 29. Detailed Storage Pool Report

When *Cache Migrated Files?* is set to *yes*, the value for %Util should not change because of migration, because cached copies of files migrated to the next storage pool remain in disk storage.

This example shows that utilization remains at 42%, even after files have been migrated to the BACKTAPE storage pool, and the current amount of data eligible for migration is 29.6%.

When *Cache Migrated Files?* is set to *no*, the value for %Util more closely matches the value for %Migr because cached copies are not retained in disk storage.

Requesting Information on Storage Occupancy

Task	Required Privilege Class
Query the server for information about server storage	Any administrator

Any administrator can request information about server storage occupancy. Use the QUERY OCCUPANCY command for reports with information broken out by node or file space. Use this report to determine the amount of space used by:

- Client node and file space
- Storage pool or device class
- Type of data (backup, archive, or space-managed)

You can also use this report to evaluate the average size of workstation files stored in server storage.

Amount of Space Used by Client Node

Any administrator can request information about the space used by each client node and file space:

- How much data has been backed up, archived, or migrated to server storage
- How many of the files that are in server storage have been backed up to a copy storage pool
- The amount of storage space being used

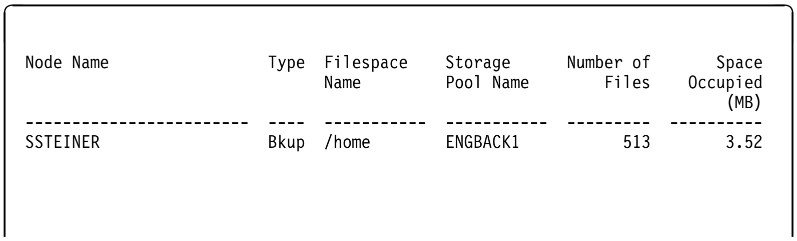
To determine the amount of server storage space used by the /home file space belonging to the client node SSTEINER, for example, enter:

```
query occupancy ssteiner /home
```

Remember that file space names are case-sensitive and must be entered exactly as they are known to the server. Use the QUERY FILESPACE command to determine the correct capitalization. For more information, see “Requesting Information about File Spaces” on page 298.

Figure 30 on page 152 shows the results of the query. The report shows the number of files backed up, archived, or migrated from the /home file space belonging to SSTEINER. The report also shows how much space is occupied in each storage pool.

If you back up the ENGBACK1 storage pool to a copy storage pool, the copy storage pool would also be listed in the report. To determine how many of the client node's files in the primary storage pool have been backed up to a copy storage pool, compare the number of files in each pool type for the client node.



```
Node Name                 Type  Filespace    Storage      Number of       Space
                                Name         Pool Name        Files    Occupied
                                                                           (MB)
------------------------  ----  -----------  -----------  ---------  ----------
SSTEINER                  Bkup  /home        ENGBACK1           513        3.52
```

Figure 30. A Report of the Occupancy of Storage Pools by Client Node

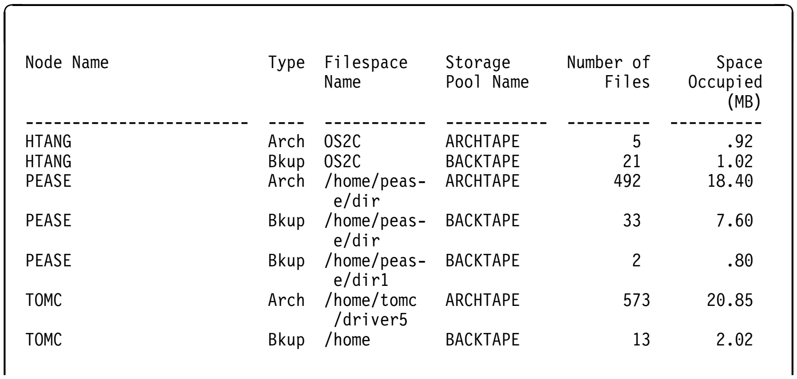
Amount of Space Used by Storage Pool or Device Class

You can monitor the amount of space being used by an individual storage pool, a group of storage pools, or storage pools categorized by a particular device class. Creating occupancy reports on a regular basis can help you with capacity planning.

To query the server for the amount of data stored in backup tape storage pools belonging to the TAPE8MM device class, for example, enter:

```
query occupancy devclass=tape8mm
```

Figure 31 displays a report on the occupancy of tape storage pools assigned to the TAPE8MM device class.



```
Node Name                 Type  Filespace    Storage      Number of       Space
                                Name         Pool Name        Files    Occupied
                                                                           (MB)
------------------------  ----  -----------  -----------  ---------  ----------
HTANG                     Arch  OS2C         ARCHTAPE            5         .92
HTANG                     Bkup  OS2C         BACKTAPE           21        1.02
PEASE                     Arch  /home/peas-  ARCHTAPE          492       18.40
                                  e/dir
PEASE                     Bkup  /home/peas-  BACKTAPE           33        7.60
                                  e/dir
PEASE                     Bkup  /home/peas-  BACKTAPE            2         .80
                                  e/dir1
TOMC                      Arch  /home/tomc   ARCHTAPE           573       20.85
                                  /driver5
TOMC                      Bkup  /home        BACKTAPE            13        2.02
```

Figure 31. A Report on the Occupancy of Storage Pools by Device Class

Note: For archived data, you may see “(archive)” in the Filespace Name column instead of a file space name. This means that the data was archived before collocation by file space was supported by the server.

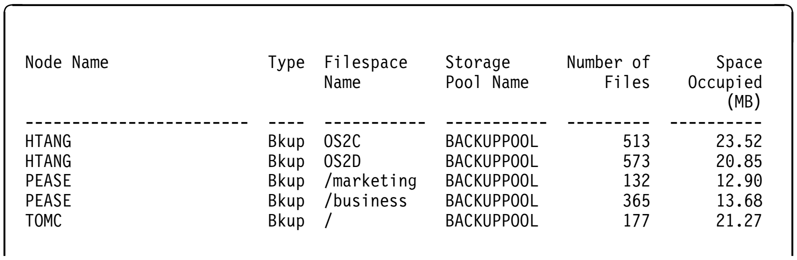
Amount of Space Used by Backed Up, Archived, or Space-Managed Files

You can query the server for the amount of space used by backed up, archived, and space-managed files. By determining the average size of workstation files stored in server storage, you can estimate how much storage capacity you might need when registering new client nodes to the server. See “Estimating Space Needs for Storage Pools” on page 135 and “Estimating Space for Archived Files in a Random Access Storage Pool” on page 136 for information about planning storage space.

To request a report about backup versions stored in the disk storage pool named BACKUPPOOL, for example, enter:

```
query occupancy stgpool=backuppool type=backup
```

Figure 32 displays a report on the amount of server storage used for backed up files.



```
Node Name                  Type  Filespace    Storage      Number of       Space
                                 Name         Pool Name        Files    Occupied
                                                                            (MB)
------------------------   ----  -----------  -----------  ---------  ----------
HTANG                      Bkup  OS2C         BACKUPPOOL         513       23.52
HTANG                      Bkup  OS2D         BACKUPPOOL         573       20.85
PEASE                      Bkup  /marketing   BACKUPPOOL         132       12.90
PEASE                      Bkup  /business    BACKUPPOOL         365       13.68
TOMC                       Bkup  /            BACKUPPOOL         177       21.27
```

Figure 32. A Report of the Occupancy of Backed Up Files in Storage Pools

To determine the average size of backup versions stored in BACKUPPOOL, complete the following steps using the data provided in Figure 32:

1 Add the number of megabytes of space occupied by backup versions.

In this example, backup versions occupy 92.22MB of space in BACKUPPOOL.

2 Add the number of files stored in the storage pool.

In this example, 1760 backup versions reside in BACKUPPOOL.

3 Divide the space occupied by the number of files to determine the average size of each file backed up to the BACKUPPOOL.

In this example, the average size of each workstation file backed up to BACKUPPOOL is about 0.05MB, or approximately 50KB.

You can use this average to estimate the capacity required for additional storage pools that are defined to ADSM.

Deleting a Storage Pool

Task	Required Privilege Class
Delete storage pools	System

Before a storage pool can be deleted, ensure that:

- All volumes within the storage pool have been deleted

  Ensure that you have saved any readable data that you want to preserve by issuing the MOVE DATA command. Moving all of the data that you want to preserve may require you to issue the MOVE DATA command several times.

  Before you begin deleting all volumes that belong to the storage pool, change the access mode of the storage pool to unavailable so that no files can be written to or read from volumes in the storage pool.

  See “Deleting a Storage Pool Volume with Data” on page 179 for information about deleting storage volumes.

- The storage pool is not identified as the next storage pool within the storage hierarchy

  To determine whether this storage pool is referenced as the next storage pool within the storage hierarchy, query for storage pool information as described in “Monitoring the Use of Storage Pool Space” on page 145.

  Update any storage pool definitions to remove this storage pool as a subordinate storage pool in the storage hierarchy by performing one of the following:

  – Naming another storage pool as the next storage pool in the storage hierarchy

  – Entering double quotes ("") on the *next* parameter to remove this storage pool from the storage hierarchy definition.

  See “Defining or Updating Storage Pools” on page 137 for information about updating storage pool definitions.

- The storage pool to be deleted is not specified as the destination for any copy group in any management class within the active policy set of any domain. Also, a storage pool to be deleted cannot be the destination for space-managed files (specified in any management class within the active policy set of any domain). If this pool is a destination and the pool is deleted, operations fail because there is no storage space to store the data.

Restoring Storage Pools

An administrator can recreate files in a primary storage pool using duplicate copies in copy storage pools by issuing the RESTORE STGPOOL command. The files must have been copied to the copy storage pools by using the BACKUP STGPOOL command.

Task	Required Privilege Class
Restoring storage pools	System, unrestricted storage, or restricted storage

The RESTORE STGPOOL command restores specified primary storage pools that have files with the following problems:

- The primary copy of the file has been identified as having data-integrity errors during a previous operation. Files with data-integrity errors are marked as damaged.
- The primary copy of the file resides on a volume that has an access mode of DESTROYED. For how the access mode of a volume changes to the DESTROYED access mode, see "How Restore Processing Works" on page 133.

When you restore a storage pool, be prepared to provide the following information:

Primary storage pool
Specifies the name of the primary storage pool that is being restored.

Copy storage pool
Specifies the name of the copy storage pool from which the files are to be restored. This information is optional. If you do not specify a particular copy storage pool, ADSM restores the files from any copy storage pool where it can find them.

New storage pool
Specifies the name of the new primary storage pool to which to restore the files. This information is optional. If you do not specify a new storage pool, ADSM restores the files to the original primary storage pool.

Maximum number of processes
Specifies the number of parallel processes that are used for restoring files.

Preview
Specifies whether you want to preview the restore operation before it is actually performed.

See "Correcting Damaged Files" on page 350 and "Backup and Recovery Scenarios" on page 352 for examples of using the RESTORE STGPOOL command.

What Happens When a Storage Pool Is Restored

When you restore a storage pool, ADSM determines which files are in the storage pool being restored, according to the ADSM database. Using file copies from a copy storage pool, ADSM restores the files that were in the storage pool to the same or a different storage pool.

Note: Cached copies of files are never restored. References to any cached files that have been identified as having data-integrity errors or that reside on a *destroyed* volume will be removed from the database during restore processing.

The RESTORE STGPOOL command with the PREVIEW=YES parameter can be used to identify volumes that contain damaged primary files. During restore processing, a message is issued for every volume in the restored storage pool that contains damaged, non-cached files. To identify the specific files that are damaged on these volumes, use the QUERY CONTENT command.

After the files are restored, the old references to these files in the primary storage pool are deleted from the database. This means that ADSM now locates these files on the volumes to which they were restored, rather than on the volumes on which they were previously stored. If a destroyed volume becomes empty because all files have been restored to other locations, the destroyed volume is automatically deleted from the database.

The RESTORE STGPOOL command generates a background process that can be canceled with the CANCEL PROCESS command. If a RESTORE STGPOOL background process is canceled, some files may have already been restored prior to the cancellation. To display information about background processes, use the QUERY PROCESS command.

Restoring Files to a Storage Pool with Collocation

When restoring to a primary storage pool that has collocation enabled, the server restores files by client node and client file space. This process preserves the collocation of client files. However, if the copy storage pool being used to restore files does not have collocation enabled, restore processing can be slow.

If you need to use a copy storage pool that is not collocated to restore files to a primary storage pool that is collocated, you can improve performance by:

1. Restoring the files first to a random access storage pool (on disk).
2. Allowing or forcing the files to migrate to the target primary storage pool.

   For the random access pool, set the target storage pool as the next storage pool. Adjust the migration threshold to control when migration occurs to the target storage pool.

When a Storage Pool Restoration is Incomplete

The restoration of a storage pool volume may be incomplete. Use the QUERY CONTENT command to get more information on the remaining files on volumes for which restoration was incomplete. The restoration may be incomplete for one or more of the following reasons:

- Files were either never backed up or the backup copies are marked as damaged.
- A copy storage pool was specified on the RESTORE command, but files were backed up to a different copy storage pool. If you suspect this is a problem, use the RESTORE command again without specifying a copy storage pool from which to restore files. The PREVIEW option can be used on the second RESTORE command, if you do not actually want to restore files.
- Volumes in the copy storage pool needed to perform the restore operation are offsite or unavailable. Check the activity log for messages that occurred during restore processing.

- Backup file copies in copy storage pools were moved or deleted by other ADSM processes during restore processing. To prevent this problem, do not issue the following commands for copy storage pool volumes while restore processing is in progress:
  - MOVE DATA
  - DELETE VOLUME (DISCARDDATA=YES)
  - AUDIT VOLUME (FIX=YES)

  Also, you can prevent reclamation processing for your copy storage pools by setting the RECLAIM parameter to 100 with the UPDATE STGPOOL command.

Chapter 9. Managing Storage Pool Volumes

The sections listed in the following table begin at the indicated pages.

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, “Interface Cross-Reference” on page 467 lists each command and shows if its function is also available on the administrative client GUI.

© Copyright IBM Corp. 1995, 1997

Storage Pool Volumes

Volumes in storage pools contain backed up, archived, or space-managed data from clients. Storage pools are either random access or sequential access, depending on the device type of the device class to which the pool is assigned.

Random access storage pools are always associated with the DISK device class, and all volumes are one of the following:

- Fixed-size files on a disk, which must be created by using the DSMFMT utility before the ADSM server can access them
- Raw partitions on a disk, which must be defined before the server can access them

See “Preparing Volumes for Random Access Storage Pools” on page 161 for details.

Each volume defined in a sequential access storage pool must be of the same type as the device type of the associated device class. The device types are:

3570	A volume is a 3570 tape cartridge.
3590	A volume is a 3590 tape cartridge.
4MM	A volume is a 4mm tape cartridge.
8MM	A volume is an 8mm tape cartridge.
CARTRIDGE	A volume is a 3490 cartridge system tape.
DLT	A volume is a digital linear tape.
FILE	A volume is a file in the file system of the server machine.
OPTICAL	A volume is a two-sided 5.25 inch rewritable optical cartridge.
WORM	A volume is a two-sided 5.25 inch write-once optical cartridge.
WORM12	A volume is a two-sided 12-inch write-once optical cartridge.
WORM14	A volume is a two-sided 14-inch write-once optical cartridge.

See “Preparing Volumes for Sequential Access Storage Pools” on page 162.

Access Modes for Storage Pool Volumes

Access to any volume in a storage pool is determined by the access mode assigned to that volume. You can change the access mode of a volume. The ADSM server can also change the access mode based on what happens when it tries to access a volume. For example, if the server cannot write to a read/write access volume, the server automatically changes the access mode to read-only.

The access modes are:

Read/write Allows files to be read from or written to a volume in the storage pool.

If the server cannot write to a read/write access volume, the server automatically changes the access mode to read-only.

Read-only Allows files to be read from but not written to a disk or tape volume.

Unavailable Specifies that the volume is not available for any type of access by the ADSM server.

Destroyed Specifies that a primary storage pool volume has been permanently damaged. Neither users nor system processes (like migration) can access files stored on the volume.

This access mode is used to indicate an entire volume that should be restored using the RESTORE STGPOOL or RESTORE VOLUME command. After all files on a destroyed volume are restored to other volumes, the destroyed volume is automatically deleted from the database.

Only volumes in primary storage pools can be updated to destroyed.

If you update a random access storage pool volume to destroyed, you cannot vary the volume online. If you update a sequential access storage pool volume to destroyed, ADSM does not attempt to mount the volume.

If a volume contains no files and the UPDATE VOLUME command is used to change the access mode to destroyed, the volume is deleted from the database.

Offsite Specifies that a copy storage pool volume is at an offsite location and therefore cannot be mounted. Use this mode to help you track volumes that are offsite. ADSM treats offsite volumes differently, as follows:

- Mount requests are not generated for offsite volumes
- Data can be reclaimed or moved from offsite volumes by retrieving files from other storage pools
- Empty, offsite scratch volumes are not deleted from the copy storage pool

Only volumes in a copy storage pool can be updated to offsite.

Preparing Volumes for Random Access Storage Pools

Prepare a volume for use in a random access storage pool by performing the following steps:

1 Create and format the volume.

Use the DSMFMT utility from the operating system command line. For example, create a volume named stgvol.001 with a size of 21MB for use in a random access storage pool, by entering:

```
> dsmfmt -m stgvol.001 21
```

Note: For performance reasons, allocate storage pool volumes on disk drives that reside on the ADSM server machine, not on remotely mounted file systems.

2 Define the disk storage volume to ADSM.

Use the administrative client graphical user interface or enter the DEFINE VOLUME command from the ADSM administrative client command line. This command informs the server of the name of the new volume that can be used to store client data. See "Defining Storage Pool Volumes" on page 163.

Note: When you run the DSMFMT utility, you create a standard file. The name of this file is the name used for the storage volume when it is defined to the server.

Another option for preparing a volume is to create a raw partition on a formatted disk. For details, see the operating system's format command. After the partition is created, you define the volume to ADSM.

Attention: ADSM cannot always detect if data already exists on a raw partition. Existing data can be destroyed if you define a volume to that partition.

Preparing Volumes for Sequential Access Storage Pools

For sequential access storage pools with other than FILE device type, you must prepare volumes for use. When the server accesses a sequential access volume, it checks the volume name in the header to ensure that the correct volume is being accessed. To prepare a volume:

1 Label the volume. Any volumes associated with the following device types must be labeled before the server can use them:

- 3570
- 3590
- 4MM
- 8MM
- CARTRIDGE
- DLT
- OPTICAL
- WORM
- WORM12
- WORM14

See "Labeling Sequential Access Volumes" on page 68.

2 For storage pools in automated libraries, use the CHECKIN LIBVOLUME command to check the volume into the library. See “Informing the Server about New Volumes in a Library” on page 71.

3 You can skip this step if you allowed scratch volumes in the storage pool by specifying a nonzero MAXSCRATCH parameter.

If you have not allowed scratch volumes in the storage pool, identify the volume, by name, to the ADSM server so that it can be accessed later. For details, see “Defining Storage Pool Volumes.”

Defining Storage Pool Volumes

Task	Required Privilege Class
Define volumes in any storage pool	System or unrestricted storage
Define volumes in specific storage pools	System, unrestricted storage, or restricted storage for those pools

When you define a storage pool volume, you inform the server that the volume is available for storing backup, archive, or space-managed data.

For a random access storage pool, volumes must be predefined. For a sequential access storage pool, the ADSM server can use dynamically acquired scratch volumes, predefined volumes, or a combination.

To define a volume named VOL1 in the ENGBACK3 storage pool, enter:

```
define volume engback3 vol1
```

Using Scratch Volumes: You do not have to define volumes in sequential storage pools if you use the MAXSCRATCH parameter when you define or update the storage pool. Setting MAXSCRATCH to a nonzero value lets the storage pool dynamically acquire volumes as needed. ADSM automatically defines the volumes as they are acquired. The volumes are also automatically deleted from the storage pool when the server no longer needs them.

Before a sequential access scratch volume can be used, it must have a standard label. See “Preparing Volumes for Sequential Access Storage Pools” on page 162.

Updating Storage Pool Volumes

Task	Required Privilege Class
Update volumes	System or operator

You can update the attributes of a storage pool volume assigned to a primary or copy storage pool. Update a volume to:

- Reset any error state for a volume, by updating the volume to an access mode of read/write.
- Change the access mode of a volume, for example if a tape cartridge is moved offsite (offsite access mode) or damaged (destroyed access mode).
- Change the location for a volume in a sequential access storage pool.

When using the UPDATE VOLUME command, be prepared to supply some or all of the following information:

Volume name
Specifies the name of the storage pool volume to be updated. You can specify a group of volumes to update by using wildcard characters in the volume name, or by specifying the storage pool, device class, current access mode, or status of the volumes you want to update. See the parameters that follow.

New access mode
Specifies the new access mode for the volume (how users and system processes (like migration) can access files in the storage pool volume).

A random access volume must be varied offline before you can change its access mode to *unavailable* or *destroyed*. To vary a volume offline, use the VARY command.

If a scratch volume that is empty and has an access mode of offsite is updated so that the access mode is read/write, read-only, or unavailable, the volume will be deleted from the database.

Location
Specifies the location of the volume. This parameter can be specified only for volumes in sequential-access storage pools.

Storage pool
Restricts the update to volumes in the specified storage pool.

Device class
Restricts the update to volumes in the specified device class.

Current access mode
Restricts the update to volumes that currently have the specified access mode.

Status
Restricts the update to volumes with the specified status (online, offline, empty, pending, filling, or full).

Preview
Specifies whether you want to preview the update operation without actually performing the update.

An example of when to use the UPDATE VOLUME command is if you accidentally damage a volume, VOL1. You can change the access mode to unavailable so that

ADSM does not try to write or read data from the volume. Enter the following command:

```
update volume vol1 access=unavailable
```

Monitoring the Use of Storage Pool Volumes

Task	Required Privilege Class
Display information about volumes	Any administrator

You can query the server for general information about storage pool volumes, or you can view a detailed report to evaluate:

- Current access mode and status of the volume
- Amount of available space on the volume
- Amount of reclaimable space on a sequential access volume
- Location
- Contents of a storage pool volume (user files on the volume)

Requesting General Information about Storage Pool Volumes

To query the server for general information about all volumes defined to the server, enter:

```
query volume
```

The following shows an example of the output of this standard query. The example illustrates that data is being stored on the 8mm tape volume named ADSM01, as well as on several other volumes in various storage pools.

```
Volume Name               Storage      Device      Estimated  %Util   Volume
                          Pool Name    Class Name   Capacity           Status
                                                        (MB)
------------------------  -----------  ----------  ---------  -----  --------
/dev/raixvol1             AIXPOOL1     DISK            240.0   26.3  On-Line
/dev/raixvol2             AIXPOOL2     DISK            240.0   36.9  On-Line
/dev/rdosvol1             DOSPOOL1     DISK            240.0   72.2  On-Line
/dev/rdosvol2             DOSPOOL2     DISK            240.0   74.1  On-Line
/dev/ros2vol1             OS2POOL1     DISK            240.0   55.7  On-Line
/dev/ros2vol2             OS2POOL2     DISK            240.0   51.0  On-Line
ADSM00                    TAPEPOOL     TAPE8MM       2,472.0    0.0  Filling
ADSM01                    TAPEPOOL     TAPE8MM       2,472.0    2.2  Filling
```

Requesting Detailed Information about Storage Pool Volumes

To query the server for a detailed report on volume ADSM01 in the storage pool named TAPEPOOL, enter:

```
query volume adsm01 format=detailed
```

The following shows the output of this detailed query.

```
                   Volume Name: ADSM01
             Storage Pool Name: TAPEPOOL
             Device Class Name: TAPE8MM
       Estimated Capacity (MB): 2,472.0
                         %Util: 26.3
                 Volume Status: Filling
                        Access: Read/Write
        Pct. Reclaimable Space: 5.3
               Scratch Volume?: No
               In Error State?: No
      Number of Writable Sides: 1
       Number of Times Mounted: 4
             Write Pass Number: 2
     Approx. Date Last Written: 12/04/1995 11:33:26
        Approx. Date Last Read: 12/03/1995 16:42:55
           Date Became Pending:
        Number of Write Errors: 0
         Number of Read Errors: 0
               Volume Location:
Last Update by (administrator): ADSMADMIN
         Last Update Date/Time: 12/02/1995 13:20:14
```

Use this report to:

- Ensure that the volume is available for use.

  Check the *volume status* to see if a disk volume has been varied offline, or if a sequential access volume is currently being filled with data.

  Check the *access mode* to determine whether files can be read from or written to this volume.

- Monitor the use of storage space.

  The *estimated capacity* is determined by the device class associated with the storage pool to which this volume belongs. Based on the estimated capacity, the system tracks the percentage of space occupied by client files. In this example, 26.3% of the estimated capacity is currently in use.

- Monitor the life of a sequential access volume.

In this example, ADSM01 is not a scratch volume, which means that it will be reused by the TAPEPOOL storage pool after space has been reclaimed or deleted from the volume.

The *write pass number* indicates the number of times the volume has been written to, starting from the beginning of the volume. A value of one indicates that a volume is being used for the first time. In this example, ADSM01 has a write pass number of two, which indicates space on this volume may have been reclaimed or deleted once before. Be sure to compare this value to the specifications provided with the media that you are using. In particular, the manufacturer recommendations for the maximum number of write passes for some types of tape media may require that you retire your tape volumes after reaching the limit in order to ensure the integrity of your data. To retire volumes, move the data off the volume by using the MOVE DATA command. See “Moving Files from One Volume to Another Volume” on page 174.

Use the *number of times mounted* and the *approximate date last written to or read from* to help you estimate the life of the volume. For example, if more than six months have passed since the last time this volume has been written to or read from, you should audit the volume to ensure that files can still be accessed. See “Auditing a Storage Pool Volume” on page 170 for information about auditing a volume.

- Monitor the error status of the volume.

  The server reports when the volume is in an error state and automatically updates the access mode of the volume to read-only. The *number of write errors* and *number of read errors* indicate the type and severity of the problem. Audit a volume when it is placed in error state. See “Auditing a Storage Pool Volume” on page 170 for information about auditing a volume.

- Determine the location of a volume in a sequential access storage pool.

  When you define or update a sequential access volume, you can give location information for the volume. The detailed query displays this location name. The location information can be useful to help you track volumes, for example, offsite volumes in copy storage pools.

- Determine when the state of a volume in a sequential access storage pool became *pending*.

  A sequential access volume is placed in the pending state after the last file is deleted or moved from the volume. All the files that pending volumes had contained were expired or deleted, or were moved from the volume. Volumes remain in the pending state for as long as specified with the REUSEDELAY parameter for the storage pool to which the volume belongs.

Requesting Information about Storage Pool Volume Contents

Any administrator can request information about the contents of a storage pool volume. Viewing the contents of a storage volume is useful when a volume is damaged or before you:

- Request the server to correct any inconsistencies

- Move files from one volume to other volumes
- Delete a volume from a storage pool

Because ADSM tracks the contents of a storage volume through its database, the requested volume need not be accessed in order to determine its contents.

The report generated by a QUERY CONTENT command shows the contents of a volume. This report can be extremely large and may take a long time to produce. To reduce the size of this report, narrow your search by selecting one or all of the following search criteria:

Node name
Name of the node whose files you want to include in the query.

File space name
Remember that file space names are case-sensitive and must be entered exactly as they are known to the server. Use the QUERY FILESPACE command to find out the correct capitalization.

Number of files to be displayed
Enter a positive integer, such as 10, to list the first ten files stored on the volume. Enter a negative integer, such as -15, to list the last fifteen files stored on the volume.

Filetype
Specifies which types of files, that is, backup versions, archive copies, or space-managed files, or a combination of these.

Format of how the information is displayed
Standard or detailed information for the specified volume.

Damaged
Specifies whether to restrict the query output either to files that are known to be damaged, or to files that are not known to be damaged.

Copied
Specifies whether to restrict the query output to either files that are backed up to a copy storage pool, or to files that are not backed up to a copy storage pool.

Viewing a Standard Report on the Contents of a Volume

To view the first seven backup files on volume ADSM01 from file space /usr on client node TOMC, for example, enter:

```
query content adsm01 node=tomc filespace=/usr count=7 type=backup
```

Figure 33 on page 169 displays a standard report which shows the first seven files from file space /usr on TOMC stored in ADSM01.

```
Node Name                Type Filespace  Client's Name for File
                              Name
------------------------ ---- ---------- --------------------------------------
TOMC                     Bkup /usr       /bin/ acctcom
TOMC                     Bkup /usr       /bin/ acledit
TOMC                     Bkup /usr       /bin/ aclput
TOMC                     Bkup /usr       /bin/ admin
TOMC                     Bkup /usr       /bin/ ar
TOMC                     Bkup /usr       /bin/ arcv
TOMC                     Bkup /usr       /bin/ banner
```

Figure 33. A Standard Report on the Contents of a Volume

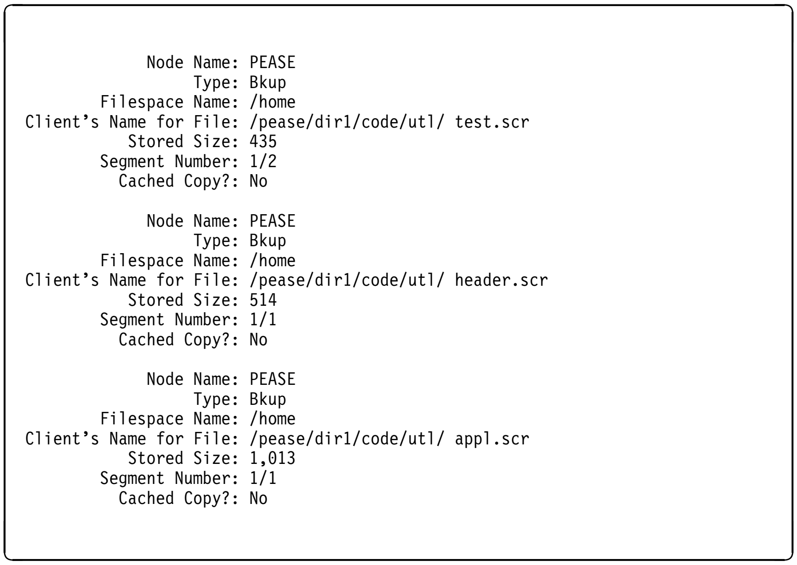
Viewing a Detailed Report on the Contents of a Volume

To query the server to display detailed information about the last three files stored on volume VOL1, enter:

```
query content vol1 count=-3 format=detailed
```

Figure 34 on page 170 displays a detailed report that shows the last three files, in reverse order, stored on VOL1. For example, the *test.scr* file is the last file stored on the volume. The segment number, 1/2, identifies that this is the first volume on which *test.scr* resides. The file spans to a second tape volume.

For disk volumes, the *Cached copy?* field identifies whether the file is a cached copy of a file that has been migrated to the next storage pool in the hierarchy.



```
            Node Name: PEASE
                 Type: Bkup
       Filespace Name: /home
Client’s Name for File: /pease/dir1/code/utl/ test.scr
          Stored Size: 435
       Segment Number: 1/2
         Cached Copy?: No

            Node Name: PEASE
                 Type: Bkup
       Filespace Name: /home
Client’s Name for File: /pease/dir1/code/utl/ header.scr
          Stored Size: 514
       Segment Number: 1/1
         Cached Copy?: No

            Node Name: PEASE
                 Type: Bkup
       Filespace Name: /home
Client’s Name for File: /pease/dir1/code/utl/ appl.scr
          Stored Size: 1,013
       Segment Number: 1/1
         Cached Copy?: No
```

Figure 34. Viewing a Detailed Report of the Contents of a Volume

Auditing a Storage Pool Volume

Use this section to help you audit storage pool volumes for data integrity.

Task	Required Privilege Class
Audit volumes in storage pools over which they have authority	Restricted storage privilege
Audit a volume in any storage pool	System privilege, unrestricted storage privilege

The server database contains information about files on storage pool volumes. If there are inconsistencies between the information in the database and the files actually stored in a storage pool volume, users cannot access their files.

To ensure that all files are accessible on volumes in a storage pool, audit any volumes you suspect may have problems by using the AUDIT VOLUME command. You should audit a volume when:

- The volume is damaged
- The volume has not been accessed for a long period of time, for example, after six months
- A read or write error occurs while accessing the volume
- The database has been restored to an earlier point in time, and the volume is either a disk volume or a volume that was identified as being reused or deleted since the database backup took place

What Happens When You Audit Storage Pool Volumes

When you audit a volume, a background process is started. During the auditing process, the server:

- Records results of the audit in the activity log
- Sends informational messages about processing to the server console
- Prevents new files from being written to the volume

You can specify whether you want the server to correct the database if inconsistencies are detected (that is, to delete database records where inconsistencies are found). The system default is to report inconsistencies that are found, but to not correct the errors.

If files with integrity errors are detected, the handling of these files depends on the following:

- The type of storage pool to which the volume is assigned
- The FIX option of the AUDIT VOLUME command
- The location of file copies

To display the results of a volume audit after it has completed, use the QUERY ACTLOG command. See “Requesting Information from the Activity Log” on page 257.

Volumes in a Primary Storage Pool

For a volume in a primary storage pool, the values for the FIX parameter on an AUDIT VOLUME command have the following effects:

FIX=NO

ADSM reports, but does not delete, any database records that refer to files found with logical inconsistencies.

If the AUDIT VOLUME command detects a data-integrity error in a file:

- ADSM marks the file as *damaged* in the database. If a backup copy is stored in a copy storage pool, the file can be restored using the RESTORE VOLUME or RESTORE STGPOOL command.

- If the file is a cached copy, references to the file on this volume can be deleted by issuing the AUDIT VOLUME command and specifying FIX=YES.

If the AUDIT VOLUME command does not detect a data-integrity error in a file that had previously been marked as damaged, the state of the file is reset so that the file can be used. For example, if a dirty tape head caused some files to be marked damaged, you can clean the head and then audit the volume to make the files accessible again.

FIX=YES

ADSM fixes any inconsistencies as they are detected.

If the AUDIT VOLUME command detects a data-integrity error in a file:

- If a backup copy is not stored in a copy storage pool, ADSM deletes all database records that refer to the file.

- If a backup copy is stored in a copy storage pool, ADSM marks the file as damaged in the database. The file can then be restored using the RESTORE VOLUME or RESTORE STGPOOL command.
- If the file is a cached copy, ADSM deletes the database records that refer to the cached file. The primary file is stored on another volume.

If the AUDIT VOLUME command does not detect a data-integrity error in a file that had previously been marked as damaged, ADSM resets the state of the file so that it can be used. For example, if a dirty tape head caused some files to be marked damaged, you can clean the head and then audit the volume to make the files accessible again.

Volumes in a Copy Storage Pool

For volumes in a copy storage pool, the values for the FIX parameter on an AUDIT VOLUME command have the following effects:

FIX=NO

The error is reported and the file copy marked as *damaged* in the database.

FIX=YES

References to the file on the audited volume are deleted from the database.

Auditing a Volume in a Disk Storage Pool

For example, to audit the /dev/vol1 disk volume and have only summary messages sent to the activity log and server console, enter:

```
audit volume /dev/vol1 quiet=yes
```

The audit volume process is run in the background and the server returns the following message:

```
ANR2313I Audit Volume NOFIX process started for volume /dev/vol1
(process id 4).
```

To view the status of the audit volume process, enter:

```
query process
```

The following figure displays an example of the audit volume process report.

```
 Process Process Description      Status
  Number
-------- ------------------------ ---------------------------------------------
       4 Audit Volume             Storage Pool BACKUPPOOL, Volume
           (Inspect Only)          /dev/vol1, Files Processed: 680,
                                   Irretrievable Files Found: 0, Partial Files
                                   Skipped: 0
```

To display the results of a volume audit after it has completed, you can issue the QUERY ACTLOG command.

Auditing Multiple Volumes in a Sequential Access Storage Pool

When you audit a sequential storage volume containing files that span multiple volumes, the server selects all associated volumes and begins the audit process with the first volume on which the first file resides. For example, Figure 35 shows five volumes defined to ENGBACK2. In this example, File A spans VOL1 and VOL2, and File D spans VOL2, VOL3, VOL4, and VOL5.

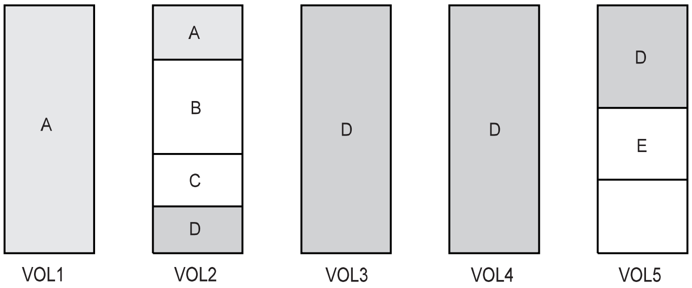


Figure 35. Tape Volumes with Files A, B, C, D, and E

If you request that the server audit volume VOL3, the server first accesses volume VOL2, because File D begins at VOL2. When volume VOL2 is accessed, the server *only* audits File D. It does not audit the other files on this volume.

Because File D spans multiple volumes, the server accesses volumes VOL2, VOL3, VOL4, and VOL5 to ensure that there are no inconsistencies between the database and the storage pool volumes.

For volumes that require manual mount and dismount operations, the audit process can require significant manual intervention.

Auditing a Single Volume in a Sequential Access Storage Pool

To audit a single volume in a sequential storage pool, you can request that the server skip any files that span from the single volume to other volumes in the storage pool. This option is useful when the volume you want to audit contains part of a file, the rest of which resides on a different, damaged volume.

For example, to audit only volume VOL5 in the example in Figure 35 and have the server fix any inconsistencies found between the database and the storage volume, enter:

```
audit volume vol5 fix=yes skippartial=yes
```

Moving Files from One Volume to Another Volume

You can move files from one volume to another volume in the same or a different storage pool. The volumes can be onsite volumes or offsite volumes. During normal operations, you do not need to move data. You might need to move data in some situations, for example, when you need to salvage any readable data from a damaged ADSM volume.

During the data movement process, the server:

- Moves any readable files to available volumes in the specified destination storage pool
- Deletes any cached copies from a disk volume
- Attempts to bypass any files that previously have been marked as damaged

During the data movement process, users cannot access the volume to restore or retrieve files, and no new files can be written to the volume.

Note: Files in a copy storage pool do not move when primary files are moved.

Task	Required Privilege Class
Move files from a volume in any storage pool to an available volume in any storage pool	System or unrestricted storage
Move files from one volume to an available volume in any storage pool to which you are authorized	Restricted storage

Moving Data to Other Volumes in the Same Storage Pool

Moving files from one volume to other volumes in the same storage pool is useful:

- When you want to free up all space on a volume so that it can be deleted from the ADSM server

  See “Deleting Storage Pool Volumes” on page 178 for information about deleting backed up, archived, or space-managed data before you delete a volume from a storage pool.
- To salvage readable files from a volume that has been damaged
- When you want to delete cached files from disk volumes

If you want to force the removal of cached files, you can delete them by moving data from one volume to another volume. During the move process, ADSM deletes cached files remaining on disk volumes.

If you move data between volumes within the same storage pool and you run out of space in the storage pool before all data is moved from the target volume, then you cannot move all the data from the target volume. In this case, consider moving data to available space in another storage pool as described in “Moving Data to Another Storage Pool.”

Moving Data to Another Storage Pool

You can move all data from a volume in one storage pool to volumes in another storage pool. You might do this, for example, when you have only one tape drive in a library and you want to manually reclaim tape volumes. When you specify a target storage pool that is different than the source storage pool, ADSM uses the storage hierarchy to move data if more space is required.

Note: Data cannot be moved from a primary storage pool to a copy storage pool. Data in a copy storage pool cannot be moved to any other storage pool.

You can move data from random access storage pools to sequential access storage pools. For example, if you have a damaged disk volume and you have a limited amount of disk storage space, you could move all files from the disk volume to a tape storage pool. Moving files from a disk volume to a sequential storage pool may require many volume mount operations if the target storage pool is collocated. Ensure that you have sufficient personnel and media to move files from disk to sequential storage.

Moving Data from an Offsite Volume in a Copy Storage Pool

You can move data from offsite volumes without bringing the volume onsite. Processing of the MOVE DATA command for primary storage pool volumes does not affect copy storage pool files.

Processing of the MOVE DATA command for volumes in copy storage pools is similar to that of primary storage pools, with the following exceptions:

- Most volumes in copy storage pools may be set to an access mode of *offsite*, making them ineligible to be mounted. During processing of the MOVE DATA command, valid files on offsite volumes are copied from the original files in the primary storage pools. In this way, valid files on offsite volumes are copied without having to mount these volumes. These new copies of the files are written to another volume in the copy storage pool.
- With the MOVE DATA command, you can move data from any primary storage pool volume to any primary storage pool. However, you can move data from a copy storage pool volume *only* to another volume within the same copy storage pool.

When you move files from a volume marked as offsite, ADSM:

1. Determines which files are still active on the volume from which you are moving data
2. Obtains these files from a primary storage pool or from another copy storage pool
3. Copies the files to one or more volumes in the destination copy storage pool

Procedure for Moving Data

1 Before you move files from a volume, complete the following steps:

a. If you want to ensure that no new files are written to a volume after you move data from it, change the volume's access mode to read-only. This prevents the server from filling the volume with data again as soon as data is moved. You might want to do this if you want to delete the volume.

   See "Updating Storage Pool Volumes" on page 163 for information about updating the access mode of a storage pool volume.

b. Ensure sufficient space is available on volumes within the specified destination storage pool by:

   1) Querying the source storage volume to determine how much space is required on other volumes. See "Monitoring the Use of Storage Pool Volumes" on page 165 for information about requesting information about a storage volume.

   2) Querying the specified destination storage pool to ensure there is sufficient capacity to store the files being moved. See "Monitoring the Use of Storage Pool Space" on page 145 for information about querying a storage pool.

c. If you need more storage space, define volumes or increase the maximum number of scratch volumes in the specified destination storage pool.

   See "Defining Storage Pool Volumes" on page 163 for preparing volumes to be used for server storage.

d. If you are moving files from a volume in a sequential storage pool to another volume in the same storage pool, ensure that the mount limit of the device class associated with the storage pool is greater than one.

   See "Requesting Information about a Device Class" on page 102 for requesting information about the mount limit value for the device class.

e. If you are moving files from a tape volume to a tape storage pool, ensure that the two tape drives required are available.

2 Move the data using the MOVE DATA command.

For example, to move the files stored in the /dev/vol3 volume to any available volume in the STGTMP1 storage pool, enter:

```
move data /dev/vol3 stgpool=stgtmp1
```

When you move data from a volume, the server starts a background process and sends informational messages, such as:

```
ANR1140I Move Data process started for volume /dev/vol3
(process ID 32).
```

Requesting Information about the Data Movement Process

To request information on the data movement process, enter:

```
query process
```

Figure 36 shows an example of the report that you receive about the data movement process.

```
 Process Process Description  Status
  Number
-------- -------------------- -------------------------------------------------
      32 Move Data            Volume /dev/vol3, (storage pool BACKUPPOOL),
                               Target Pool STGTMP1, Moved Files: 49, Moved
                               Bytes: 9,121,792, Unreadable Files: 0,
                               Unreadable Bytes: 0. Current File (bytes):
                               3,522,560

                               Current output volume: VOL1.
```

Figure 36. Information on the Data Movement Process

Monitoring the Movement of Data between Volumes

You can query the server for volume information to monitor the movement of data between volumes. For example, to see how much data has moved from the source volume in the move operation example, enter:

```
query volume /dev/vol3 stgpool=backuppool
```

Near the beginning of the move process, querying the volume from which data is being moved gives the following results:

```
Volume Name         Storage       Device         Estimated    %Util     Volume
                    Pool Name     Class Name      Capacity              Status
                                                      (MB)
---------------     -----------   ----------     ---------    -----    --------
/dev/vol3           BACKUPPOOL    DISK                15.0     59.9    On-Line
```

Querying the volume to which data is being moved (VOL1, according to the process query output) gives the following results:

```
Volume Name         Storage       Device         Estimated    %Util     Volume
                    Pool Name     Class Name      Capacity              Status
                                                      (MB)
----------------    -----------   ----------     ---------    -----    --------
VOL1                STGTMP1       8500DEV          4,944.0      0.3    Filling
```

At the end of the move process, querying the volume from which data was moved gives the following results:

```
Volume Name         Storage       Device         Estimated    %Util     Volume
                    Pool Name     Class Name      Capacity              Status
                                                      (MB)
----------------    ----------    ----------     ---------    -----    --------
/dev/vol3           BACKUPPOOL    DISK                15.0      0.0    On-Line
```

Deleting Storage Pool Volumes

You can delete volumes, and optionally the client files they contain, from either primary or copy storage pools.

If files that are not cached are deleted from a primary storage pool volume, any copies of these files in copy storage pools will also be deleted.

Files in a copy storage pool are never deleted unless:

- The volume that contains the copy file is deleted by using the DISCARDDATA=YES option.
- A data-integrity error is detected by using AUDIT VOLUME with the FIX=YES option for a copy storage pool volume.
- The primary file is deleted because of:
  - Policy-based file expiration
  - File space deletion
  - Deletion of the primary storage pool volume

Usage tip: If you are deleting many volumes, delete the volumes one at a time. Concurrently deleting many volumes can adversely affect server performance.

Task	Required Privilege Class
Delete volumes from any storage pool	System or unrestricted storage
Delete volumes from storage pools over which they have authority	Restricted storage

Deleting an Empty Storage Pool Volume

You can delete empty storage pool volumes. For example, to delete an empty volume named ADSM03, enter:

```
delete volume adsm03
```

On an administrative client, you will receive the following confirmation messages, unless the client is running with the NOCONFIRM option:

```
ANR2200W  This command will delete volume ADSM03
from its storage pool after verifying that the volume
contains no data.
Do you wish to proceed? (Y/N)
```

After you respond yes, the server generates a background process to delete the volume.

Deleting a Storage Pool Volume with Data

To prevent you from accidentally deleting backed up, archived, or space-managed files from server storage, the server does not allow you to delete a volume that contains user data unless you specify DISCARDDATA=YES on the DELETE VOLUME command.

For example, to discard all data from volume ADSM03 and delete the volume from its storage pool, enter:

```
delete volume adsm03 discarddata=yes
```

The server generates a background process and deletes data in a series of batch database transactions. After all files have been deleted from the volume, the server deletes the volume from the storage pool. If the volume deletion process is canceled or if a system failure occurs, the volume might still contain data. Reissue the DELETE VOLUME command and explicitly request the server to discard the remaining files on the volume.

To delete a volume but not the files it contains, move the files to another volume. See “Moving Files from One Volume to Another Volume” on page 174 for information about moving data from one volume to another volume.

Residual data: Even after you move data, residual data may remain on the volume because of I/O errors or because of files that were previously marked as damaged. (ADSM does not move files that are marked as damaged.) To delete any volume that contains residual data that cannot be moved, you must explicitly specify that files should be discarded from the volume.

Restoring Storage Pool Volumes

An administrator can recreate files in primary storage pool volumes using copies in a copy storage pool by issuing the RESTORE VOLUME command.

Task	Required Privilege Class
Restore volumes in any storage pool for which they have authority	System, unrestricted storage, or restricted storage

Use the RESTORE VOLUME command to restore all files that are currently stored on one or more volumes in the same primary storage pool, and that were previously backed up to copy storage pools by using the BACKUP STGPOOL command.

When using the RESTORE VOLUME command, be prepared to supply some or all of the following information:

Volume name
Specifies the name of the volume in the primary storage pool for which to restore files.

Usage tip: To restore more than one volume in the same primary storage pool, issue this command once and specify a list of volumes to be restored. When you specify more than one volume, ADSM attempts to minimize volume mounts for the copy storage pool.

Copy storage pool name
Specifies the name of the copy pool from which the files are to be restored. This information is optional. If you do not specify a particular copy storage pool, ADSM restores the files from any copy storage pool where it can find them.

New storage pool
Specifies the name of the new primary storage pool to which to restore the files. This information is optional. If you do not specify a new storage pool, ADSM restores the files to the original primary storage pool.

Maximum number of processes
Specifies the maximum number of parallel processes that are used for restoring files.

Preview
Specifies whether you want to preview the restore operation without actually restoring data.

See “Recovering a Lost or Damaged Storage Pool Volume” on page 356 for an example of using the RESTORE VOLUME command.

What Happens When a Volume Is Restored

When you restore a volume, ADSM obtains a copy of each file that was on the volume from a copy storage pool, and then stores the files on a different volume.

Cached Files: Cached copies of files are never restored. References to any cached files that reside on a volume that is being restored are removed from the database during restore processing.

After files are restored, the old references to these files in the primary storage pool are deleted from the database. ADSM will now locate these files on the volumes to which they were restored, rather than on the volume on which they were previously stored.

This command changes the access mode of the volumes being restored to *destroyed*. When the restoration is complete (when all files on the volume are restored to other locations), the destroyed volume is empty and is then automatically deleted from the database.

The RESTORE VOLUME command generates a background process that can be canceled with the CANCEL PROCESS command. If a RESTORE VOLUME background process is canceled, some files may have already been restored prior to the cancellation. To display information on background processes, use the QUERY PROCESS command.

When a Volume Restoration is Incomplete

The restoration of a volume may be incomplete. Use the QUERY CONTENT command to get more information on the remaining files on volumes for which restoration was incomplete. The restoration may be incomplete for one or more of the following reasons:

- Files were either never backed up or the backup copies are marked as damaged.
- A copy storage pool was specified on the RESTORE command, but files were backed up to a different copy storage pool. If you suspect this is a problem, use the RESTORE command again without specifying a copy storage pool from which to restore files. The PREVIEW option can be used on the second RESTORE command, if you do not actually want to restore files.
- Volumes in the copy storage pool needed to perform the restore operation are offsite or unavailable. Check the activity log for messages that occurred during restore processing.
- Backup file copies in copy storage pools were moved or deleted by other ADSM processes during restore processing. To prevent this problem, do not issue the following commands for copy storage pool volumes while restore processing is in progress:
  - MOVE DATA
  - DELETE VOLUME (DISCARDDATA=YES)

- AUDIT VOLUME (FIX=YES)

Also, you can prevent reclamation processing for your copy storage pools by setting the RECLAIM parameter to 100 with the UPDATE STGPOOL command.

Part 3. Policies

© Copyright IBM Corp. 1995, 1997

Chapter 10. Managing Policies

ADSM policies control how and when user files are backed up and archived to server storage and how user files are migrated to server storage.

The sections listed in the following table begin at the indicated pages.

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, “Interface Cross-Reference” on page 467 lists each command and shows if its function is also available on the administrative client GUI.

© Copyright IBM Corp. 1995, 1997

Operations Controlled by Policy

ADSM policies govern the following operations, which are discussed in this section:

- Backup and restore
- Archive and retrieve
- Client migration and recall

Backup and Restore

To guard against the loss of information, ADSM can copy files, subdirectories, and directories to media controlled by ADSM. Backups can be controlled by administrator-defined policies and schedules, or users can request backups of their own data. ADSM provides two types of backup:

Incremental backup
The backup of files according to policy defined in the backup copy group of the management class for the files. An incremental backup typically backs up all files that are new or that have changed since the last incremental backup.

Selective backup
Backs up only files that the user specifies. The files must also meet some of the policy requirements defined in the backup copy group.

When a user restores a backup version of a file, ADSM sends a copy of the file to the client node. The backup version remains in ADSM storage.

If more than one backup version exists, a user can restore the active backup version of the file or any inactive backup versions.

Archive and Retrieve

To preserve files for later use or for records, a user can request ADSM to copy files, subdirectories, and directories for long-term storage on media controlled by ADSM. When users archive files, they can choose to have ADSM erase the original files from their workstation after the files are archived.

When a user retrieves a file, ADSM sends a copy of the file to the client node. The archived file remains in ADSM storage.

Migration and Recall

If the Hierarchical Storage Management (HSM) feature of ADSM is activated on a client node, users can migrate files from client node storage to server storage and recall files to the client node as needed. HSM frees space on client nodes for new data and makes more efficient use of your storage.

Files that are migrated and recalled with the HSM client are also called *space-managed* files.

For details about using HSM on clients, see *ADSM Using the UNIX HSM Clients*.

Migration

When a file is migrated to the server, it is replaced on the client node with a small stub file of the same name as the original file. The stub file contains data needed to locate the migrated file on server storage.

ADSM provides selective and automatic migration. Selective migration lets users migrate files by name. The two types of automatic migration are:

Threshold If space usage exceeds a high threshold set at the client node, migration begins and continues until usage drops to the low threshold also set at the client node.

Demand If an out-of-space condition occurs for a client node, migration begins and continues until usage drops to the low threshold.

To prepare for efficient automatic migration, ADSM copies a percentage of user files from the client node to the server. The *premigration* process occurs whenever ADSM completes an automatic migration. The next time free space is needed at the client node, the files that have been premigrated to the server can quickly be changed to stub files on the client. The default premigration percentage is the difference between the high and low thresholds.

Files are selected for automatic migration and premigration based on the number of days since the file was last accessed and also on other factors set at the client node.

Recall

ADSM provides selective and transparent recall. Selective recall lets users recall files by name. Transparent recall occurs automatically when a user accesses a migrated file.

Reconciliation

Migration and premigration can create inconsistencies between client node and server storage. For example, if a user deletes a migrated file from the client node, the copy remains at the server. At regular intervals set at the client node, ADSM compares client node and server storage and reconciles the two by deleting from the server any outdated files or files that do not exist at the client node.

Policy Objects

Policy administrators specify how files are backed up, archived, migrated from client node storage, and managed in ADSM storage through ADSM policy objects. These objects implement ADSM policies. Figure 37 on page 188 shows the objects and their relationships.

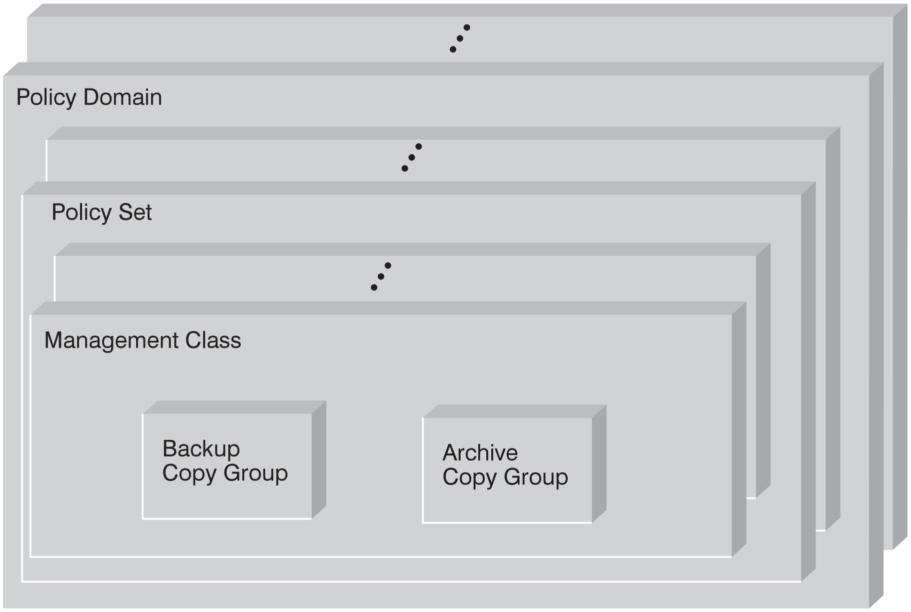


Figure 37. ADSM Policy Objects

Backup copy group

Controls how ADSM performs backup processing of files associated with the management class. A backup copy group determines the following:

- If a file is backed up (even if it has not changed since the last backup)
- How many days must elapse before a file can be backed up again
- How to handle files that are in use during backup
- Where the server stores backup versions of files and directories
- How many backup versions the server keeps of files and directories
- How long the server keeps backup versions of files and directories

Archive copy group

Controls how ADSM performs archive processing of files associated with the management class. An archive copy group determines the following:

- How to handle files that are in use during archive
- Where the server stores archived copies of files
- How long the server keeps archived copies of files

Management class

Associates backup and archive groups with files and specifies if and how client node files are migrated to storage pools. A management class can contain one backup copy group, one archive copy group, both a backup and archive copy group, or no copy groups. Users can *bind* (that is, associate) their files to a management class through the include-exclude list.

Policy set

Specifies the management classes that are available to groups of users. Policy sets contain one or more management classes: a *default management class* and any number of additional management classes.

Policy domain

Lets an administrator group client nodes by the policies that govern their files and by the administrators who manage their policies. A policy domain contains one or more policy sets, but only one policy set (named ACTIVE) can be active at a time. ADSM uses the active policy set to manage files for client nodes assigned to a policy domain.

You can use policy domains to:

- Provide default storage management policies
- Group client nodes with similar storage management requirements
- Direct files from different groups of clients to different storage hierarchies based on need (different file destinations with different storage characteristics)
- Restrict the number of management classes to which clients have access

Management Classes

Each client node is assigned to a single policy domain, and the client node has access only to the management classes contained in the domain. The management classes specify whether client files are migrated to storage pools (hierarchical storage management). The copy groups in these management classes specify the number of backup versions retained in ADSM storage and the length of time to retain backup versions and archive copies.

For example, if a group of users needs only one backup version of their files, you can create a policy domain that contains only one management class whose backup copy group allows only one backup version. Then you can assign the client nodes for these users to the policy domain. See “Administrator Registration of Client Nodes” on page 294 for information on registering client nodes and assigning policy domains to them.

Management Class Configuration

Before defining a management class, consider whether the management class should contain:

A backup copy group and an archive copy group

For example, most users need to back up and archive documents, spread sheets, and graphics.

A backup copy group only

For example, some users only want to back up application files (such as database, log, or history files that change daily).

An archive copy group only

A management class that contains only an archive copy group is useful for users who create:

- Point-in-time files. For example, an engineer can archive the design of an electronic component and the software that created the design. Later, the engineer can use the design as a base for a new electronic component.
- Files that are rarely used but need to be retained for a long time. A client can erase the original file without affecting how long the archive copy is retained in ADSM storage. Examples include legal records, patient records, and tax forms.

Attention: A management class that contains neither a backup nor an archive copy group prevents a file from ever being backed up or archived. This type of management class is not recommended for most users. Use such a management class carefully to prevent users from mistakenly selecting it. If users bind their files to a management class without copy groups, ADSM issues warning messages.

Default Management Classes

Each policy set must include a default management class, which is used:

- To manage files that are not bound to a specific management class, as defined by the INCLUDE option in the include-exclude list.
- To manage existing backup versions when a management class name is deleted from the server as described in "How Files Are Associated with a Management Class" on page 192.
- To manage existing archive copies when a management class is deleted from the server. ADSM does not rebind archive copies but does use the archive copy group (if one exists) in the default management class.

A typical default management class should do the following:

- Meet the storage management needs for most of your users
- Contain both a backup copy group and an archive copy group
- Set serialization static or shared static to ensure the integrity of backed up and archived files
- Retain backup versions and archive copies for a sufficient amount of time
- Retain directories for at least as long as any files are associated with the directory

Other management classes can contain copy groups tailored either for the needs of special sets of users or for the needs of most users under special circumstances.

The Include-Exclude List

A user can define an include-exclude list to specify which files are eligible for backup services, which files can be migrated from the client (space-managed), and how ADSM manages backed up, archived, and space-managed files.

If a user does not create an include-exclude list:

- All files belonging to the user are eligible for backup services.
- The default management class governs backup, archive, and space-management policies.

With an include-exclude list, users can:

- Exclude files or directories from backup and client migration operations

  For example, Figure 38 shows that the SSTEINER node ID excludes all core files from being eligible for backup and client migration.

- Include any previously excluded files

  For example, Figure 38 shows that the files in the /home/ssteiner directory are excluded. The include statement that follows, however, means that the /home/ssteiner/options.scr file is eligible for backup and client migration.

- Bind a file to a specific management class

  For example, Figure 38 shows that all files and subdirectories belonging to the /home/ssteiner/driver5 directory are managed by the criteria defined in the MCENGBK2 management class.

```
exclude /.../core
exclude /home/ssteiner/*
include /home/ssteiner/options.scr
include /home/ssteiner/driver5/.../* mcengbk2
```

Figure 38. Example of an Include-Exclude List

ADSM processes the include-exclude list from the bottom up, and stops when it finds an include or exclude statement that matches the file it is processing. The order in which the include and exclude options are listed therefore affect which files are included and excluded. For example, suppose you switch the order of two lines in the example, as follows:

```
include /home/ssteiner/options.scr
exclude /home/ssteiner/*
```

The exclude statement comes last, and excludes all files in the /home/steiner directory. When ADSM is processing the include-exclude list for the options.scr file, it finds the exclude statement first. This time, the options.scr file is *excluded.*

For information on how to create an include-exclude list, see the user's publication for the appropriate client.

How Files Are Associated with a Management Class

Binding is the process of associating a file with a management class. The policies defined in the management class then apply to the bound files. Binding occurs when a file is backed up, archived, or migrated by the client.

- For backing up a file, a client can specify a management class in the client's include-exclude list (include-exclude options file for UNIX clients), or can accept the default management class.

  For directories, the client can specify a management class by using the DIRMC option in the client options file. If no management class is specified for a directory, ADSM chooses the management class with the longest retention period in the backup copy group (retention period for the only backup version).
- For archiving a file, the client can specify a management class in the client's include-exclude list, can specify a management class with the ARCHMC option on the archive command, or can accept the default management class.
- For migrating a file, a client can specify a management class in the client's include-exclude options file, or can accept the default management class.

The default management class is the management class identified as the default in the active policy set. If a client backs up, archives, and migrates a file to the same server, the management class specified for a file using an include-exclude option applies no matter what the operation (backup, archive, or migrate). If a client backs up and archives a file to one server, and migrates the file to a different server, the client can specify one management class for the file for backup and archive, and a different one for migrating. See the user's publication for the appropriate client for details.

A file remains bound to a management class name even if the attributes of the management class change. The following scenario illustrates this process:

1. A file named REPORT.TXT is bound to the default management class that contains a backup copy group specifying that up to three backup versions can be retained in server storage.
2. During the next week, three backup versions of REPORT.TXT are stored in ADSM storage. The active and two inactive backup versions are bound to the default management class.
3. The administrator assigns a new default management class that contains a backup copy group specifying only up to two backup versions.
4. The administrator then activates the policy set, and the new default management class takes effect.
5. REPORT.TXT is backed up again, bringing the number of versions to four. ADSM determines that according to the new backup copy group only two versions are to be retained. Therefore, ADSM marks the two oldest versions for deletion.
6. Expiration processing occurs (see “Expiration Processing” on page 193 for details). REPORT.TXT is still bound to the default management class, which now includes new retention criteria. Therefore, the two versions marked for deletion are purged, and one active and one inactive backup version remain in storage.

Rebinding Files to Management Classes

Rebinding is the process of associating a file with a new management class. Backup versions of files are rebound in the following cases:

- The user changes the management class specified in the include-exclude list and does a backup.
- An administrator activates a policy set in the same policy domain as the client node, and the policy set does not contain a management class with the same name as the management class to which a file is currently bound.
- An administrator assigns a client node to a different policy domain, and the active policy set in that policy domain does not have a management class with the same name.

Backup versions of a directory can be rebound when the user specifies a different management class using the DIRMC option in the client option file, and when the directory gets backed up.

If a file is bound to a management class that no longer exists, ADSM uses the default management class to manage the backup versions. When the user does another backup, ADSM rebinds the file and any backup versions to the default management class. If the default management class does not have a backup copy group, ADSM uses the backup retention grace period specified for the policy domain.

Note: Archive copies are never rebound because each archive operation creates a different archive copy. Archive copies remain bound to the management class name specified when the user archived them. If the management class no longer exists or no longer contains an archive copy group, ADSM uses the default management class. If the default management class does not contain an archive copy group, ADSM uses the archive retention grace period specified for the policy domain.

Expiration Processing

Backup and archive copy groups can specify the criteria that make copies of files eligible for deletion from server storage. However, even when a file becomes eligible for deletion, the file is not deleted until expiration processing occurs. If expiration processing does not occur periodically, storage pool space occupied by expired client files is not reused, and the ADSM server requires increased storage space.

See “Running Expiration Processing to Delete Expired Files” on page 213 for details about how to invoke expiration processing.

File Eligibility for Policy Operations

This section describes how ADSM selects files for the following operations:

- Full and partial incremental backups
- Selective backup
- Archive
- Migration from a client node (hierarchical storage management)

Incremental Backup

Clients can choose to back up their files using full or partial incremental backup. A full incremental backup ensures that clients' backed-up files are always managed according to policies. Clients should use full incremental backup whenever possible.

When a client uses partial incremental backup, only files that have changed since the last incremental backup are backed up. Attributes in the management class that would cause the file to be backed up when doing a full incremental backup are ignored. For example, unchanged files are not backed up even when they are assigned to a management class that specifies absolute mode and the frequency (minimum days between backups) specified has passed. The server also does less processing; for example, the server does not expire files or rebind management classes to files during a partial incremental backup. Because a partial incremental backup should complete more quickly and require less memory, however, clients may need to use it if the backup window is limited.

If clients must use partial incremental backups, they should periodically perform full incremental backups to ensure that complete backups are done and backup files are stored according to policies. For example, clients can do partial incremental backups every night during the week, and a full incremental backup on the weekend.

Full Incremental Backup

When a user requests a full incremental backup, ADSM performs the following steps to determine eligibility:

1. Checks each file against the user's include-exclude list:
   - Files that are excluded are not eligible for backup.
   - If files are not excluded and a management class is specified with the INCLUDE option, ADSM uses that management class.
   - If files are not excluded but a management class is not specified with the INCLUDE option, ADSM uses the default management class.
   - If no include-exclude list exists, all files in the client domain are eligible for backup, and ADSM uses the default management class.
2. Checks the management class of each included file:
   - If there is a backup copy group, ADSM goes to step 3.
   - If there is no backup copy group, the file is not eligible for backup.
3. Checks the *mode*, *frequency*, and *serialization* defined in the backup copy group.

   Mode Specifies whether the file is backed up only if it has changed since the last backup (*modified*) or whenever a backup is requested (*absolute*).

   Frequency Specifies the minimum number of days that must elapse between backups.

   Serialization Specifies how files are handled if they are modified while being backed up and what ADSM does if modification occurs.

- If the mode is *modified* and the minimum number of days have elapsed since the file was last backed up, ADSM determines if the file has been changed since it was last backed up:
  - If the file has been changed and the serialization requirement is met, the file is backed up.
  - If the file has not been changed, it is not backed up.
- If the mode is *modified* and the minimum number of days have not elapsed since the file was last backed up, the file is not eligible for backup.
- If the mode is *absolute*, the minimum number of days have elapsed since the file was last backed up, and the serialization requirement is met, the file is backed up.
- If the mode is *absolute* and the minimum number of days have not elapsed since the file was last backed up, the file is not eligible for backup.

Partial Incremental Backup

When a user requests a partial incremental backup, ADSM performs the following steps to determine eligibility:

1. Checks each file against the user's include-exclude list:
   - Files that are excluded are not eligible for backup.
   - If files are not excluded and a management class is specified with the INCLUDE option, ADSM uses that management class.
   - If files are not excluded but a management class is not specified with the INCLUDE option, ADSM uses the default management class.
   - If no include-exclude list exists, all files in the client domain are eligible for backup, and ADSM uses the default management class.
2. Checks the management class of each included file:
   - If there is a backup copy group, ADSM goes to step 3.
   - If there is no backup copy group, the file is not eligible for backup.
3. Checks the date and time of the last incremental backup by the client, and the *serialization* requirement defined in the backup copy group. (Serialization specifies how files are handled if they are modified while being backed up and what ADSM does if modification occurs.)
   - If the file has not changed since the last incremental backup, the file is not backed up.
   - If the file has changed since the last incremental backup and the serialization requirement is met, the file is backed up.

Selective Backup

When a user requests a selective backup, ADSM performs the following steps to determine eligibility:

1. Checks the file against any include or exclude statements contained in the user include-exclude list:
   - Files that are not excluded are eligible for backup. If a management class is specified with the INCLUDE option, ADSM uses that management class.
   - If no include-exclude list exists, the files selected are eligible for backup, and ADSM uses the default management class.
2. Checks the management class of each included file:
   - If the management class contains a backup copy group and the serialization requirement is met, the file is backed up. Serialization specifies how files are handled if they are modified while being backed up and what ADSM does if modification occurs.
   - If the management class does not contain a backup copy group, the file is not eligible for backup.

An important difference between selective backup and full incremental backup is that a file is backed up, without regard for whether the file has changed. This result may not always be what you want. For example, suppose a management class specifies to keep three backup versions of a file. If the client uses incremental backup, the file is backed up only when it changes, and the three versions in storage will be at different levels. If the client uses selective backup, the file is backed up regardless of whether it has changed. If the client uses selective backup on the file three times without changing the file, the three versions of the file in server storage are identical. Earlier, different versions are lost.

Archive

When a user requests the archiving of a file or a group of files, ADSM performs the following steps to determine eligibility:

1. Checks the files against the user's include-exclude list to see if any management classes are specified:
   - ADSM uses the default management class for files that are not bound to a management class.
   - If no include-exclude list exists, ADSM uses the default management class unless the user specifies another management class. See the user's publication for the appropriate client for details.
2. Checks the management class for each file to be archived.
   - If the management class contains an archive copy group and the serialization requirement is met, the file is archived. Serialization specifies how files are handled if they are modified while being archived and what ADSM does if modification occurs.

- If the management class does not contain an archive copy group, the file is not archived.

Automatic Migration from a Client Node

A file is eligible for automatic migration from a client node if it meets all of the following criteria:

- It resides on a node on which the root user has added and activated hierarchical storage management. It must also reside in a local file system to which the root user has added space management, and not in the root (/) or /tmp file system.
- It is not excluded from migration in the include-exclude list.
- It meets management class requirements for migration:
  - The file is not a character special file, a block special file, a FIFO special file (that is, a named pipe file) or a directory.
  - The file is assigned to a management class that calls for space management.
  - The management class calls for automatic migration after a specified number of days, and that time has elapsed.
  - A backup version of the file exists if the management class requires it.
  - The file is larger than the stub file that would replace it (plus one byte) or the file system block size, whichever is larger.

How Client Migration Works with Backup and Archive

As an administrator, you can define a management class that specifies automatic migration under certain conditions. For example, if the file has not been accessed for at least 30 days and a backup version exists, the file is migrated. You can also define a management class that allows users to selectively migrate whether or not a backup version exists. Users can also choose to archive files that have been migrated:

- If the file is backed up or archived to the server to which it was migrated, ADSM copies the file from the migration storage pool to the backup or archive storage pool. For a tape-to-tape operation, each storage pool must have a tape drive.
- If the file is backed up or archived to a different server, ADSM accesses the file by using the migrate-on-close recall mode. The file resides on the client node only until ADSM stores the backup version or the archived copy in the backup or archive storage pool.

When a client restores a backup version of a migrated file, ADSM deletes the migrated copy of the file from server storage the next time reconciliation is run.

When a client archives a file that is migrated and does not specify that the file is to be erased after it is archived, the migrated copy of the file remains in server storage. When a client archives a file that is migrated and specifies that the file is to be erased, ADSM deletes the migrated file from server storage the next time reconciliation is run.

The default management class delivered with ADSM specifies that a backup version of a file must exist before the file is eligible for migration.

Using the Standard Storage Management Policies

ADSM provides a set of policy objects, named STANDARD. If you use these standard objects, you can begin using ADSM immediately.

When you register a client node, the default is to assign the node to the STANDARD policy domain. If users register their own workstations during open registration, they are also assigned to the STANDARD policy domain.

ADSM provides a standard policy domain, policy set, management class, backup copy group, and archive copy group. Each policy object is named STANDARD. The attributes of the ADSM-supplied objects are as follows:

Standard Policy Domain
When a backed up file is no longer associated with a backup copy group, it remains in server storage for 30 days (backup retention grace period).

When an archived file is no longer associated with an archive copy group, it remains in server storage for 365 days (archive retention grace period).

Standard Policy Set (ACTIVE)
The default management class is STANDARD.

Standard Management Class
Client files are not space-managed (no client HSM).

Standard Backup Copy Group
Files are backed up to the default disk storage pool, BACKUPPOOL.

An incremental backup is performed only if the file has changed since the last backup.

Files cannot be backed up while they are being modified.

Up to two backup versions of a file on the client's system are retained in server storage. The most recent backup version is retained for as long as the original file is on the client file system. All other versions are retained for up to 30 days after they become inactive.

One backup version of a file that has been deleted from the client's system is retained in server storage for 60 days.

Standard Archive Copy Group
Files are backed up to the default disk storage pool, ARCHIVEPOOL.

Files cannot be archived while they are being modified.

An archive copy is kept for up to 365 days.

Creating Your Own Storage Management Policies

Task	Required Privilege Class
Define or copy a policy domain	System
Update a policy domain over which you have authority	Restricted policy
Define, update, or copy policy sets and management classes in any policy domain	System or unrestricted policy
Define, update, or copy policy sets and management classes in policy domains over which you have authority	Restricted policy
Define or update copy groups in any policy domain	System or unrestricted policy
Define or update copy groups that belong to policy domains over which you have authority	Restricted policy
Assign a default management class to a nonactive policy set in any policy domain	System or unrestricted policy
Assign a default management class to a nonactive policy set in policy domains over which you have authority	Restricted policy
Validate and activate policy sets in any policy domain	System or unrestricted policy
Validate and activate policy sets in policy domains over which you have authority	Restricted policy
Start inventory expiration processing	System

You may need more flexibility in your storage management policies than the standard ADSM policy objects provide. If so, you can create your own policies in either of two ways: you can define the objects by specifying each attribute, or you can copy existing objects and update only those attributes that you want to change. The following table shows another advantage of copying objects: some associated objects are copied in a single operation.

If you copy:	You create:
Policy Domain	A new policy domain with: • A copy of each policy set from the original domain • A copy of each management class in each original policy set • A copy of each copy group in each original management class
Policy Set	A new policy set **in the same policy domain** with: • A copy of each management class in the original policy set • A copy of each copy group in the original management class
Management Class	A new management class **in the same policy set** and a copy of each copy group in the management class

The rest of this chapter describes the tasks involved in creating new storage management policies for your installation:

1 Define policy domains to manage groups of client nodes. See page 202.

2 Define policy sets for different storage management policies. See page 203.

3 Define management classes to match users' storage management requirements. See page 204.

4 Define backup copy groups to specify which files can be backed up and how to manage backup versions. See page 205.

5 Define archive copy groups to specify whether a file can be archived if it is in use and to manage archive copies. See page 209.

6 Assign a default management class to each policy set to match the most common storage management requirements of client nodes in the policy domain. See page 211.

7 Validate all policy sets, and activate one policy set for each policy domain. See page 212.

8 Start expiration processing. See page 213.

To help users take advantage of ADSM, you can set up the policy environment by doing the following:

- Create include-exclude lists for inexperienced users or for users who have simple storage management needs
- Provide a sample include-exclude list to users who want to specify how ADSM manages their files. You can show users who prefer to manage their own files how to:
  - Request information about management classes.
  - Select a management class that meets backup and archive requirements.
  - Use include-exclude lists to bind management classes to their files.

  For information on how to create an include-exclude list, see the user's publication for the appropriate client.
- Automate incremental back up procedures by defining schedules for each policy domain. Then associate schedules with client nodes in each policy domain. For information on schedules, see Chapter 11, “Automating Operations” on page 223.

Example: Sample Policy Objects

Figure 39 on page 201 shows the policies for an engineering department. This example is used throughout the rest of this chapter.

The domain contains two policy sets, STANDARD and SUMMER. The policy set named STANDARD is active. Only one policy set can be active at a time. When a policy set is activated, the server makes a copy of the policy set and names it ACTIVE.

The ACTIVE policy set contains four management classes: ENGINEERING, MCENG, MCENGBK3, and MCENGAR2. The default management class is MCENG.

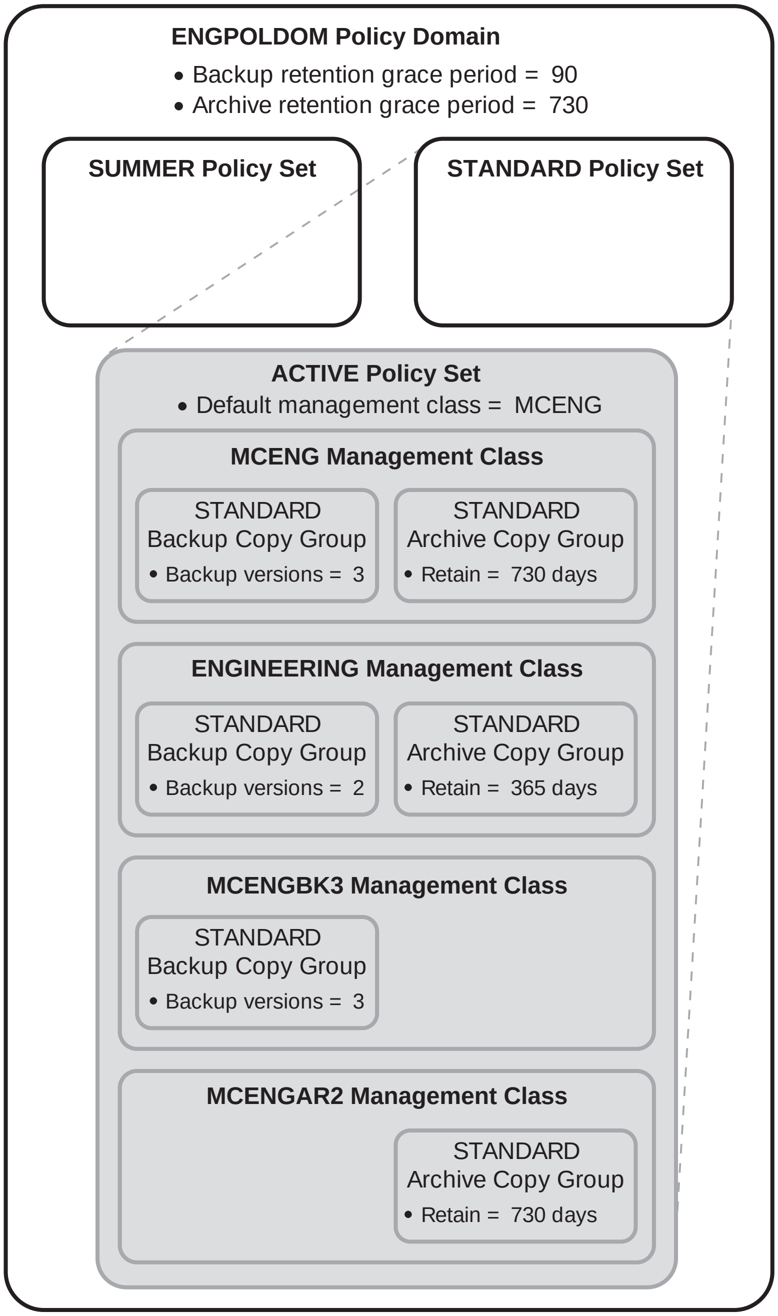

ENGPOLDOM Policy Domain
• Backup retention grace period = 90
• Archive retention grace period = 730
SUMMER Policy Set
STANDARD Policy Set
ACTIVE Policy Set
• Default management class = MCENG
MCENG Management Class
STANDARD Backup Copy Group
• Backup versions = 3
STANDARD Archive Copy Group
• Retain = 730 days
ENGINEERING Management Class
STANDARD Backup Copy Group
• Backup versions = 2
STANDARD Archive Copy Group
• Retain = 365 days
MCENGBK3 Management Class
STANDARD Backup Copy Group
• Backup versions = 3
MCENGAR2 Management Class
STANDARD Archive Copy Group
• Retain = 730 days

Figure 39. An Example of Policy Objects Defined for an Engineering Department

Defining and Updating a Policy Domain

When you update or define a policy domain, you specify:

Backup Retention Grace Period

Specifies the number of days to retain an inactive backup version when the server cannot rebind the file to an appropriate management class. The backup retention grace period protects backup versions from being immediately expired when the management class to which a file is bound no longer exists or no longer contains a backup copy group, and the default management class does not contain a backup copy group.

Backup versions of the file managed by the grace period are retained in server storage only for the backup retention grace period. This period starts from the day of the backup. For example, if the backup retention grace period for the STANDARD policy domain is used and set to 30 days, backup versions using the grace period expire in 30 days from the day of the backup.

Backup versions of the file continue to be managed by the grace period unless one of the following occurs:

- The client binds the file to a management class containing a backup copy group and then backs up the file
- A backup copy group is added to the file's management class
- A backup copy group is added to the default management class

Archive Retention Grace Period

Specifies the number of days to retain an archive copy when the management class for the file no longer contains an archive copy group and the default management class does not contain an archive copy group. The retention grace period protects archive copies from being immediately expired.

The archive copy of the file managed by the grace period is retained in ADSM storage for the number of days specified by the archive retention grace period. This period starts from the day on which the file is first archived. For example, if the archive retention grace period for the policy domain STANDARD is used, an archive copy expires 365 days from the day the file is first archived.

The archive copy of the file continues to be managed by the grace period unless an archive copy group is added to the file's management class or to the default management class.

Example: Defining a Policy Domain

To create a new policy domain you can do one of the following:

- Copy an existing policy domain and update the new domain
- Define a new policy domain from the beginning

Note: When you copy an existing domain, you also copy any associated policy sets, management classes, and copy groups.

For example, to copy and update, follow this procedure:

1 Copy the STANDARD policy domain to the ENGPOLDOM policy domain by entering:

```
copy domain standard engpoldom
```

ENGPOLDOM now contains the standard policy set, management class, backup copy group, and archive copy group.

2 Update the policy domain ENGPOLDOM so that the backup retention grace period is extended to 90 days and the archive retention grace period is extended to 2 years by entering:

```
update domain engpoldom description='Engineering Policy Domain'
backretention=90 archretention=730
```

Defining and Updating a Policy Set

When you define or update a policy set, specify:

Policy domain name
Names the policy domain to which the policy set belongs

Example: Defining a Policy Set

A business with seasonal employees needs two policy sets. During most of the year, most users would use the STANDARD policy set. During the summer, it would activate the SUMMER policy set to provide new management classes for users who are seasonal employees. To create the SUMMER policy set in the STANDARD policy domain, the business would perform the following steps:

1 Copy the STANDARD policy set and name the new policy set SUMMER:

```
copy policyset standard standard summer
```

Note: When you copy an existing policy set, you also copy any associated management classes and copy groups.

2 Update the description of the policy set named SUMMER:

```
update policyset standard summer
description='Policy set activated during summer for STANDARD domain'
```

Defining and Updating a Management Class

When you define or update a management class, specify:

Policy domain name
Names the policy domain to which the management class belongs.

Policy set name
Names the policy set to which the management class is assigned.

Whether hierarchical storage management (HSM) is to be done
Specifies that the files are eligible for both automatic and selective migration, only selective migration, or no migration.

How frequently files can be migrated
Specifies the minimum number of days that must elapse since a file was last accessed before it is eligible for automatic migration.

Whether backup is required
Specifies whether a backup version of a file must exist before the file can be migrated.

Where migrated files are to be stored
Specifies the name of the storage pool in which migrated files are stored. Your choice could depend on factors such as:

- The number of client nodes migrating to the storage pool. When many user files are stored in the same storage pool, volume contention can occur as users try to migrate files to or recall files from the storage pool.
- How quickly the files must be recalled. If users need immediate access to migrated versions, you can specify a disk storage pool as the destination.

Note: You cannot specify a copy storage pool as a destination.

Example: Define a New Management Class

Create a new management class containing a backup copy group and an archive copy group:

1 Copy the STANDARD management class from the STANDARD policy set to the new management class (named MCENG) by entering:

```
copy mgmtclass engpoldom standard standard mceng
```

The server copies the management class description, standard backup copy group, and standard archive copy group to MCENG.

2 Update the description of the MCENG management class by entering:

```
update mgmtclass engpoldom standard mceng
description='Engineering Mgmt Class - Backup & Archive Copy Groups'
```

Defining and Updating a Backup Copy Group

To define or update a backup copy group on the graphical user interface or command line, specify:

Where backed up files are to be stored

Specifies a defined storage pool. Your choice can depend on factors such as:

- The number of client nodes backing up to the storage pool. When many user files are stored in the same storage pool, volume contention can occur as users try to back up to or restore files from the storage pool.
- How quickly the files must be restored. If users need immediate access to backup versions, you could specify a disk storage pool as the destination.

Note: You cannot specify a copy storage pool.

If files can be modified during backup

Specifies how files are handled if they are modified while being backed up and what ADSM does if modification occurs. This attribute, called serialization, can be one of four values:

Static

Specifies that if the file or directory is modified during a backup, ADSM does not back it up. ADSM does not retry the backup.

Shared Static

Specifies that if the file or directory is modified during a backup, ADSM does not back it up. However, ADSM retries the backup as many times as specified by the CHANGINGRETRIES option in the client options file.

Dynamic

Specifies that a file or directory is backed up on the first attempt, even if the file or directory is being modified during the backup.

Shared Dynamic

Specifies that if a file or directory is modified during a backup attempt, ADSM backs it up on its last try even if the file or directory is being modified. ADSM retries the backup as many times as specified by the CHANGINGRETRIES option in the client options file.

For most files, set serialization to either static or shared static to prevent the server from backing up a file while it is being modified.

Attention: If a file is backed up while it is in use (shared dynamic or dynamic serialization), the copy may not contain all the changes and may not be usable.

However, you may want to define a copy group with a serialization of shared dynamic or dynamic for files where log records are continuously added, such as an error log. If you only have copy groups that use static or shared static, these files may never be backed up because they are constantly in use. With shared dynamic or dynamic, the log files are backed up. However, the backup version may contain a truncated message.

Note: When certain users or processes open files, they deny read access to the files for any other user or process. When this happens, even with serialization set to dynamic or shared dynamic, ADSM does not back up the file.

How frequently files can be backed up

Specifies the minimum number of days that must elapse between full incremental backups. Frequency works with the mode parameter, which specifies whether a file or directory is considered for full incremental backup only if it has changed since the last backup or regardless of whether it has been changed. ADSM does not check this attribute when a user requests a partial incremental backup or a selective backup for a file. You can select from two modes:

Modified

A file is considered for full incremental backup only if it has changed since the last backup. A file is considered changed if any of the following items is different:

- Date on which the file was last modified
- File size
- File owner
- File permissions

Absolute

A file is considered for full incremental backup regardless of whether it has changed since the last backup.

For example, if frequency is 3 and mode is modified, a file or directory is backed up only if it has been changed and if three days have passed. If frequency is 3 and mode is absolute, a file or directory is backed up after three days have passed whether or not the file has changed.

Use the modified mode when users want to retain multiple backup versions. If the mode is set to absolute, users may have three *identical* backup versions, rather than three different backup versions.

Absolute mode can be useful for forcing a full backup. It can also be useful for ensuring that OS/2 files with extended attributes are backed up, because ADSM does not detect changes to the extended attributes.

When you set the mode to absolute, set frequency to 0 if you want to ensure that a file is backed up each time full incremental backups are scheduled for or initiated by a client.

How many backup versions to retain

Specifies the number of backup versions. Multiple versions of files are useful when users continually update files and sometimes need to restore the original file from which they started. Two parameters determine how many active and inactive backup copies to retain:

Versions Data Exists

The maximum number of different backup versions that the server retains for files and directories currently on the workstation.

If users select a management class that allows more than one backup version, the most current version is called the *active* version. All other versions are called *inactive* versions.

For example, in Figure 40 on page 208, the most current version of REPORT.TXT was created on Friday at 3 p.m. There are two inactive versions of REPORT.TXT.

When the maximum number of backup versions is exceeded, the oldest version expires and the server deletes it the next time expiration processing runs.

For example, if the maximum number of versions allowed for MEMO.DAT is three, and a user runs a backup process that creates a fourth version, the oldest version expires. In this example, the backup version created on Thursday at 8:05 a.m. expires.

How many inactive versions ADSM keeps is also related to the parameter for how long inactive versions are kept (Retain Extra Versions). Inactive versions can expire when their age exceeds the value specified for retaining extra versions, even when the number of versions is not exceeded.

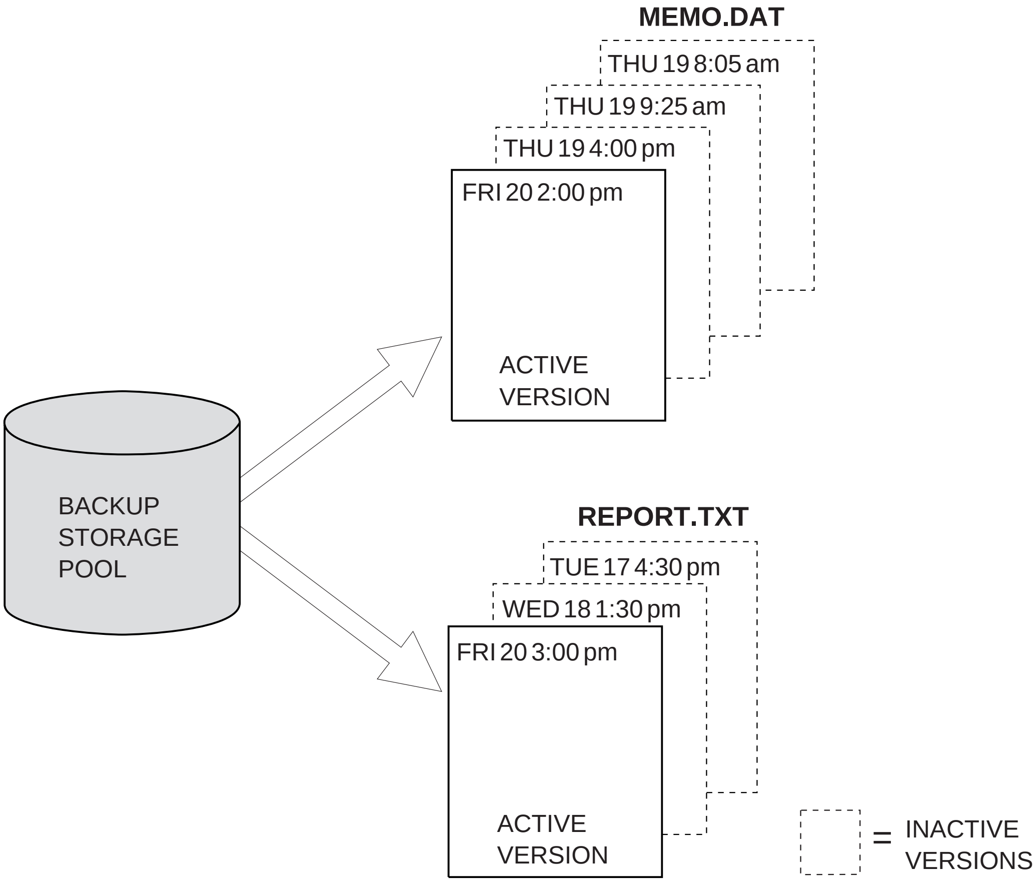


Figure 40. Example of Active and Inactive Versions of Backed Up Files

Versions Data Deleted

The maximum number of different backup versions that the server retains for files and directories that have been erased from a workstation. The server ignores this parameter while the file or directory remains on the workstation.

If users erase a file or directory from their client nodes, then the next time a full incremental backup is run, the server changes the active backup version to inactive. The oldest versions that are more than the number specified by this parameter then expire, and the server deletes them the next time expiration processing runs.

The expiration date for the remaining versions is based on the Retain Extra Versions and Retain Only Version parameters.

How long to retain files in storage

Specifies how long to retain backup versions:

Retain Extra Versions

Specifies how many days ADSM retains a backup version after that version becomes inactive (that is, a more recent backup version is stored). The value of this parameter determines which versions are deleted during inventory expiration processing.

If NOLIMIT is specified, inactive backup versions are deleted based on the Versions Data Exists or Versions Data Deleted parameters.

Retain Only Version

Specifies how many days ADSM retains the only backup version it has of a file when the original file has been deleted from the workstation.

If NOLIMIT is specified, the last version is retained forever unless a user or administrator deletes the file from server storage.

Example: Define a Backup Copy Group

Define a backup copy group belonging to the MCENG management class in the STANDARD policy set belonging to the ENGPOLDOM policy domain. This new copy group must do the following:

- Let users back up changed files, regardless of how much time has elapsed since the last backup
- Retain up to 4 inactive backup versions when the original file resides on the user workstation
- Retain up to four inactive backup versions when the original file is deleted from the user workstation
- Retain extra inactive backup versions for 90 days
- If there is only one backup version, retain it for 600 days after the original is deleted from the workstation
- Prevent files from being backed up if they are in use
- Store files in the ENGBACK1 storage pool

To define the backup copy group, enter:

```
define copygroup engpoldom standard mceng standard
destination=engback1 serialization=static
verexists=5 verdeleted=4 retextra=90 retonly=600
```

Defining and Updating an Archive Copy Group

To define or update an archive copy group on the graphical user interface or command line, specify:

Where archived files are to be stored

Specifies a defined storage pool. Your choice can depend on factors such as:

- The number of client nodes archiving files to the storage pool. When many user files are stored in the same storage pool, volume contention can occur as users archive files to and retrieve files from the storage pool.
- How quickly the files must be restored. If users need immediate access to archive copies, you could specify a disk storage pool as the destination.

Note: You cannot specify a copy storage pool as a destination.

If files can be modified during archive

Specifies how files are handled if they are modified while being archived and what ADSM does if modification occurs. This attribute, called serialization, can be one of four values:

Static

Specifies that if the file is modified during an archiving process, ADSM does not archive it. ADSM does not retry the archive.

Shared Static

Specifies that if the file is modified during an archive process, ADSM does not archive it. However, ADSM retries the archive process as many times as specified by the CHANGINGRETRIES option in the client options file.

Dynamic

Specifies that a file is archived on the first attempt, even if the file is being modified during the archive process.

Shared Dynamic

Specifies that if the file is modified during the archive attempt, ADSM archives it on its last try even if the file is being modified. ADSM retries the archive process as many times as specified by the CHANGINGRETRIES option in the client options file.

For most files, set serialization to either static or shared static to prevent the server from archiving a file while it is being modified.

Attention: If a file is archived while it is in use (shared dynamic or dynamic serialization), the copy may not contain all the changes and may not be usable.

However, you may want to define a copy group with a serialization of shared dynamic or dynamic for files where log records are continuously added, such as an error log. If you only have copy groups that use static or shared static, these files may never be archived because they are constantly in use. With shared dynamic or dynamic, the log files are archived. However, the archive copy may contain a truncated message.

Note: When certain users or processes open files, they deny read access to the files for any other user or process. When this happens, even with serialization set to dynamic or shared dynamic, ADSM does not back up the file.

How long to retain an archived copy

Specifies the number of days to retain an archived copy in storage. When the time elapses, the archived copy expires and ADSM deletes the file the next time expiration processing runs.

Example: Define an Archive Copy Group

Define an archive copy group belonging to the MCENG class that:

- Allows users to archive a file if it is not in use
- Retains the archive copy for 730 days
- Stores files in the ENGARCH1 storage pool

To define a STANDARD archive copy group to the MCENG management class in the STANDARD policy set belonging to the ENGPOLDOM policy domain, enter:

```
define copygroup engpoldom standard mceng standard
type=archive destination=engarch1 serialization=static
retver=730
```

Assigning a Default Management Class

After you have defined your policy sets and the management classes that they contain, you must assign a default management class for each policy set. See “Default Management Classes” on page 190 for suggestions about the content of default management classes.

Example: Assign a Default Management Class

To assign the STANDARD management class as the default management class for the SUMMER policy set in the STANDARD policy domain, enter:

```
assign defmgmtclass standard summer standard
```

The STANDARD management class was copied from the STANDARD policy set to the SUMMER policy set (see “Example: Defining a Policy Set” on page 203). Before the new default management class takes effect, you must activate the policy set.

Validating and Activating Policy Sets

After you have defined your policy sets and assigned management classes to them, you can validate those policy sets and activate one policy set for the policy domain.

Validating Policy Sets

When you validate a policy set, the server examines the management class and copy group definitions in the specified policy set and reports on conditions that need to be considered if the policy set is activated.

Validation fails if the policy set does not contain a default management class. The following conditions result in warning messages during validation:

- The storage destinations specified for backup, archive, or migration do not refer to defined storage pools.

  A backup, archive, or migration operation will fail when the operation involves storing a file in a storage pool that does not exist.

- A storage destination specified for backup, archive, or migration is a copy storage pool.

- The default management class does not contain a backup or archive copy group.

  When the default management class does not contain a backup or archive copy group, any user files bound to the default management class *are not* backed up or archived.

- The current ACTIVE policy set names a management class that is not defined in the policy set being validated.

  When users back up files that were bound to a management class that no longer exists in the active policy set, backup versions are rebound to the default management class. See “How Files Are Associated with a Management Class” on page 192 for details.

  When the management class to which an archive copy is bound no longer exists and the default management class does not contain an archive copy group, the archive retention grace period is used to retain the archive copy. See “Defining and Updating a Policy Domain” on page 202 for details.

- The current ACTIVE policy set contains copy groups that are not defined in the named policy set.

  When users perform a backup and the backup copy group no longer exists in the management class to which a file is bound, backup versions are managed by the default management class if it contains a backup copy group. If the default management class does not contain a backup copy group, backup versions are managed by the backup retention grace period, and the workstation file is not backed up. See “Defining and Updating a Policy Domain” on page 202.

- A management class specifies that a backup version must exist before a file can be migrated from a client node, but the management class does not contain a backup copy group.

Activating Policy Sets

To activate a policy set, specify a policy domain and policy set name. When you activate a policy set, the server:

- Performs a final validation of the contents of the policy set
- Copies the original policy set to the active policy set

After a policy set has been activated, the original and the ACTIVE policy sets are two separate objects. For example, updating the original policy set has no effect on the

ACTIVE policy set. You cannot update the ACTIVE policy set. To change its contents, you must do the following:

1. Copy the ACTIVE policy set to a policy set with another name.
2. Update the new policy set.
3. Validate the new policy set.
4. Activate the new policy set to have the server use the changes.

Example: Validating and Activating a Policy Set

Validating and activating the SUMMER policy set in the STANDARD policy domain is a two-step process:

1 To validate the SUMMER policy set, enter:

```
validate policyset standard summer
```

2 To activate the SUMMER policy set, enter:

```
activate policyset standard summer
```

Running Expiration Processing to Delete Expired Files

Copies of files that have expired are not deleted from server storage until expiration processing occurs. You can run expiration processing either automatically or by command. You control automatic expiration processing by using the expiration interval option (EXPINTERVAL) in the ADSM server options file (dsmserv.opt). You can set the option by editing the dsmserv.opt file (see *ADSM Administrator's Reference*). You can manually start expiration processing by issuing the following command:

```
expire inventory
```

Expiration processing then deletes eligible backup versions and archive file copies. Backup versions are eligible based on policy in the backup copy group (how long and how many inactive versions are kept). Archive file copies are eligible based on policy in the archive copy group (how long archived copies are kept).

When expiration processing runs, normally ADSM sends detailed messages about policy changes made since the last time expiration processing ran. The messages are about changes made that affect client files, such as deleting a management class or a copy group. You can reduce the number of messages about policy changes that are generated during expiration processing by using a *quiet* option in the server options, or a QUIET=YES parameter with the EXPIRE INVENTORY command. When you use the quiet option or parameter, ADSM issues messages about policy changes during expiration processing only when files are deleted, and either the default management class or retention grace period for the domain has been used to expire the files.

Querying Policy Objects

Task	Required Privilege Class
Query any policy domain, policy set, management class, or copy group	Any administrator

You can request information about the contents of ADSM policy objects. For example, you might want to do this before creating new objects or helping users to choose policies that fit their needs.

You can specify the output of a query in either standard or detailed format. The examples in this book are in standard format. Refer to *ADSM Administrator's Reference* for examples of detailed format output.

Querying Copy Groups

To request information about backup copy groups (the default) in the ENGPOLDOM engineering policy domain, enter:

```
query copygroup engpoldom * *
```

The following shows the output from the query. It shows that the ACTIVE policy set contains two backup copy groups that belong to the MCENG and STANDARD management classes.

```
Policy     Policy     Mgmt       Copy       Versions  Versions    Retain   Retain
Domain     Set Name   Class      Group          Data      Data     Extra     Only
Name                  Name       Name         Exists   Deleted  Versions  Version
---------  ---------  ---------  ---------  --------  --------  --------  -------
ENGPOLDOM  ACTIVE     MCENG      STANDARD          5         4        90      600
ENGPOLDOM  ACTIVE     STANDARD   STANDARD          2         1        30       60
ENGPOLDOM  STANDARD   MCENG      STANDARD          5         4        90      600
ENGPOLDOM  STANDARD   STANDARD   STANDARD          2         1        30       60
ENGPOLDOM  SUMMER     MCENG      STANDARD          2         1        30       60
ENGPOLDOM  SUMMER     STANDARD   STANDARD          2         1        30       60
```

To request information about archive copy groups in the ENGPOLDOM engineering policy domain, enter:

```
query copygroup engpoldom * type=archive
```

The following shows the output from the query.

```
Policy        Policy        Mgmt          Copy            Retain
Domain        Set Name      Class         Group          Version
Name                        Name          Name
---------     ---------     ---------     ---------     --------
ENGPOLDOM     ACTIVE        MCENG         STANDARD           730
ENGPOLDOM     ACTIVE        STANDARD      STANDARD           365
ENGPOLDOM     STANDARD      MCENG         STANDARD           730
ENGPOLDOM     STANDARD      STANDARD      STANDARD           365
ENGPOLDOM     SUMMER        MCENG         STANDARD           730
ENGPOLDOM     SUMMER        STANDARD      STANDARD           365
```

Querying Management Classes

To request information about management classes in the ENGPOLDOM engineering policy domain, enter:

```
query mgmtclass engpoldom * *
```

The following figure is the output from the query. It shows that the ACTIVE policy set contains the MCENG and STANDARD management classes.

```
Policy       Policy       Mgmt         Default      Description
Domain       Set Name     Class        Mgmt
Name                      Name         Class ?
---------    ---------    ---------    ---------    ------------------------
ENGPOLDOM    ACTIVE       MCENG        Yes          Engineering Management
                                                     Class with Backup and
                                                     Archive Copy Groups
ENGPOLDOM    ACTIVE       STANDARD     No

ENGPOLDOM    STANDARD     MCENG        Yes          Engineering Management
                                                     Class with Backup and
                                                     Archive Copy Groups
                                                     versions
ENGPOLDOM    STANDARD     STANDARD     No

ENGPOLDOM    SUMMER       MCENG        Yes          Engineering Management
                                                     Class with Backup and
                                                     Archive Copy Groups
                                                     versions
ENGPOLDOM    SUMMER       STANDARD     No
```

Querying Policy Sets

To query the system for information about policy sets in the ENGPOLDOM engineering policy domain, enter:

```
query policyset engpoldom *
```

The following figure is the output from the query. It shows an ACTIVE policy set and two inactive policy sets, STANDARD and SUMMER.

```
Policy         Policy         Default        Description
Domain         Set Name       Mgmt
Name                          Class
                              Name
---------      ---------      ---------      ------------------------
ENGPOLDOM      ACTIVE         MCENG          Policy Set Activated
                                              During Summer
ENGPOLDOM      STANDARD

ENGPOLDOM      SUMMER         MCENG          Policy Set Activated
                                              During Summer
```

Querying Policy Domains

To request information about a policy domain (for example, to determine if any client nodes are registered to that policy domain), enter:

```
query domain *
```

The following figure is the output from the query. It shows that both the ENGPOLDOM and STANDARD policy domains have client nodes assigned to them.

```
Policy       Activated    Activated     Number of    Description
Domain       Policy       Default      Registered
Name         Set          Mgmt              Nodes
                          Class
---------    ---------    ---------    ----------    ------------------------
ENGPOLDOM    SUMMER       ENGMC                 3    Engineering Policy
                                                      Domain
STANDARD     STANDARD     STANDARD              3    Installed default policy
                                                      domain.
```

Deleting Policy Objects

You cannot delete the ACTIVE policy set or objects in that policy set. When you delete a policy object, you also delete any objects belonging to it.

Task	Required Privilege Class
Delete policy domains	System
Delete any policy sets, management classes, or copy groups	System or unrestricted policy
Delete policy sets, management classes, or copy groups that belong to policy domains over which you have authority	Restricted policy

You can delete the policy objects named STANDARD that ADSM provides. However, all STANDARD policy objects are restored whenever you reinstall the ADSM server. If you reinstall the server after the STANDARD policy objects have been deleted, messages are issued during processing of a subsequent DSMSERV AUDIT DB command. The messages indicate that "an instance count does not agree with actual data." The DSMSERV AUDIT DB command corrects this problem, but does not delete the restored STANDARD policy objects.

Deleting Copy Groups

You can delete a backup or archive copy group that does not belong to a management class in the ACTIVE policy set.

For example, to delete the backup and archive copy groups belonging to the MCENG and STANDARD management classes in the SUMMER policy set, enter:

```
delete copygroup engpoldom summer mceng type=backup

delete copygroup engpoldom summer standard type=backup

delete copygroup engpoldom summer mceng type=archive

delete copygroup engpoldom summer standard type=archive
```

Deleting Management Classes

You can delete a management class that does not belong to the ACTIVE policy set.

For example, to delete the MCENG and STANDARD management classes from the SUMMER policy set, enter:

```
delete mgmtclass engpoldom summer mceng

delete mgmtclass engpoldom summer standard
```

When you delete a management class from a policy set, the server deletes the management class and all copy groups that belong to the management class in the specified policy domain.

Deleting Policy Sets

Authorized administrators can delete any policy set other than the ACTIVE policy set. For example, to delete the SUMMER policy set from the ENGPOLDOM engineering policy domain, enter:

```
delete policyset engpoldom summer
```

When you delete a policy set, the server deletes all management classes and copy groups that belong to the policy set within the specified policy domain.

Deleting Policy Domains

You can delete a policy domain that has no client nodes registered to it. To determine if any client nodes are registered to a policy domain, issue the QUERY DOMAIN or the QUERY NODE command.

For example, to delete the STANDARD policy domain, perform the following steps:

1. Request a list of all client nodes assigned to the STANDARD policy domain by entering:

```
query node * domain=standard
```

2. If client nodes are assigned to the policy domain, remove them in either of the following ways:

   - Assign each client to a new policy domain. For example, enter the following commands:

```
update node htang domain=engpoldom
update node tomc domain=engpoldom
update node pease domain=engpoldom
```

   If the active policy set in ENGPOLDOM does not have the same management class names as in the active policy set of the STANDARD policy domain, then backup versions of files may be bound to a different management class name, as described in “How Files Are Associated with a Management Class” on page 192.

   - Delete each node from the STANDARD policy domain.

3 Delete the policy domain by entering:

```
delete domain standard
```

When you delete a policy domain, the server deletes the policy domain and all policy sets (including the ACTIVE policy set), management classes, and copy groups that belong to the policy domain.

Part 4. Automating Operations

© Copyright IBM Corp. 1995, 1997

Chapter 11. Automating Operations

ADSM includes a central scheduling component that allows the automatic processing of administrative commands and client operations during a specific time period when the schedule is activated.

Administrative commands can be scheduled for use in tuning server operations and to start functions that require significant server or system resources during times of low usage. Automating these operations allows the administrator to ensure that server resources are available when needed by clients.

Administrators can use central scheduling to automate client operations so that clients do not have to perform the operations manually. You can schedule the following client operations:

- Backups (incremental and selective)
- Archives
- Restores
- Retrieves
- Client operating system commands
- Executable scripts on the client (containing operating system commands, ADSM commands, or both)

Each administrative command and each scheduled client operation is called an *event*. Each scheduled event is tracked by the server and recorded in the database. Event records can be deleted from the database as needed to recover database space.

The sections listed in the following table begin at the indicated pages.

Section	Page
Tasks:	
Automating server operations	224
Automating client operations	225
Coordinating client schedules	228
Tailoring schedules	234
Copying schedules	238
Deleting schedules	239
Managing client node associations	242
Managing scheduled events	239

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, refer to *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

© Copyright IBM Corp. 1995, 1997

Appendix C, “Interface Cross-Reference” on page 467 lists each command and shows if its function is also available on the administrative client GUI.

Automating Server Operations

You can define a schedule to automate administrative commands. Functions that require significant server or system resources can be automatically scheduled to execute at a time when server resources are available and client node processing is at a minimum.

Notes:

1. Scheduled administrative command output is directed to the activity log. This output cannot be redirected. For information about the length of time activity log information is retained in the database, see “Managing the Activity Log” on page 255.
2. You cannot schedule MACRO or QUERY ACTLOG commands.

This section describes how to set up a basic administrative command schedule using ADSM defaults. To later update or tailor your schedules, see “Tailoring Schedules” on page 234.

Task	**Required Privilege Class**
Define, update, copy, or delete administrative schedules	System
Display information about scheduled operations	Any administrator

Defining the Schedule

Use the DEFINE SCHEDULE command to create a new schedule to process an administrative command. Include the following parameters:

- Specify the administrative command to be issued (CMD=).
- Specify whether the schedule is to be activated (ACTIVE=).

For example:

```
define schedule backup_archivepool type=administrative
cmd='backup stgpool archivepool recoverypool' active=yes
```

This command results in the following:

- The schedule created is *BACKUP_ARCHIVEPOOL*.
- The schedule is to process the administrative command:

  `backup stgpool archivepool recoverypool`

  This command specifies that primary storage pool ARCHIVEPOOL is backed up to the copy storage pool RECOVERYPOOL.

- The schedule is currently active.
- Administrative command output is redirected to the activity log.
- The following defaults are in effect:
  - The start date and time defaults to the current date and time.
  - The length of the startup window is 1 hour.
  - The priority for the schedule is 5. If schedules conflict, the schedule with the highest priority (lowest number) is run first.
  - The schedule never expires.

To change the defaults, see “Tailoring Schedules” on page 234.

Verifying the Schedule

You can verify the details of what you have scheduled by using the QUERY SCHEDULE command. When you use the QUERY SCHEDULE command, you must specify the TYPE=ADMINISTRATIVE parameter to view an administrative command schedule. The following figure shows an example of a report that is displayed after you enter:

```
query schedule backup_archivepool type=administrative
```

```
*   Schedule Name       Start Date/Time       Duration   Period   Day
-   ----------------    --------------------  --------   ------   ---
    BACKUP_ARCHIVEP-    11/15/1995 14:08:11        1 H      1 D    Any
     OOL
```

Note: The asterisk (*) in the first column specifies whether the corresponding schedule has expired. If there is an asterisk in this column, the schedule has expired.

You can check when the schedule is projected to run and whether it ran successfully by using the QUERY EVENT command. For information about querying events, see “Querying Event Records” on page 239.

Automating Client Operations

To automate client operations, you can define a new schedule or update an existing schedule. When you define a schedule, you assign it to a specific policy domain. You can define more than one schedule for each policy domain.

This section describes how to automate a basic client operation, incremental backup. The example uses ADSM defaults. To later update or tailor your schedules, see “Tailoring Schedules” on page 234.

To set up a client schedule on the server:

1. Define a schedule (DEFINE SCHEDULE command).
2. Associate client nodes with the schedule (DEFINE ASSOCIATION command).
3. Ensure that the clients start the client scheduler to use the server's schedule.
4. Verify the schedule (QUERY SCHEDULE and QUERY EVENT commands).

Task	Required Privilege Class
Define, update, copy, or delete any client schedules	System or unrestricted policy
Define, update, copy, or delete client schedules for specific policy domains	System, unrestricted policy, or restricted policy for those domains
Display information about scheduled operations	Any administrator

Defining the Client Schedule

To define a schedule for incremental backups, use the DEFINE SCHEDULE command. You must specify the policy domain to which the schedule belongs and the name of the schedule (the policy domain must already be defined). For example:

```
define schedule engpoldom weekly_backup
```

This command results in the following:

- Schedule *WEEKLY_BACKUP* is defined for policy domain *ENGPOLDOM*.
- The following defaults are in effect:
  - The scheduled action is an incremental backup (the default action).
  - The priority for the operation is 5. If schedules conflict, the schedule with the highest priority (lowest number) is run first.
  - The schedule window begins now and the schedule itself has 1 hour to start.
  - The start window is scheduled every day.
  - The schedule never expires.

To change the defaults, see "Tailoring Schedules" on page 234.

Associating Client Nodes with Schedules

Client nodes process operations according to the schedules associated with the nodes. To associate client nodes with a schedule, use the DEFINE ASSOCIATION command. A client node can be associated with more than one schedule. However, a node must be assigned to the policy domain to which a schedule belongs.

After a client schedule is defined, you can associate client nodes with it by identifying the following information:

- Policy domain to which the schedule belongs

- List of client nodes to be associated with the schedule

To associate the ENGNODE client node with the WEEKLY_BACKUP schedule, both of which belong to the ENGPOLDOM policy domain, enter:

```
define association engpoldom weekly_backup engnode
```

Starting the Scheduler on the Clients

The client scheduler must be started before work scheduled by the ADSM administrator can be initiated.

To start the client scheduler, the client must issue the SCHEDULE command provided with the ADSM backup-archive client. For example, on an OS/2 client, issue the following command:

```
> dsmc schedule
```

The client can choose to start the client scheduler when the operating system is started, or can start it at any appropriate time. For example, an OS/2 client can include the SCHEDULE command in the startup.cmd file to start the client scheduler when the operating system is started.

For more information, refer to the appropriate *ADSM Using the Backup-Archive Client*.

After the client node starts the client scheduler, it continues to run and initiates scheduled events until it is stopped.

Verifying the Schedule

You can verify what you have scheduled by using the QUERY SCHEDULE command. You can check whether the schedule ran successfully by using the QUERY EVENT command.

Verifying the Schedule

When you request information about schedules, the server displays the following information:

- Schedule name
- Policy domain name
- Type of operation to be performed
- Start date and time for the initial startup window
- Duration of the startup window
- Time period between startup windows
- Day of the week on which scheduled operations can begin

The following figure shows an example of a report that is displayed after you enter:

```
query schedule engpoldom
```

```
Domain       * Schedule Name    Action Start Date/Time      Duration Period Day
------------ - ---------------- ------ -------------------- -------- ------ ---
ENGPOLDOM      MONTHLY_BACKUP   Inc Bk 09/21/1995 12:45:14      2 H    2 Mo Sat
ENGPOLDOM      WEEKLY_BACKUP    Inc Bk 09/21/1995 12:46:21      4 H    1 W  Sat
```

Checking whether the Schedule Completed Successfully

A scheduled client operation, called an *event*, is tracked by the server. You can get information about projected and actual scheduled processes by using a general query. You can get information about scheduled processes that did not complete successfully by using exception reporting.

For example, you can issue the following command to find out which events were missed in the ENGPOLDOM policy domain for the WEEKLY_BACKUP schedule in the previous week:

```
query event engpoldom weekly_backup begindate=-7 begintime=now
enddate=today endtime=now exceptionsonly=yes
```

For more information about managing event records, see “Managing Scheduled Event Records” on page 239.

Coordinating Client Schedules

By coordinating client schedules, you can control the workload that scheduled operations place on the server and clients.

The following sections describe:

- Setting the scheduling mode
- Specifying the schedule period for incremental backup operations
- Controlling the server’s scheduled workload
- Controlling client contact with the server

Task	Required Privilege Class
• Set the scheduling mode • Set the maximum percentage of sessions for scheduled operations • Randomize schedule start times • Set how often clients query the server • Set the maximum number of times the client node scheduler can retry a command that failed • Set the time between retry attempts	System

Setting the Scheduling Mode

The central scheduler on the server uses the default of both *client-polling* and *server-prompted* scheduling modes to process scheduled client operations. This default mode is specified as *any*. When the scheduling mode is *any*, the client can choose either scheduling mode. If you specify only one mode for the server, the clients must specify the same mode in their options file. Otherwise, scheduled client work is not processed. The default mode for the clients is *polling*.

Setting Client-Polling Scheduling Mode on the Server

You can use the client-polling scheduling mode with all communication methods.

With this mode, the following occurs:

1. A client node queries the server at prescribed time intervals to obtain a schedule. This interval is set with a client node option. For information about client options, refer to the appropriate *ADSM Using the Backup-Archive Client*.
2. When the scheduled start time begins, the client node performs the scheduled operation and sends the results to the server.
3. The client node then queries the server for its next scheduled operation.

To have clients poll the server for scheduled operations, enter:

```
set schedmodes polling
```

Note: When the scheduling mode on the server is set to polling, the mode on the client node also must be set to polling for scheduled work to be processed.

Setting the Server-Prompted Scheduling Mode on the Server

You can use the server-prompted scheduling mode only with client nodes that communicate with the server by using the TCP/IP communication method.

With this mode, the following occurs:

1. Client nodes register their addresses with the server.
2. The server contacts the client when scheduled operations need to be performed and a session is available.

3. When contacted, the client node queries the server for the operation, performs the operation, and sends the results to the server.

To have the server prompt client nodes when operations need to be performed, enter:

```
set schedmodes prompted
```

Note: When the scheduling mode on the server is set to prompted, the scheduling mode on the client node also must be set to prompted for scheduled work to be processed.

Setting the Any Scheduling Mode on the Server

To let the server support both client-polling and server-prompted scheduling modes, enter:

```
set schedmodes any
```

In this case, the client node may choose the scheduling mode and scheduled work will begin as specified.

Setting the Scheduling Mode on Client Nodes

Users (root users on UNIX systems) set the scheduling mode on client nodes. They specify either the client-polling or the server-prompted scheduling mode on the command line or in the client user options file (client system options file on UNIX systems).

For more information, refer to the appropriate *ADSM Using the Backup-Archive Client*.

Specifying the Schedule Period for Incremental Backup Operations

When you define a backup copy group, you specify the copy frequency, which is the minimum interval between successive backups. See “Defining and Updating a Backup Copy Group” on page 205. When you define a schedule, you specify the length of time between processing of the schedule. Consider the backup copy group frequencies you have defined in each management class in a policy domain when you specify the schedule period for incremental backups. Schedules for incremental backups do not need to be processed more often than the backup copy group frequency.

Controlling the Server’s Scheduled Workload

To enable the server to complete all schedules for clients, you may need to use trial and error to control the workload. To estimate how long client operations take, test schedules on several representative client nodes. Keep in mind, for example, that the first incremental backup for a client node takes longer than subsequent incremental backups.

Increasing the size of the startup window (by increasing the schedule's duration) can also affect whether a schedule completes successfully. A larger startup window gives the client node more time to attempt initiation of a session with the server.

The settings for randomization and the maximum percentage of scheduled sessions can affect whether schedules are successfully completed for client nodes. Users receive a message if all sessions are in use when they attempt to process a schedule. If this happens, you can increase randomization and the percentage of scheduled sessions allowed to make sure the server can handle the workload.

An administrator can:

- Set the maximum percentage of concurrent client/server sessions for scheduled operations
- Randomize schedule start times for client operations

Setting the Maximum Percentage of Sessions for Scheduled Operations

The number of concurrent client/server sessions is defined by the MAXSESSIONS server option for the maximum client sessions, but you can set a maximum percentage of concurrent client/server sessions allowed for processing scheduled operations. Limiting the number of sessions available for scheduled operations ensures that sessions are available when users initiate any unscheduled operations, such as restoring or retrieving files, or backing up or archiving files.

If the number of sessions for scheduled operations is insufficient, you can increase either the total number of sessions or the maximum percentage of scheduled sessions. However, increasing the total number of sessions can adversely affect server performance, and increasing the maximum percentage of scheduled sessions can reduce the server opportunity to process unscheduled operations.

For example, assume that the maximum number of sessions between client nodes and the server is 80. If you want 25 percent of these sessions to be used by central scheduling, enter:

```
set maxschedsessions 25
```

The server allows 20 sessions to be used for scheduled operations.

For information about the MAXSESSIONS option, refer to *ADSM Administrator's Reference*.

Randomizing Schedule Start Times

To randomize start times for schedules means to scatter each schedule's start time across its startup window. A startup window is the start time and duration during which a schedule must be initiated.

For the client-polling scheduling mode, you can specify the percentage of the startup window that the server can use to randomize start times for different client nodes associated with a schedule.

If you set randomization to 0, no randomization occurs. This process can result in communication errors if many client nodes try to contact the server at the same instant.

The maximum percentage of randomization allowed is 50 percent. This limit ensures that half of the startup window is available for retrying scheduled commands that have failed.

It is possible, especially after a client node or the server has been restarted, that a client node may not poll the server until *after* the beginning of the startup window in which the next scheduled event is to start. In this case, the starting time is randomized over the specified percentage of the *remaining* duration of the startup window.

Consider the following situation:

- The startup window for a particular event is from 8:00 to 9:00
- Ten client nodes are associated with the schedule
- Nine client nodes poll the server before 8:00
- One client node does not poll the server until 8:30

To set randomization to 50 percent enter:

```
set randomize 50
```

The result is that the nine client nodes that polled the server *before* the beginning of the startup window are assigned randomly selected starting times between 8:00 and 8:30. The client node that polled at 8:30 receives a randomly selected starting time that is between 8:30 and 8:45.

Controlling Contact with the Server

To control how often client nodes contact the server to perform a scheduled operation, an administrator can set:

- How often clients query the server
- The number of command retry attempts
- The amount of time between retry attempts

Users (root users on UNIX systems) can also set these values in their client user options files (client system options files for UNIX systems). However, user values are overridden by the values that the administrator specifies.

The client node communication paths to the server can vary widely with regard to response time or the number of gateways. In such cases, you can choose *not* to set these values so that users can tailor them for their own needs.

Setting How Often Clients Query the Server

For the client-polling scheduling mode, you can specify the maximum number of hours the scheduler on a client node waits between attempts to contact the server to obtain a schedule.

You can set this period to correspond to the frequency with which the schedule changes are being made. If client nodes poll more frequently for schedules, changes to scheduling information (through administrator commands) are propagated more quickly to client nodes. However, increased polling by client nodes also increases network traffic.

If you want to have all clients using polling mode contact the server every 24 hours, enter:

```
set queryschedperiod 24
```

Setting the Number of Command Retry Attempts

You can specify the maximum number of times the scheduler on a client node can retry a scheduled command that fails.

The maximum number of command retry attempts does not limit the number of times that the client node can contact the server to obtain a schedule. The client node never gives up when trying to query the server for the next schedule.

Be sure not to specify so many retry attempts that the total retry time is longer than the average startup window.

If you want to have all client schedulers retry a failed attempt to process a scheduled command only twice, enter:

```
set maxcmdretries 2
```

Setting the Amount of Time between Retry Attempts

You can specify the number of minutes the scheduler on a client node waits between retry attempts after a failed attempt to contact the server or after a scheduled command fails to process. You can use this number in conjunction with the number of command retry attempts to control when a client node contacts the server to process a failed command.

Try setting this period to half of the estimated time it takes to process an average schedule.

If you want to have the client scheduler retry failed attempts to contact the server or to process scheduled commands every 15 minutes, enter:

```
set retryperiod 15
```

Tailoring Schedules

To control more precisely when and how your schedules run, you can specify values for schedule parameters instead of accepting the defaults when you define or update schedules.

You can define or update schedules for both administrative commands and client operations. Some parameters for the DEFINE and UPDATE commands apply to both administrative command and client schedules, while others only apply to one type of schedule. This section describes the following:

- Common schedule parameters
- Parameters for administrative command schedules
- Parameters for client schedules

Common Schedule Parameters

The following parameters on the DEFINE and UPDATE commands apply both to administrative command and client schedules:

Schedule name

All schedules must have a unique name, which can be up to 30 characters.

Initial start date, time, and day

You can specify a past date, the current date, or a future date for the initial start date for a schedule with the STARTDATE parameter.

You can specify a start time, such as 6 p.m. with the STARTTIME parameter.

You can also specify the day of the week on which the startup window begins with the DAYOFWEEK parameter. If the start date and start time fall on a day that does not correspond to your value for the day of the week, the start date and time are shifted forward in 24-hour increments until the day of the week is satisfied.

If you select a value for the day of the week other than ANY, then depending on the values for PERIOD and PERUNITS, schedules may not be processed when you expect. Use the QUERY EVENT command to project when schedules will be processed to ensure that you achieve the desired result.

Duration of a startup window

You can specify the duration of a startup window, such as 12 hours, with the DURATION and DURUNITS parameters. The server must start the scheduled service within the specified duration but does not necessarily complete it within that period of time. If the schedule needs to be retried for any reason, the retry attempt must begin before the startup window elapses or the operation does not restart.

Make the window duration long enough so that all client nodes scheduled for that window have a chance to start the operation. You may have to set the window to a longer period if the number of client nodes processing the schedule is greater than the number of available scheduled sessions.

If the schedule does not start during the startup window, the server records this as a *missed event* in the database. To identify any schedules that may have been missed, you can get an exception report from the server for events. For more information, see “Querying Event Records” on page 239.

How often to run the scheduled service

You can set the schedule frequency based on a period of hours, days, weeks, months, or years with the PERIOD and PERUNITS parameters. To have weekly backups, for example, set the period to one week with PERIOD=1 and PERUNITS=WEEKS.

Expiration date

You can specify an expiration date for a schedule with the EXPIRATION parameter if the services it initiates are required for only a specific period of time. If you set an expiration date, the schedule is not used after that date, but it still exists. You must delete the schedule to remove it from the database.

Priority

You can assign a priority to schedules with the PRIORITY parameter. For example, if you define two schedules for one client node, and they have the same startup window, the server runs the schedule with the highest priority first. A schedule with a priority of 1 is started before a schedule with a priority of 3.

Specifying Administrative Command Schedule Parameters

The following parameters on the DEFINE and UPDATE commands apply only to administrative command schedules:

Administrative schedule name

If you are defining or updating an administrative command schedule, you **must** specify the schedule name.

Type of schedule

If you are updating an administrative command schedule, you **must** specify TYPE=ADMINISTRATIVE on the UPDATE command. If you are defining a new administrative command schedule, this parameter is assumed if the CMD parameter is specified.

Command

When you define an administrative command schedule, you **must** specify the complete command that is processed with the schedule with the CMD parameter. These commands are used to tune server operations or to start functions that require significant server or system resources. The functions include:

- Migration

- Reclamation
- Export and import
- Database backup

Whether or not the schedule is active

Administrative command schedules can be active or inactive when they are defined or updated. Active schedules are processed when the specified command window occurs. Inactive schedules are not processed until they are made active by an UPDATE SCHEDULE command with the ACTIVE parameter set to YES.

Example: Defining and Updating an Administrative Command Schedule

To schedule the backup of the ARCHIVEPOOL primary storage pool, enter:

```
define schedule backup_archivepool type=administrative
cmd='backup stgpool archivepool recoverypool'
active=yes startime=20:00 period=2
```

This command specifies that, starting today, the ARCHIVEPOOL primary storage pool is to be backed up to the RECOVERYPOOL copy storage pool every two days at 8 p.m.

To update the BACKUP_ARCHIVEPOOL schedule, enter:

```
update schedule backup_archivepool type=administrative
startime=22:00 period=3
```

Starting with today, the BACKUP_ARCHIVEPOOL schedule begins the backup every three days at 10 p.m.

Specifying Client Schedule Parameters

The following parameters on the DEFINE and UPDATE commands apply only to client schedules:

Domain name

A client schedule belongs to a policy domain. Only clients in that domain can use the schedule.

Type of action

The following actions are possible:

- Perform an incremental backup
- Perform a selective backup
- Archive selected files
- Restore selected files
- Retrieve selected files

- Issue a client command
- Run an executable script (called a macro by ADSM; also known as a command file, a batch file, or a script on different client operating systems)

Restrictions: Not all clients can run all scheduled operations, even though ADSM allows you to define the schedule on the server and associate it with the client. For example, a Windows 3.1 client cannot run a schedule when the action is to restore or retrieve files, issue a command, or run an executable script. A Macintosh client cannot run a schedule when the action is to restore or retrieve files, or run an executable script.

Which files or commands to process

For incremental backup operations, you can specify which file spaces to process with the OBJECTS parameter, or allow the server to perform the backup based on the default client domain specified in the client user options file. Users can specify a default client domain by using the DOMAIN option in the client user options file. For information about specifying the DOMAIN option, refer to *ADSM Using the Backup-Archive Client* for the appropriate client.

For selective backup, archive, restore, and retrieve operations, you must specify the files to process. You can use wildcard characters to select multiple files. The file spaces and file names must follow the naming conventions of the client node. Therefore, you may need to define different schedules for different platforms.

If you are scheduling a command, you must specify the entire command.

If you are scheduling the running of an executable script, you must specify the executable script file name.

Client options

You can specify options that are supplied to the DSMC command when the schedule is processed. You can specify most options from the client's option file. For more information, refer to the appropriate client manual.

When applicable, these options override the options specified by a client node after it has successfully contacted the server.

Do not include the following options because they have no effect on the execution of the scheduled command:

- MAXCMDRETRIES
- QUERYSCHEDPERIOD
- RETRYPERIOD
- SCHEDLOGNAME
- SCHEDMODE
- TCPCLIENTADDRESS
- TCPCLIENTPORT

To help you decide which client options and which file names or file spaces to specify when defining or updating a schedule, you can try them out during an unscheduled operation from the client node. For information about client options, refer to *ADSM Using the Backup-Archive Client* for the appropriate client.

Example: Defining a New Client Schedule

You can define a new schedule for backing up or archiving client nodes in a specified policy domain. When you define a schedule, you assign it to a specific policy domain. You can define more than one schedule for each policy domain.

To define a schedule for incremental backups for clients in the ENGPOLDOM policy domain, enter:

```
define schedule engpoldom engweekly action=incremental
period=1 perunits=weeks
```

This command sets the frequency for schedule ENGWEEKLY to one week. This frequency for scheduled incremental backups matches the backup copy group frequency of the management class in the STANDARD policy set of the ENGPOLDOM policy domain.

Example: Updating an Existing Client Schedule

You can update an existing client schedule for backing up or archiving client nodes in a specified policy domain.

To update the ENGWEEKLY client schedule, enter:

```
update schedule engpoldom engweekly period=5 perunits=days
```

The ENGWEEKLY schedule is updated so that the incremental backup period is now every five days.

Copying Schedules

You can create a new schedule by copying an existing client or administrative schedule. When you copy a schedule, ADSM copies the following information:

- A description of the schedule
- All parameter values from the original schedule

You can then update the new schedule to meet your needs. You can copy a client schedule to another policy domain or to a newly named schedule in the same policy domain.

When you copy a client schedule, none of the client node associations are copied to the new schedule. You must associate the new schedule with client nodes before it can be used. The associations for the old schedule are not changed. See “Associating Client Nodes with Schedules” on page 226 for more information.

To copy the WINTER client schedule that belongs to policy domain DOMAIN1 to DOMAIN2 and name the new schedule WINTERCOPY, enter:

```
copy schedule domain1 winter domain2 wintercopy
```

To copy the BACKUP_ARCHIVEPOOL administrative schedule and name the new schedule BCKSCHED, enter:

```
copy schedule backup_archivepool bcksched type=administrative
```

Deleting Schedules

When you delete a schedule, all associations with client nodes are also deleted. See “Associating Client Nodes with Schedules” on page 226.

To delete all schedules in the ENGPOLDOM policy domain, enter:

```
delete schedule engpoldom *
```

Managing Scheduled Event Records

Task	Required Privilege Class
Display information about events	Any administrator
Set the retention period for event records	System
Delete event records	System or unrestricted policy

Each scheduled administrative command and each scheduled client operation is called an *event*. All scheduled events, including their status, are tracked by the server.

Querying Event Records

To help manage schedules for client operations and administrative commands, you can request information about scheduled and completed events. You can request general or exception reporting queries.

- To get information about past and projected scheduled processes, use a general query. If the time range you specify includes the future, the query output shows which events should occur in the future based on current schedules.
- To get information about scheduled processes that did not complete successfully, use exception reporting.

To minimize the processing time when querying events:

- Minimize the time range

- For client schedules, restrict the query to those policy domains, schedules, and client node names for which information is required

Query events regularly to see which events did not run successfully. For example, you can issue the following command to find out which events were missed in the previous 24 hours, for the DAILY_BACKUP schedule in the STANDARD policy domain:

```
query event standard daily_backup begindate=-1 begintime=now
enddate=today endtime=now exceptionsonly=yes
```

Figure 41 shows an example of the results of this query. To find out why a schedule was missed or failed, you may need to check the schedule log on the client node itself. For example, a schedule can be missed because the scheduler was not started on the client node.

```
Scheduled Start      Actual Start         Schedule Name Node Name     Status
-------------------- -------------------- ------------- ------------- ---------
03/06/1996 20:30:00                       DAILY_BACKUP  ANDREA        Missed
03/06/1996 20:30:00                       DAILY_BACKUP  EMILY         Missed
```

Figure 41. Exception Report of Events

Figure 42 shows an example of a general report for client node GOODELL that is displayed after you enter:

```
query event standard weekly_backup node=goodell
enddate=today+7
```

```
Scheduled Start      Actual Start         Schedule Name Node Name     Status
-------------------- -------------------- ------------- ------------- ---------
03/09/1996 06:40:00  03/09/1996 07:38:09  WEEKLY_BACKUP GOODELL       Started
03/16/1996 06:40:00                       WEEKLY_BACKUP GOODELL       Future
```

Figure 42. General Report of Events

To query an event for an administrative command schedule, you must specify the TYPE=ADMINISTRATIVE parameter. Figure 43 on page 241 shows an example of the results of the following command:

```
query event * type=administrative
```

```
Scheduled Start         Actual Start            Schedule Name     Status
--------------------    --------------------    -------------     ---------
03/17/1996 14:08:11     03/17/1996 14:08:14     BACKUP_ARCHI-     Completed
                                                 VEPOOL
```

Figure 43. Query Results for an Administrative Schedule

Removing Event Records from the Database

You can specify how long event records stay in the database before the server deletes them. You can also manually remove event records from the database.

If you issue a query for event records that have been removed, the status of those events may appear as *Uncertain*. To ensure that you find out about any missed events before the event records are deleted from the database, you should query events at least as often as you delete records from the database.

Setting the Event Record Retention Period

You can specify the retention period for event records in the database. After the retention period passes, the server automatically removes the event records from the database. At installation, the retention period is set to 10 days.

To set the retention period to 15 days, enter:

```
set eventretention 15
```

Event records are automatically removed from the database after both of the following conditions are met:

- The specified retention period has passed
- The startup window for the event has elapsed

Deleting Event Records

Because event records are deleted automatically, you do not have to manually delete them from the database. However, you may want to manually delete event records to increase available database space.

To delete all event records written prior to 11:59 p.m. on June 30, 1996, enter:

```
delete event 06/30/1996 23:59
```

Managing Client Associations with Schedules

Task	Required Privilege Class
Associate client nodes with any client schedules	System, unrestricted policy, or restricted policy

Querying Associations

You can display information about which client nodes are associated with a specific schedule. For example, you should query an association before deleting a client schedule.

When you query the system for information about node associations, the server returns the following information:

- Name of the schedule
- Name of the policy domain to which the schedule belongs
- Names of the clients that are currently associated with the schedule

The following figure shows the report that is displayed after you enter:

```
query association engpoldom
```

```
Policy Domain Name: ENGPOLDOM
     Schedule Name: MONTHLY_BACKUP
  Associated Nodes: MAB SSTEINER

Policy Domain Name: ENGPOLDOM
     Schedule Name: WEEKLY_BACKUP
  Associated Nodes: MAB SSTEINER
```

Deleting Associations

When you delete the association of a client node to a client schedule, the client data is no longer managed according to the schedule. However, the remaining client nodes still use the schedule.

To delete the association of the ENGNOD client with the ENGWEEKLY schedule, enter:

```
delete association engpoldom engweekly engnod
```

Rather than delete a schedule, you may want to delete all associations to it and save the schedule for possible use in the future.

Part 5. Maintaining the Server

© Copyright IBM Corp. 1995, 1997

Chapter 12. Managing Server Operations

Administrators can manage server operations. These operations include such tasks as starting and halting the server, managing client sessions, and monitoring server information. The sections listed in the following table begin at the indicated pages.

Section	Page
Tasks:	
Starting, halting, or restarting the server	245
Managing client sessions	249
Disabling or enabling server access	251
Managing server processes	252
Varying disk volumes online or offline	253
Requesting information about server status	254
Setting the server name	255
Querying server options	255
Managing the activity log	255
Monitoring accounting records	257
Getting help on commands and error messages	259

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, "Interface Cross-Reference" on page 467 lists each command and shows if its function is also available on the administrative client GUI.

Starting, Halting, and Restarting the Server

Task	Required Privilege Class
Start, halt, and restart the server	System or operator

Starting the Server

You can start the server in console mode, as a background mode, or specify other modes as part of the dsmserv command. To start the server from the Sun Solaris window prompt, change to the /opt/IBMadsm-s/bin directory, and enter the following command:

```
./dsmserv
```

ADSM displays the following information when the server is started:

© Copyright IBM Corp. 1995, 1997

- Product licensing and copyright information
- Processing information about the server options file
- Communication protocol information
- Database and recovery log information
- Storage volume information
- Server generation date
- Progress and error messages encountered during installation

The following events occur when the server is started:

- The server invokes the communication methods specified in the server options file.
- The server uses the volumes specified in the dsmserv.dsk file for the database and recovery log to record activity. It also identifies storage pool volumes to be used.
- The server starts an ADSM server console session that is used to operate and administer the server until administrative clients are registered to the server.
- Uses the STANDARD policy that is shipped with ADSM.

Running the Server in Background Mode

You may choose to run the server in the background. When the server runs in the background, you control the server through your administrative client.

Attention: *Before* you run the server in the -quiet mode, ensure the following conditions exist:

1. An administrative node has been registered and granted system authority. See “Registering Administrators or Updating Information” on page 284.
2. The administrative client options file has been updated with the correct SERVERNAME and TCPPORT options.
3. The administrative client can access the ADSM server.

If you do not follow these steps, you cannot control the server. When this occurs, you can only stop the server by canceling the process, using the process number displayed at startup. You may not be able to take down the server cleanly without this process number.

To start the server running in the background, enter the following:

```
nohup ./dsmserv -quiet &
```

You can check your directory for the output created in nohup.out to determine if the server has started. The output is redirected to the file nohup.out, which can grow considerably over time.

Starting the Server in Other Modes

If desired, you can specify any combination of the following command-line switches as part of the dsmserv command:

-quiet
: The server runs as a foreground process and does not read commands from the server console. Output messages print to the standard output.

-noexpire
: Suppresses inventory expiration. See “Running Expiration Processing to Delete Expired Files” on page 213 for additional information.

-options *filename*
: Specifies an explicit options file name when running more than one server.

Running Multiple Servers on a Single Machine

To have multiple servers running on a single machine, issue the DSMSERV INSTALL command from different directories to create multiple pairs of recovery log and database files. Do not attempt to install the server executable files in more than one directory.

The following procedure shows how you can set up an additional ADSM server:

1. Determine the directory where you want the server files created, for example: /opt/myserver, and create that directory.

   ```
   mkdir /opt/myserver
   ```

2. Copy the dsmserv.opt file to your directory (myserver).

   ```
   cp /opt/IBMadsm-s/bin/dsmserv.opt dsmserv.opt
   ```

   Note: Ensure that the TCPPORT option is unique from all other ADSM servers.

3. Define your environment variables, for example:

   a. To define the DSMSERV_DIR, enter:

      ```
      DSMSERV_DIR=/opt/IBMadsm-s/bin
      export DSMSERV_DIR
      ```

   b. To define the DSMSERV_CONFIG to point to the server options file, enter:

      ```
      DSMSERV_CONFIG=/opt/myserver/bin/dsmserv.opt
      export DSMSERV_CONFIG
      ```

4. Format the database and recovery log files, for example:

   ```
   /opt/IBMadsm-s/bin/dsmfmt -m -db dbvol2 5
   /opt/IBMadsm-s/bin/dsmfmt -m -log logvol2 9
   ```

5. Create the database and recovery log in the desired directory for the new server, for example:

   ```
   /opt/IBMadsm-s/bin/dsmserv install 1 logvol2 1 dbvol2
   ```

6. You must be in the correct subdirectory to invoke the appropriate server with the dsmserv command.

Notes:

1. You will need additional license authorizations to run additional servers.
2. When the server is started, it searches the current directory for the existence of the DSMSERV.DSK file. If the file is found, the names of the recovery log and database files are used for server operation. If the DSMSERV.DSK file is not found in the current directory, ADSM issues an error message (ANR0212E) and server initialization stops.

Halting the Server

You can halt the server without warning if an unplanned operating system problem requires you to return control to the operating system.

When you halt the server, all processes are abruptly stopped and client sessions are canceled, even if they are not completed. Any in-progress transactions are rolled back when the server is restarted. When the server is halted, administrator activity is not possible.

If possible, halt the server only after current administrative and client node sessions have completed or canceled. To shut down the server without severely impacting administrative and client node activity with the server, you must:

1. Disable the server to prevent new client node sessions from starting, as described in “Disabling or Enabling Server Access” on page 251.
2. Query for session information to identify any existing administrative and client node sessions, as described in “Requesting Information about Client Sessions” on page 249.
3. Notify any existing administrative and client node sessions that you plan to shut down the server. ADSM does not provide a network notification facility; you must use external means to notify users.
4. Cancel any existing administrative or client node sessions, as described in “Canceling a Client Session” on page 250.
5. Find out if any other processes are running, such as server migration or inventory expiration, by using the QUERY PROCESS command. If a database backup process is running, allow it to complete before halting the server. If other types of processes are running, cancel them by using the CANCEL PROCESS command.
6. Halt the server to shut down all server operations by using the HALT command.

Note: The QUIESCE option on the HALT command is recommended *only* if you plan to do a database dump by using the DSMSERV DUMPDB command immediately after halting. Because ADSM supports online database backup (BACKUP DB command), the DSMSERV DUMPDB command should be rarely, if ever, needed.

Restarting the Server

To start the server after it has been halted, follow the instructions in “Starting the Server” on page 245.

When you restart the server after it has been halted, ADSM rolls back any operations that had been in process to ensure that the database remains in a consistent state.

Managing Client Sessions

Task	Required Privilege Class
Display information about client sessions	Any administrator
Cancel a client session	System or operator

A *client session* can be either an administrative or a client node session.

If you want to prevent clients from accessing the server for an extended period of time, use the LOCK and UNLOCK commands for client node and administrator sessions, or disable the server.

For information on locking or unlocking administrators from the server, see “Locking and Unlocking Administrators from the Server” on page 291. For information on locking or unlocking client nodes from the server, see “Locking and Unlocking Client Nodes” on page 296.

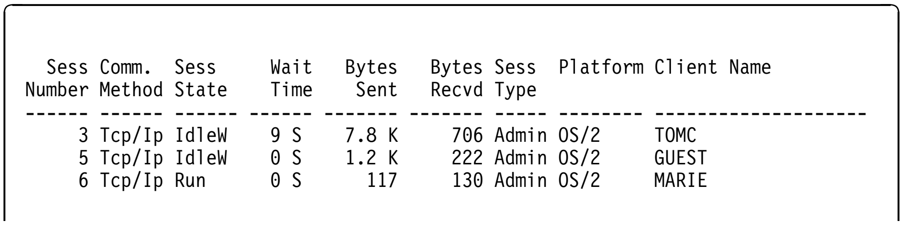
Requesting Information about Client Sessions

When administrators or users access ADSM, an administrative or client node session is established with the server. Each client session is assigned a unique session number.

To request information about client sessions, enter:

```
query session
```

Figure 44 shows a sample client session report.



```
  Sess Comm.  Sess     Wait   Bytes   Bytes Sess  Platform Client Name
Number Method State    Time    Sent   Recvd Type
------ ------ ------ ------ ------- ------- ----- -------- --------------------
     3 Tcp/Ip IdleW    9 S    7.8 K     706 Admin OS/2     TOMC
     5 Tcp/Ip IdleW    0 S    1.2 K     222 Admin OS/2     GUEST
     6 Tcp/Ip Run      0 S      117     130 Admin OS/2     MARIE
```

Figure 44. Information about Client Sessions

Check the *session state* and *wait time* to determine the session state of the server and how long (in seconds, minutes, or hours) the session has been in the current state. The server session state can be one of the following:

Start Connecting with a client session.

Run Executing a client request.

End Ending a client session.

RecvW Waiting to receive an expected message from the client while a database transaction is in progress. A session in this state is subject to the COMMTIMEOUT limit.

SendW Waiting for acknowledgement that the client has received a message sent by the server.

MediaW Waiting for removable media to become available.

IdleW Waiting for communication from the client, and a database transaction is NOT in progress. A session in this state is subject to the IDLETIMEOUT limit.

If a client does not initiate communication within the specified time limit set by the IDLETIMEOUT option in the server options file, then ADSM cancels the client session.

For example, if the IDLETIMEOUT option is set to 30 minutes, and a user does not initiate any operations within those 30 minutes, then ADSM cancels the client session. The client session is automatically reconnected to the server when it starts to send data again.

Canceling a Client Session

You may cancel a client session when:

- A user is unable to continue with work because the system is not responding
- You want all sessions canceled before halting the server

To cancel a client session, you must identify it by session number. You can display a session number by issuing a query for session information. For example, if the session number is 6, you cancel that session by entering:

```
cancel session 6
```

If you want to cancel all backup and archive sessions, enter:

```
cancel session all
```

If an operation, such as a backup or an archive process, is interrupted when you cancel the session, ADSM rolls back the results of the current transaction. That is, any changes made by the operation that are not yet committed to the database are undone. If necessary, the cancellation process may be delayed.

When user and administrator sessions are canceled, those persons must access the server again. If they were in the process of performing a function when the session was canceled, they must reissue their last command.

If the session you cancel is currently waiting for a media mount, the mount request is automatically canceled. If a volume associated with the client session is currently being mounted by an *automated* library, the cancel may not take effect until the mount is complete.

If the session is in the Run state when it is canceled, the cancellation process does not take place until the session enters the SendW, RecvW, or IdleW state.

Disabling or Enabling Server Access

Task	Required Privilege Class
Disable and enable client node access to the server	System or operator
Display server status	Any administrator

Disabling the server prevents users from establishing client node sessions with the server. This command does not affect system processes like migration and reclamation.

To disable the server, enter:

```
disable
```

When you disable the server, administrators can still access it, and current client node activity completes unless the user logs off or you cancel the client node session.

After the server has been disabled, you can enable the server to resume normal operations and allow users to access it by entering:

```
enable
```

You can issue the QUERY STATUS command to determine if the server is enabled or disabled.

Managing Server Processes

Task	Required Privilege Class
Display information about a server background process	Any administrator
Cancel a server process	System

When a user or administrator issues an ADSM command or uses a graphical user interface to perform an operation, the server starts a process. Some examples of an operation are registering a client node, deleting a management class, or canceling a client session.

Many processes occur quickly and are run in the foreground, while others take longer to complete. To allow you to perform other tasks during long-running operations, ADSM runs the following operations as background processes:

- Auditing an automated library
- Auditing licenses
- Auditing a volume
- Backing up the database
- Backing up a storage pool
- Defining a database volume copy
- Defining a recovery log volume copy
- Deleting a database volume
- Deleting a file space
- Deleting a recovery log volume
- Deleting a storage volume
- Expiring the inventory
- Exporting or importing data
- Extending the database or recovery log
- Migrating files from one storage pool to the next storage pool
- Moving data from a storage volume
- Reclaiming space from tape storage volumes
- Reducing the database or recovery log
- Restoring a storage pool
- Restoring a volume
- Varying a database or recovery log volume online

The server assigns each background process an ID number and displays the process ID when the operation starts. For example, if you issue an EXPORT NODE command, ADSM displays a message similar to the following:

```
EXPORT NODE started as Process 10
```

Requesting Information about Server Processes

You can request information about server background processes. If you know the process ID number, you can use the number to limit the search. However, if you do not know the process ID, you can display information about all background processes by entering:

```
query process
```

Figure 45 shows a server background process report after a DELETE FILESPACE command was issued. The report displays a process ID number, a description and a completion status for each background process.

```
 Process Process Description      Status
  Number
-------- ------------------------ ---------------------------------------------
       2 DELETE FILESPACE         Deleting filespace DRIVE_D for node CLIENT1:
                                   172 files deleted.
```

Figure 45. Information about Background Processes

Canceling Server Processes

You can cancel a server background process by specifying its ID number in the following command:

```
cancel process 2
```

You can issue the QUERY PROCESS command to find the process number. See “Requesting Information about Server Processes” for details.

If the process you want to cancel is currently waiting for a tape volume to be mounted (for example, a process initiated by EXPORT, IMPORT, or MOVE DATA commands), the mount request is automatically canceled. If a volume associated with the process is currently being mounted by an *automated* library, the cancel may not take effect until the mount is complete.

Varying Disk Volumes Online or Offline

Task	**Required Privilege Class**
Vary a disk volume online or offline	System or operator

To perform maintenance on a disk volume or to upgrade disk hardware, you can vary a disk volume offline. For example, to vary the disk volume named adsm/storage/pool001 offline, enter:

```
vary offline adsm/storage/pool001
```

If ADSM encounters a problem with a disk volume, the server automatically varies the volume offline.

After you have replaced the disk volume, you can make it available to the server by varying the volume online. For example, to make the disk volume named adsm/storage/pool001 available to the server, enter:

```
vary online adsm/storage/pool001
```

Requesting Information about Server Status

Any administrator can request information about the general server parameters defined by SET commands. To query the status of the server, enter:

```
query status
```

ADSM displays information about the server, such as:

- When the server was installed
- Whether the server is enabled or disabled
- Whether client registration is open or closed
- Whether passwords are required for client/server authentication
- How long passwords are valid
- Whether accounting records are being generated
- How long messages remain in the activity log before being deleted
- How many client sessions can concurrently communicate with the server
- How many client node sessions are available for scheduled work
- What percentage of the scheduling startup window is randomized
- What scheduling mode is being used
- How frequently client nodes can poll for scheduled work
- How many times and how frequently a client node can retry a failed attempt to perform a scheduled operation
- How long event records are retained in the database

Setting the Server Name

Task	Required Privilege Class
Specify the server name	System

At installation, the server name is set to ADSM. After installation, you can use the SET SERVERNAME command to change the server name. You can use the QUERY STATUS command to see the name of the server.

To specify the server name as WELLS_DESIGN_DEPT., for example, enter the following:

```
set servername wells_design_dept.
```

Querying Server Options

Task	Required Privilege Class
Query server options	Any administrator

Use the QUERY OPTION command to display information about one or more server options.

You can issue the QUERY OPTION command with no operands to display general information about all defined server options. You also can issue the QUERY OPTION command with a specific option name or pattern-matching expression to display information on one or more server options.

To display general information about all defined server options, enter:

```
query option
```

You can set options by editing the server options file (see *ADSM Administrator's Reference*).

Managing the Activity Log

Task	Required Privilege Class
Change the size of the activity log	System or unrestricted storage
Set the activity log retention period	System
Monitor the activity log	Any administrator

The activity log contains all messages normally sent to the server console during server operation. The only exceptions are responses to commands entered at the console, such as responses to QUERY commands. Examples of messages sent to the activity log include:

- When client sessions start or end
- When migration starts and ends
- When backup versions are expired
- What data is exported to tape
- When expiration processing is performed
- What export or import processing is performed

Any error messages sent to the server console are also stored in the activity log.

Use the following sections to adjust the size of the activity log, set an activity log retention period, and request information about the activity log.

Changing the Size of the Activity Log

Because the activity log is stored in the database, the size of the activity log should be factored into the amount of space allocated for the database. Allow at least 1MB of additional space for the activity log.

The size of your activity log depends on how many messages are generated by daily processing operations and how long you want to retain those messages in the activity log. When retention time is increased, the amount of accumulated data also increases requiring additional database storage.

When there is not enough space in the database or recovery log for activity log records, ADSM stops recording and sends messages to the server console. If you increase the size of the database or recovery log, ADSM starts activity log recording again. For information about increasing the size of the database or recovery log, see “Adding Space to the Database or Recovery Log” on page 266.

If you do not have enough space in the database for the activity log, you can do one of the following:

- Allocate more space to the database
- Reduce the length of time that messages are retained in the activity log

Setting the Activity Log Retention Period

You can specify how long activity log information is retained in the database by using the SET ACTLOGRETENTION command.

The server automatically deletes messages from the activity log after they have passed the specified age. At installation, the activity log retention period is set to one day. To change the retention period to 30 days, for example, enter:

```
set actlogretention 30
```

You can display the current retention period for the activity log by querying the server status.

Requesting Information from the Activity Log

You can request information stored in the activity log. To minimize processing time when querying the activity log, you can:

- Specify a time period in which messages have been generated. The default for the QUERY ACTLOG command shows all activities that have occurred in the previous hour.
- Specify the message number of a specific message or set of messages.
- Specify a string expression to search for specific text in messages.
- Specify the QUERY ACTLOG command from the command line for large queries instead of using the graphical user interface.

For example, to review messages generated on May 30 between 8 a.m. and 5 p.m., enter:

```
query actlog begindate=05/30/1996 enddate=05/30/1996
begintime=08:00 endtime=17:00
```

To request information about messages related to the expiration of files from the server storage inventory, enter:

```
query actlog msgno=0813
```

Refer to *ADSM Messages* for message numbers.

To request information about messages generated from the IMPORT NODE command, enter:

```
query actlog search='import node'
```

Monitoring ADSM Accounting Records

General-use programming interface

Task	Required Privilege Class
Set accounting records on or off	System

ADSM accounting records show the server resources used during a session. This information lets you track resources used by a client node session. At installation, accounting is set off. You can set accounting on by entering:

```
set accounting on
```

When accounting is set on, the server creates a session resource usage accounting record whenever a client node session ends.

Accounting records are stored in a file, *dsmaccnt.log*, in the directory from which the server is started. The file contains text records that can be viewed directly or can be read into a spreadsheet program.

The file remains opened while the server is running and accounting is set on. The file continues to grow until you delete it or prune old records from it. To close the file for pruning, either temporarily set accounting off or halt the server.

There are 25 fields, which are delimited by commas (,). Each record ends with a new-line character. Each record contains the following information:

Field	**Contents**
1	Product level
2	Product sublevel
3	Product name, ‘ADSM’
4	Date of accounting (mm/dd/yyyy)
5	Time of accounting (hh:mm:ss)
6	Node name of ADSM client
7	Client owner name (UNIX)
8	Client Platform
9	Authentication method used
10	Communication method used for the session
11	Normal server termination indicator (Normal=X'01', Abnormal=X'00')
12	Number of archive database objects inserted during the session
13	Amount of archived files, in kilobytes, sent by the client to the server
14	Number of archived database objects retrieved during the session
15	Amount of space, in kilobytes, retrieved by archived objects
16	Number of backup database objects inserted during the session
17	Amount of backup files, in kilobytes, sent by the client to the server
18	Number of backup database objects retrieved during the session
19	Amount of space, in kilobytes, retrieved by backed up objects
20	Amount of data, in kilobytes, communicated between the client node and the server during the session
21	Duration of the session, in seconds
22	Amount of idle wait time during the session, in seconds
23	Amount of communications wait time during the session, in seconds
24	Amount of media wait time during the session, in seconds
25	Client session type. A value of 1 or 4 indicates a general client session. A value of 5 indicates a client session that is running a schedule.

26 Number of space-managed database objects inserted during the session

27 Amount of space-managed data, in kilobytes, sent by the client to the server

28 Number of space-managed database objects retrieved during the session

29 Amount of space, in kilobytes, retrieved by space-managed objects

The following shows an example of two records:

```
0,8,ADSM,06/03/1996,16:26:37,node1,,AIX,1,Tcp/Ip,0,254,1713,0,0,47,1476,0,0,3316,960,27,5,1,4,0,0,0,0
0,8,ADSM,06/03/1996,18:01:48,node2,,OS/2,1,Tcp/Ip,1,85,610,0,0,53,611,0,0,2133,78,48,6,1,4,0,0,0,0
```

End of General-use programming interface

Getting Help on Commands and Error Messages

Any administrator can issue the HELP command to display information about administrative commands and messages from the server and the administrative command line client.

You can issue the HELP command with no operands to display a menu of help selections. You also can issue the HELP command with operands that specify help menu numbers, commands and subcommands, or message numbers.

To display the help menu, enter:

```
help
```

To display help information on the REMOVE commands, enter:

```
help remove
```

To display help information on a specific message, for example ANR0992I, enter:

```
help 0992
```

Additional information is also available in the online documentation.

Chapter 13. Managing the Database and Recovery Log

Task	Required Privilege Class
Manage disk volumes used by the database and recovery log	System or unrestricted storage
Display information about the database and recovery log	Any administrator

The sections listed in the following table begin at the indicated pages.

Section	Page
Concepts:	
Database and recovery log	261
Tasks:	
Estimating database or recovery log space requirements	264
Adding space to the database or recovery log	266
Deleting space from the database or recovery log	271
Optimizing the performance of the database or recovery log	274

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, "Interface Cross-Reference" on page 467 lists each command and shows if its function is also available on the administrative client GUI.

Database and Recovery Log

The database, recovery log, and storage pool volumes are closely related. The ADSM database contains information needed for server operations and information about client data that has been backed up, archived, and space-managed.

Note: The client data itself is stored in storage pools, not in the database.

The database contains pointers to the locations of all client files in the ADSM storage pools.

Changes to the database are recorded in the recovery log in order to maintain a consistent database image. These changes are the result of transactions between clients and the server. Examples of activities that can occur in a transaction are: defining a management class or copy group, archiving or backing up a client file, and registering an administrator or a client node.

© Copyright IBM Corp. 1995, 1997

The database contains:

- Information about client nodes and administrators
- Policies and schedules
- Server settings
- Locations of client files on server storage
- Information about server operations (for example, activity logs and event records)

The recovery log contains information about updates that have not yet been committed to the database.

Note: If the database is unusable, the entire ADSM server is unavailable. If a database is lost and cannot be recovered, the backup, archive, and space-managed data for that server is lost. See Chapter 16, “Protecting and Recovering Your Data” on page 327 for steps that you can take to protect your database.

How ADSM Processes Transactions

Both the database and the recovery log have buffer pools. To support multiple transactions from concurrent client sessions, the server holds transaction log records in the recovery log buffer pool until they can be written to the recovery log. These records remain in the buffer pool until the active buffer becomes full or ADSM forces log records to the recovery log.

Changes resulting from transactions are held in a buffer pool temporarily and not made to the database immediately. Therefore, the database and recovery log are not always consistent.

When all log records for a transaction are written to the recovery log, the server updates the database. The transaction is then committed to the database. At some point after a transaction is committed, the server deletes the transaction record from the recovery log.

How Space is Managed by the Server

ADSM tracks all volumes defined to the database as one logical volume and all volumes defined to the recovery log as another logical volume. For example, in Figure 46, the database consists of four volumes: VOL1 through VOL4. ADSM tracks the database as a single logical volume.

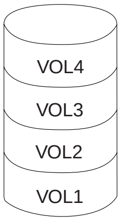


Figure 46. A Server Database

To manage the database and recovery log effectively, you must understand the following concepts:

- Available space, page 263
- Assigned capacity, page 263
- Utilization, page 263

Available Space

Not all of the space that is allocated for the database or recovery log volumes is available to be used for database and recovery log information. To calculate the available space, ADSM:

- Subtracts 1MB from each physical volume for overhead.
- Divides the remaining space into 4MB partitions. Any remaining space on a volume is unusable.

See “Step 1: Allocating Space for the Database and Recovery Log” on page 267 for an example of how this calculation is used.

Assigned Capacity

Assigned capacity is the portion of available space that can be used for database or recovery log information. During installation, the server automatically extends the database and recovery log so that assigned capacity matches the available space.

If you add volumes after installation, you increase your available space. However, to increase the assigned capacity, you must also extend the database or recovery log. See “Step 3: Extending the Capacity of the Database or Recovery Log” on page 270 for details.

Utilization

Utilization is the percent of the database or recovery log assigned capacity used at a specific time. *Maximum percent utilized* is the highest utilization since the utilization statistics were last reset.

For example, an installation performs most backups after midnight. Figure 47 on page 264 shows that utilization statistics for the recovery log were reset at 9 p.m. the previous evening and that the maximum utilization occurred at 12 a.m.

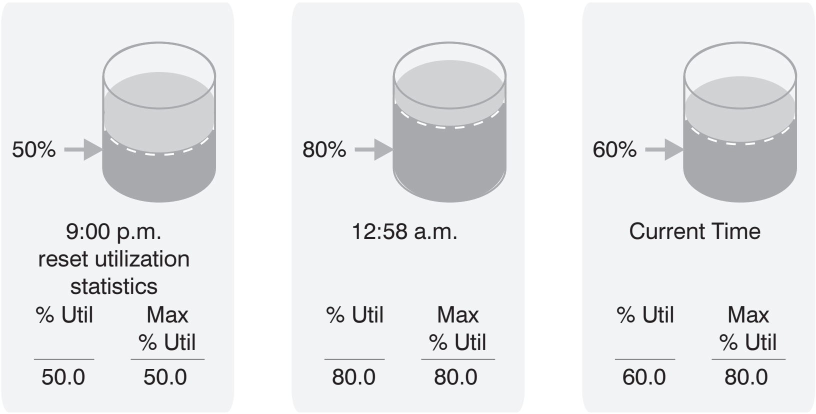


Figure 47. An Example of Recovery Log Utilization

Unless an unusually large number of objects are deleted, the database maximum percent utilized is usually close to the utilization percentage.

Estimating and Monitoring Database and Recovery Log Space Requirements

The size of your ADSM database depends on the number of client files to be stored and how ADSM manages them. If you can estimate the maximum number of files that might be in server storage at any time, you can use the following information to come up with a useful database size estimate:

- Each **version of a file** that ADSM stores requires from 500 to 600 bytes of database space.
- Each **cached** or **copy storage pool copy** of a file requires from 150 to 200 bytes of database space.
- **Overhead** requires additional database space.

For example, assume the following for your installation:

Versions of files

Backed up files
Up to 500000 client files might be backed up. And storage policies call for retaining up to 3 copies of backed up files:

```
500000 files x 3 copies = 1500000 files
```

Archived files
Up to 100000 files might be archived copies of client files.

Space-managed files
Up to 200000 files migrated from client workstations might be in server storage.

The space required for all backed up, archived, and space-managed files at 600 bytes per file is:

```
(1500000 + 100000 + 200000) x 600 = 1.0GB
```

Cached and copy storage pool files

Cached copies
Caching is enabled in the disk storage pool. The disk pool has a capacity of 5GB and uses the default high migration threshold (90%) and low migration threshold (70%). Thus, if migration begins at 90% and stops at 70%, 20% of the disk pool, or 1GB is occupied by cached files.

If the average file size is about 10KB, about 100000 files are in cache at any one time.

```
100000 files x 200 bytes = 19MB
```

Copy storage pool files
All primary storage pools are backed up to the copy storage pool:

```
(1500000 + 100000 + 200000) x 200 bytes = 343MB
```

Cached and copy storage pool files, then, require about 0.4GB of database space.

Overhead
Up to this point approximately 1.4GB is required for file versions and cached and copy storage pool files. Up to 50% additional space (or 0.7GB) should be allowed for overhead.

The database, then, should be approximately 2.1GB.

If it is not practical to estimate the number of files to be covered by your storage management policies, you can roughly estimate the database size as from 1% to 5% of the required server storage space. For example, if you need 100GB of server storage, your database should be between 1GB and 5GB. See “Estimating Space Needs for Storage Pools” on page 135 for details.

The size of the recovery log depends on the number of concurrent client sessions and the number of background processes executing on the server.

Note: The maximum number of concurrent client sessions is set in the server options.

Begin with at least 12MB for the recovery log. If you will be using the database backup and recovery functions in roll-forward mode, you should begin with at least 25MB. See “Database Backup” on page 329 and “Estimating the Size of the Recovery Log” on page 337 for more information.

Monitoring the Database and Recovery Log

After your ADSM system is operational, you should monitor the database and recovery log to see if you should add or delete space.

You can reset the maximum utilization counters for the database and recovery log to monitor daily utilization. To set the maximum utilization percentage equal to the current utilization, you might want to reset the utilization statistics each day.

Utilization statistics are reset in two ways:

- Automatically when the server is restarted
- By issuing the RESET DBMAXUTILIZATION or RESET LOGMAXUTILIZATION commands

For example, to reset the maximum utilization statistic for the database, enter:

```
reset dbmaxutilization
```

To display information about the database or recovery log, issue the QUERY DB or QUERY LOG respectively. For example:

```
query db
```

The server displays a report, like this:

```
Available Assigned   Maximum   Maximum    Page     Total      Used %Util  Max.
    Space Capacity Extension Reduction    Size     Pages     Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)
--------- -------- --------- --------- ------- --------- --------- ----- -----
       96       96         0        92   4,096    24,576        86   0.3   0.3
```

See the indicated pages for details about the following entries:

- Available space, page 263
- Assigned capacity, page 263
- Utilization and maximum utilization, page 263

On the basis of the these queries, you may decide to adjust the size of the database or recovery log. If utilization is high, you may want to add space. If utilization is low, you may want to delete space. See “Adding Space to the Database or Recovery Log” or “Deleting Space from the Database or Recovery Log” on page 271.

Adding Space to the Database or Recovery Log

During the ADSM server installation, you allocated space for the database and recovery log and defined the allocated physical volumes to the server. However, you can define additional volumes and extend the capacity of the database or recovery log. You can add or delete database or recovery log volumes while the server is running.

Attention: You must not change the size of an allocated database or recovery log volume after it has been defined to ADSM. If you change the size of a volume, ADSM may not initialize correctly, and data may be lost.

To add space to the database or recovery log perform the following steps:

"Step 1: Allocating Space for the Database and Recovery Log"
"Step 2: Defining Database or Recovery Log Volumes to ADSM" on page 268
"Step 3: Extending the Capacity of the Database or Recovery Log" on page 270

Step 1: Allocating Space for the Database and Recovery Log

The size of the database or recovery log volumes affects space utilization, as is shown in the following examples:

Example 1: An Inefficient Allocation of Space

You allocate four 24MB volumes for the database. For each volume, ADSM:

- Subtracts 1MB for overhead, leaving 23MB of available space
- Divides the 23MB into five 4MB partitions and 3MB of unused space

The available space is only 80MB out of the allocated 96MB.

Example 2: A More Efficient Allocation of Space

You allocate four 25MB volumes for the database. For each volume, ADSM:

- Subtracts 1MB of overhead, leaving 24MB of available space
- Divides the 24MB into six 4MB partitions and no unused space

The available space for the database logical volume is 96MB out of the allocated 100MB, as shown in Figure 48.

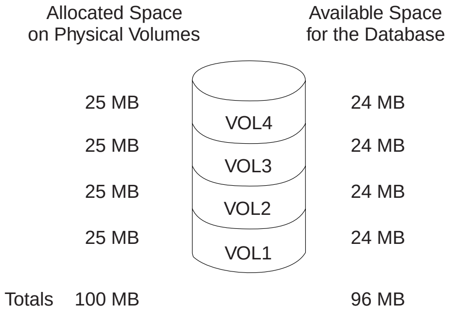


Figure 48. An Example of Available Space

Notes:

1. For performance reasons, define more than one volume for the database and recovery log, and place these volumes on separate disks to allow simultaneous access to different parts of the database or recovery log.
2. To protect database and recovery log volumes from media failure, you can use the mirroring feature. See "Mirroring the Database and Recovery Log" on page 333 for information on the mirroring feature.
3. To use disk space efficiently, allocate a few large disk volumes rather than many small disk volumes. In this way, you avoid losing space to ADSM overhead processing.

   If you already have a number of small volumes and want to consolidate the space into one large volume, see "Deleting Space from the Database or Recovery Log" on page 271.

Using the DSMFMT Command to Format Volumes

When you use the DSMFMT command from an Sun Solaris command line to format volumes, you must perform the next two steps: "Step 2: Defining Database or Recovery Log Volumes to ADSM" and "Step 3: Extending the Capacity of the Database or Recovery Log" on page 270.

To allocate an additional 101MB to the database as volume VOL5, enter:

```
./dsmfmt -db vol5 101
```

Creating a Raw Partition for a Volume

Another option for preparing a volume is to create a raw partition on a formatted disk. For details, see the operating system's format command. After the partition is created, you define the volume to ADSM.

Attention: ADSM cannot always detect if data already exists on a raw partition. Existing data can be destroyed if you define a volume to that partition.

Step 2: Defining Database or Recovery Log Volumes to ADSM

To define a database volume named VOL5, enter:

```
define dbvolume vol5
```

When VOL5 is defined, it becomes a part of the logical view of the server database. Thus, the server still sees a single logical database volume, which is now composed of five physical volumes. Because 1MB from VOL5 is used for overhead process, 100MB is added to the database to increase the available space to 196MB. However, the assigned capacity remains at 96MB, and ADSM cannot use the space until the capacity is extended (see “Step 3: Extending the Capacity of the Database or Recovery Log” on page 270).

After you define your volumes, you can verify the change by querying the database or recovery log. For example, to query the database, enter:

```
query db
```

The server displays a report, like this:

```
Available Assigned   Maximum   Maximum    Page     Total      Used %Util  Max.
    Space Capacity Extension Reduction    Size     Pages     Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)
--------- -------- --------- --------- ------- --------- --------- ----- -----
      196       96       100        92   4,096    24,576        86   0.3   0.3
```

In the information displayed, the value in the *maximum extension* field should equal the available space of the new volume. In this example, a 101MB volume was allocated. This report shows that the available space has increased by 100MB; the assigned capacity is unchanged at 96MB; and the maximum extension is 100MB. Figure 49 illustrates these changes.

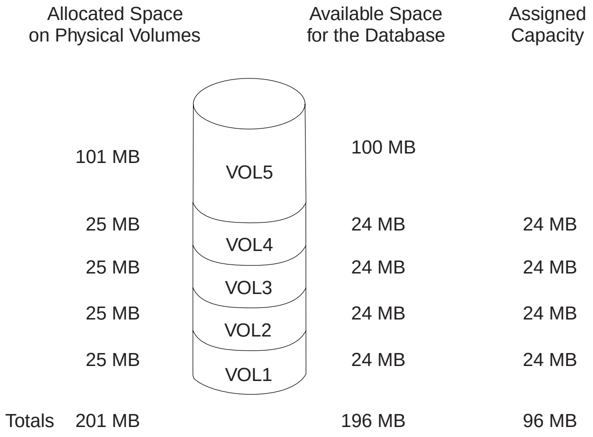


Figure 49. Adding Volumes Increases Available Space

You can also use the QUERY DBVOLUME and QUERY LOGVOLUME commands to display information about the physical volumes that make up the database and recovery log.

Step 3: Extending the Capacity of the Database or Recovery Log

You must extend the database or recovery log in 4MB increments. If you do not specify the extension in 4MB increments, ADSM rounds up to the next 4MB partition. Thus, if you specify 1MB, ADSM extends the capacity by 4MB.

For example, to increase the capacity of the database by 100MB, enter:

```
extend db 100
```

When you extend the database or recovery log, ADSM starts a background process to format the new space. You can issue a QUERY PROCESS command to check on the status of the process.

The result of this command is that the assigned capacity of the database is increased by 100MB, and now equals the available space, as shown in Figure 50.

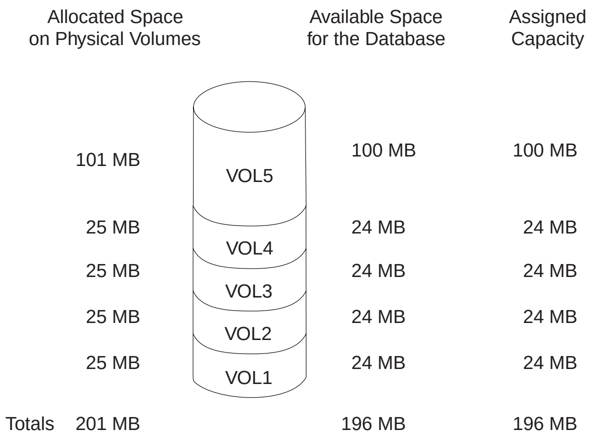


Figure 50. Extending the Capacity of the Database

You can query the database or recovery log (QUERY DB and QUERY LOG commands) to verify their assigned capacities. The server would display a report, like this:

```
Available Assigned   Maximum   Maximum    Page     Total      Used %Util  Max.
    Space Capacity Extension Reduction    Size     Pages     Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)
--------- -------- --------- --------- ------- --------- --------- ----- -----
      196      196         0       192   4,096    50,176       111   0.2   0.2
```

After the database has been extended, the available space and assigned capacity are both equal to 196MB.

Deleting Space from the Database or Recovery Log

You may want to delete database or recovery log volumes for a number of reasons:

- You have a significant amount of space that is unused.
- You want to consolidate a number of small volumes, each of which may have unusable space (see “Step 1: Allocating Space for the Database and Recovery Log” on page 267 for details).

When a database or recovery log volume is deleted, the server tries to move any data on the volume being deleted to the other physical volumes that make up the logical database or recovery log.

To delete space, perform the following steps:

1. Determine if you can delete one or more volumes (page 271).
2. As needed, reduce the capacity of the database to free up existing space in the database or recovery log, as described in “Step 2: Reducing the Capacity of the Database or Recovery Log” on page 272.
3. Delete the volume (page 273).

Step 1: Determining If Volumes Can Be Deleted

To determine if volumes can be deleted from the database or recovery log, check the volume sizes and the amount of unused space.

To check the sizes of the volumes in the database, enter:

```
query dbvolume format=detailed
```

The server displays the following type of information:

```
Volume Name (Copy 1): VOL1
         Copy Status: Sync'd
Volume Name (Copy 2):
         Copy Status: Undefined
Volume Name (Copy 3):
         Copy Status: Undefined
Available Space (MB): 24
Allocated Space (MB): 24
     Free Space (MB): 0
   .
   .
   .
```

In this example, you determine that VOL1, VOL2, VOL3, and VOL4 each have 24MB of available space, and VOL5 has 100MB.

To determine if there is enough unused space to delete one or more volumes, enter:

```
query db
```

The server displays the following type of report. Check the *Maximum Reduction* column for the amount of assigned capacity not being used.

```
Available Assigned   Maximum   Maximum    Page     Total      Used %Util  Max.
    Space Capacity Extension Reduction    Size     Pages     Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)
--------- -------- --------- --------- ------- --------- --------- ----- -----
      196      196         0       176   4,096    50,176     4,755   9.5   9.5
```

In this example, the database could be reduced by up to 176MB. This is enough space to allow the deletion of VOL1, VOL2, VOL3, and VOL4.

If there is not enough space on the remaining volumes, allocate more space and define an additional volume, as described in “Step 1: Allocating Space for the Database and Recovery Log” on page 267 and “Step 2: Defining Database or Recovery Log Volumes to ADSM” on page 268 and continue with “Step 2: Reducing the Capacity of the Database or Recovery Log.”

Step 2: Reducing the Capacity of the Database or Recovery Log

The *maximum reduction* identifies by how much you can reduce the database or recovery log. By reducing the database or recovery log, you might be able to free up enough space to delete a volume.

You can reduce the capacity of the database or recovery log in 4MB increments. If you do not reduce in 4MB increments, ADSM rounds up to the next 4MB partition. Thus, if you specify 5MB, ADSM reduces the capacity by 8MB.

For example, assume that based on the utilization of the database, VOL5 alone could contain all the data. To reduce the database by the amount of available space in VOL1 through VOL4, 96MB, enter:

```
reduce db 96
```

Reducing capacity is run as a background process and can take a long time. You can issue a QUERY PROCESS command to check on the status of the process.

You can query the database to verify how much unused space is available after reduction. For example, after reducing the database by 96MB, the assigned capacity is 100MB and the maximum extension is 96MB, as shown in the following example:

```
Available Assigned   Maximum   Maximum    Page     Total      Used %Util  Max.
    Space Capacity Extension Reduction    Size     Pages     Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)
--------- -------- --------- --------- ------- --------- --------- ----- -----
      196      100        96        92   4,096    24,576        86   0.3   0.3
```

Step 3: Deleting a Volume from the Database or Recovery Log

After you reduce the database or recovery log, use the smaller size for a few days. If the maximum utilization does not go over 70%, you can delete extra volumes.

Notes:

1. You cannot delete volumes if there is not enough free space for the server to move existing data from the volume being deleted to other physical volumes in the database or recovery log.
2. You cannot delete the last volume of the database or recovery log.

In our example, you determined that you can delete the four 24MB volumes from the database. You have reduced the database by 96MB. To delete VOL1 through VOL4 from the database, enter:

```
delete dbvolume vol1

delete dbvolume vol2

delete dbvolume vol3

delete dbvolume vol4
```

When you request that volumes be deleted from the database or recovery log, the server moves existing data from the volumes being deleted to available space on other volumes. Figure 51 on page 274 shows data moved from VOL1, VOL2, VOL3, and VOL4 to available space on VOL5.

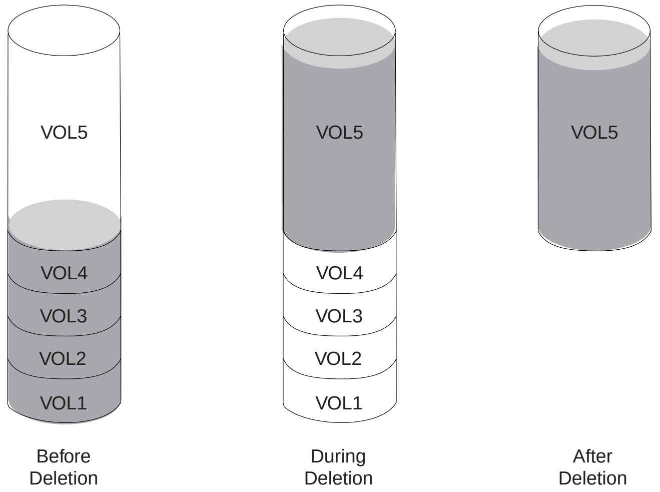


Figure 51. Deleting Database Volumes

After the data has been moved, these volumes are deleted from the server.

Optimizing the Performance of the Database or Recovery Log

The size of the database and recovery log buffer pools can affect performance at the cost of greater memory. For example, a large database buffer pool can improve performance, and a large recovery log buffer pool reduces how often the server forces records to the recovery log.

Adjusting the Database Buffer Pool

You can adjust the size of the database buffer pool by updating the server option for it.

Step 1: Resetting Database Buffer Pool Utilization Statistics

To gather statistics on database use, reset the buffer pool statistics on a regular basis and chart the results. Initially, you might want to monitor the database twice a day. Later, when most client nodes have been registered to the server, you can reset statistics each week. To reset the database buffer pool, enter:

```
reset bufpool
```

Step 2: Requesting Information about the Database Buffer Pool

To see if the database buffer pool is adequate for database performance, enter:

```
query db format=detailed
```

The server displays a report, like this:

```
  Available Space (MB): 196
Assigned Capacity (MB): 196
Maximum Extension (MB): 0
Maximum Reduction (MB): 176
     Page Size (bytes): 4,096
           Total Pages: 50,176
            Used Pages: 4,755
                 %Util: 9.5
            Max. %Util: 9.5
      Physical Volumes: 5
     Buffer Pool Pages: 128
 Total Buffer Requests: 1,193,212
        Cache Hit Pct.: 99.73
       Cache Wait Pct.: 0.00
```

Use the following fields to evaluate your current use of the database buffer pool:

Buffer Pool Pages

The number of pages in the database buffer pool. This value is determined by the server option for the size of the database buffer pool. At installation, the database buffer pool is set to 512KB, which equals 128 database pages.

Total Buffer Requests

The number of requests for database pages since the server was last started or the buffer pool was last reset. If you regularly reset the buffer pool, you can see trends over time.

Cache Hit Pct

The percentage of requests for cached database pages in the database buffer pool that were not read from disk.

A high *cache hit percentage* indicates that the size of your database buffer pool is adequate. If the cache hit percentage is below 90%, consider increasing the size of the database buffer pool.

Cache Wait Pct

The percentage of requests for database pages that had to wait for a buffer to become available in the database buffer pool.

When the cache wait percentage is greater than 0, increase the size of the database buffer pool.

Step 3: Set the Size of the Database Buffer Pool

You can set the size of the database buffer pool by setting the buffer pool size option (BUFPOOLSIZE). You can set options through the ADSM Server Utilities or by editing the server options file (see *ADSM Administrator's Reference*).

Adjusting the Recovery Log Buffer Pool

You can adjust the size of the recovery log buffer pool by updating the server option for it.

Step 1: Requesting Information about the Recovery Log Buffer Pool

To see how the buffer pool size affects recovery log performance, enter:

```
query log format=detailed
```

The server displays a report, like this:

```
  Available Space (MB): 12
Assigned Capacity (MB): 12
Maximum Extension (MB): 0
Maximum Reduction (MB): 8
     Page Size (bytes): 4,096
           Total Pages: 3,072
            Used Pages: 227
                 %Util: 7.4
            Max. %Util: 69.6
      Physical Volumes: 1
        Log Pool Pages: 32
    Log Pool Pct. Util: 6.25
    Log Pool Pct. Wait: 0.00
```

Use the following fields to optimize the log buffer pool size for your installation:

Log Pool Pages
The number of pages in the recovery log buffer pool. This value is set by the server option for the size of the recovery log buffer pool. At installation, the default setting is 128KB, which equals 32 recovery log pages.

Log Pool Pct. Util
The percentage of pages used to write changes to the recovery log after a transaction is committed.

A low value (under 10%) indicates that the size of your recovery log buffer pool is adequate. As this number increases, consider increasing the size of the recovery log buffer pool.

Log Pool Pct. Wait

The percentage of requests for a page that is not available because all pages are waiting to write to the recovery log.

If the *log pool percentage wait* value is greater than 0, increase the size of the recovery log buffer pool.

Step 2: Setting the Size of the Recovery Log Buffer Pool

You can set the size of the recovery log buffer pool by setting the buffer pool size option (LOGPOOLSIZE). You can set options through the ADSM Server Utilities or by editing the server options file (see *ADSM Administrator's Reference*).

Chapter 14. Managing Licensing, Privilege Classes, and Registration

This section provides the information necessary for a system administrator to control authorization and access to the server. The sections listed in the following table begin at the indicated pages.

Section	Page
Tasks:	
Managing ADSM licenses	279
Ensuring client/server authentication	283
Registering administrators or updating information	284
Granting administrative authority	285
Revoking or reducing administrative authority	289
Managing administrator access	290
Managing client node registration	292
Registering an application programming interface to the server	300
Managing client node access	295
Requesting information about client nodes	296
Requesting information about file spaces	298
Deleting file spaces and client nodes from the server	299

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, “Interface Cross-Reference” on page 467 lists each command and shows if its function is also available on the administrative client GUI.

Managing ADSM Licenses

Task	Required Privilege Class
Register licenses Audit licenses Schedule automatic license audits	System
Display license information	Any administrator

If an ADSM system exceeds the terms of its license agreement, one of the following occurs:

© Copyright IBM Corp. 1995, 1997

- The server issues a warning message indicating that it is not in compliance with the licensing terms.
- Operations fail because the server is not licensed for specific features.

For details, see "License Compliance" on page 282. In either case, you must contact your IBM account representative or authorized reseller to modify your agreement.

Licensed Features

The base ADSM Sun Solaris server license supports an unlimited number of administrative clients and one Sun Solaris backup-archive client.

Notes:

1. In this licensing section, the term *client* is used to refer to backup-archive clients, unless otherwise noted.
2. For current information about supported clients and devices,
   - Check with your authorized reseller
   - Call the IBM Information Support Center at 1-800-IBM-3333 and ask for STAR 20
   - Send an E-mail note to askibm-rsvp@info.ibm.com with STAR 20 in the body of the note
   - Visit the ADSM page on the World Wide Web at this address:

     http://www.storage.ibm.com/adsm

You must register a new license if you want to make any of the following changes to your license agreement:

- Add support for additional clients. The base license allows for one Sun Solaris backup-archive client. If you want to add clients in an environment other than Sun Solaris, you must register a new license for that feature also (see the next item in this list). See "Registering Additional Clients."
- Add support for clients in environments other than Sun Solaris. The base license allows only for backup-archive clients on Sun Solaris. See "Registering Clients Other Than Sun Solaris" on page 281.
- Add support for storage devices not covered by the existing agreement. See "Registering Device Support Modules" on page 281.

The enrollment certificate files for all ADSM licenses are on the ADSM installation CD-ROM. You register those licenses you want by issuing the REGISTER LICENSE command with the name of the enrollment certificate file. When registered, the licenses are stored in a file named NODELOCK in the current directory.

Registering Additional Clients

You can register the server to support a specified number of clients beyond the one Sun Solaris backup-archive client supported by the base license. Those additional clients can be in any environment for which your system is licensed (see "Registering

Clients Other Than Sun Solaris” on page 281). The following enrollment certificate files for clients are available:

1client.lic	1 backup-archive client
5client.lic	5 backup-archive clients
10client.lic	10 backup-archive clients
50client.lic	50 backup-archive clients

For example, to register ten additional clients, obtain a 10-client enrollment certificate file and enter:

```
register license file(10client.lic)
```

To register 20 additional clients, you would simply issue the previous command twice.

If you register more clients than your server is licensed to support, the server issues a warning message. However, operations continue normally.

Registering Clients Other Than Sun Solaris

You can obtain licenses for environment support features that allow the server to support clients other than Sun Solaris.

Environment support features are:

network.lic	You must have this license if you are using a network communication method, for example, TCP/IP.
desktop.lic	Desktop clients (OS/2, Windows, Apple, Novell NetWare, and DOS)
unix.lic	Any UNIX clients
oemvs.lic	The OpenEdition MVS client
spaceman.lic	HSM clients

You can register any or all environment support features. For example, if you wanted to include HP-UX clients and HSM clients, issue the following commands:

```
register license file(network.lic)
register license file(unix.lic)
register license file(spaceman.lic)
```

Registering Device Support Modules

You can obtain licenses for device support modules that let the server support a variety of storage devices. Device support modules for storage devices are numbered 1 through 4, and each module includes all devices supported by any lower-numbered module. For example, Device Support Module 4 supports any device supported by Device Support Modules 1, 2, and 3. The enrollment certificate files for devices are:

devm1to2.lic	Upgrade from Device Support Module 1 to 2
devm2to3.lic	Upgrade from Device Support Module 2 to 3
devm3to4.lic	Upgrade from Device Support Module 3 to 4
devmod1.lic	Device Module 1
devmod2.lic	Device Module 2
devmod3.lic	Device Module 3
devmod4.lic	Device Module 4

You can upgrade from one Device Support Module to another or specify a module directly. For example, to upgrade from module 3 to module 4, enter:

```
register license file(devm2to3.lic)
```

To register Device Support Module 4, enter:

```
register license file(devmod4.lic)
```

Any attempt to use an HSM client or to define a library or drive that requires a device support module fails if the proper license is not registered. If you try to mount a volume requiring a library or drive that is not licensed, the operation also fails.

Registering the Disaster Recovery Manager (DRM) Feature

You license ADSM to support the Disaster Recovery Manager (DRM) feature. The enrollment certificate file for DRM is:

drm.lic	Add DRM support

For example, enter:

```
register license file(drm.lic)
```

Saving Your Licenses

Save the CD-ROM containing your enrollment certificate files if you need to register your licenses again for any of the following reasons:

- The server is corrupted.
- The server is moved to a different machine.

License Compliance

If license terms change (for example, a new license is specified for the server), the server conducts an audit to determine if the current server configuration conforms to the license terms.

The server also periodically audits compliance with the license terms. The results of this audit are used to check and enforce license terms. If 30 days have elapsed since the previous license audit, the administrator cannot cancel the audit.

The number of client nodes for which a server is licensed is enforced when the server is in open registration mode. If the terms of the license are violated by the addition of another registered node, the server issues a warning message stating that it is out of compliance.

If the server is not licensed to support a type of client (environment support) or device (device support module), server operations fail when you try to use the client or device. If one or more of the features or device support modules are licensed on the server, you receive error messages if you exceed your license terms.

Monitoring Licenses

An administrator can monitor license compliance by:

Auditing licenses
Use the AUDIT LICENSES command or the GUI to compare the current configuration with the current licenses.

Note: During a license audit, the server calculates, by node, the amount of backup, archive, and space management storage in use. This calculation can take a great deal of CPU time and can stall other server activity. Use the NOAUDITSTORAGE option to specify that storage is not to be calculated as part of a license audit.

Displaying license information
Use the QUERY LICENSE command or the GUI to display details of your current licenses and determine licensing compliance.

Scheduling automatic license audits
Use the SET LICENSEAUDITPERIOD command or the GUI to specify the number of days between automatic audits.

Ensuring Client/Server Authentication

Task	Required Privilege Class
Set password authentication Set password expiration	System

To ensure that only authorized administrators and client nodes are communicating with an authorized server, you can require the use of passwords. You can also require that users regularly change their passwords.

Setting Password Authentication

At installation, ADSM automatically sets password authentication on. With password authentication set to on, all users must enter a password when accessing the server.

To allow administrators and client nodes to access ADSM without entering a password, issue the following command:

```
set authentication off
```

Attention: Setting password authentication off reduces data security.

Setting User Password Expiration

At installation, ADSM sets a password expiration of 90 days. You can reset the expiration period from 1 to 9999 days. For example, to set the expiration period to 120 days, issue the following command:

```
set passexp 120
```

The expiration period begins when an administrator or client node is first registered to the server. If a user password is not changed within this period, the server prompts the user to change the password the next time the user tries to access the server.

Registering Administrators or Updating Information

Task	Required Privilege Class
Register an administrator or update information about other administrators	System
Update information about yourself	Any administrator

To register an administrator, specify a user ID and password. You also can provide contact information such as the user name and telephone number. Contact information is displayed when you query administrator information (FORMAT=DETAILED).

To register the administrator with a user ID of DAVEHIL and the password of *birds*, enter the REGISTER ADMIN command:

```
register admin davehil birds contact='backup team'
```

Note: At installation, the server console is defined with a special user ID, which is named SERVER_CONSOLE. This name is reserved and cannot be used by another administrator. At installation, the SERVER_CONSOLE user ID can be used to register other administrators and grant system privilege.

An administrator with system privilege can revoke or grant new privileges to the SERVER_CONSOLE user ID. However, you cannot update, lock, rename, or remove the SERVER_CONSOLE user ID from ADSM. The SERVER_CONSOLE user ID does not have a password. Therefore, you cannot use the user ID from an administrative client unless you set authentication off.

If as an administrator you forget your password, you can reset the password by issuing the UPDATE ADMINISTRATOR command. For example, to change his password to *ganymede*, DAVEHIL enters:

```
update admin davehil ganymede
```

Note: The SERVER_CONSOLE administrator's ID and contact information cannot be updated.

Granting Administrative Authority

Task	Required Privilege Class
Grant authority to other administrators	System

After administrators are registered, they can make queries and request command-line help. To perform other ADSM functions, they must be granted authority by being assigned one or more administrative privilege classes.

This section describes the privilege classes, which are illustrated in Figure 52. An administrator with system privilege can perform any ADSM function. Administrators with policy, storage, operator, or analyst privileges can perform subsets of ADSM functions.

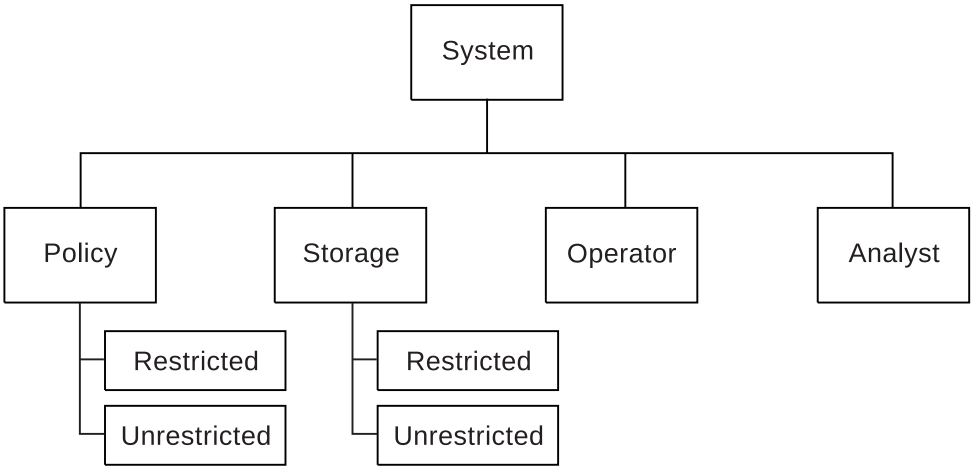


Figure 52. Administrative Privilege Classes

System Privilege

An administrator with *system privilege* can perform any ADSM administrative task.

The following tasks can be performed only by administrators with system privilege:

- Register or remove administrators
- Grant or revoke all levels of administrative authority
- Lock or unlock administrators from the server
- Rename administrators or update administrator information
- Define or delete policy domains and storage pools
- Import or export data from the server
- Cancel administrative background processes
- Set operating parameters for the server
- Perform license audits
- Grant privilege classes to other administrators

To grant the system privilege class to administrator KACZ, enter:

```
grant authority kacz classes=system
```

Unrestricted Policy Privilege

An administrator with *unrestricted policy privilege* can manage the backup and archive services for client nodes assigned to any policy domain. When new policy domains are defined to the server, an administrator with unrestricted policy privilege is automatically authorized to manage the new policy domains.

An administrator with unrestricted policy privilege can:

- Register client nodes in any policy domain
- Manage any client node access to the server
- Delete any client node files from storage pools
- Manage policy objects within any policy domain

  Note: System privilege is required to copy, define, or delete the policy domains themselves.

- Manage schedules, that automatically back up or archive files
- Associate client nodes to schedules defined in the same policy domain

To grant unrestricted policy privilege to administrator SMITH, enter:

```
grant authority smith classes=policy
```

Restricted Policy Privilege

An administrator with *restricted policy privilege* can perform the same operations as an administrator with unrestricted policy privilege **but only for specified policy domains**.

An administrator with restricted policy privilege can:

- Register a client node to an authorized policy domain
- Manage access for client nodes assigned to an authorized policy domain
- Delete files from storage pools for client nodes in authorized policy domains
- Manage policy objects in authorized policy domains
- Manage backup or archive schedules in authorized policy domains
- Associate schedules to client nodes assigned to an authorized policy domain

To grant restricted policy privilege over the policy domain named ENGPOLDOM, to administrator JONES enter:

```
grant authority jones domains=engpoldom
```

Unrestricted Storage Privilege

An administrator with *unrestricted storage privilege* has the authority to manage the database, recovery log, and all storage pools.

An administrator with unrestricted storage privilege can:

- Define volumes to the database or recovery log
- Extend or reduce the size of the database or recovery log
- Create mirrored copy sets of the database or recovery log
- Delete volumes from the database or recovery log
- Manage disk and tape device classes
- Define volumes to any disk or tape storage pools
- Move data from a storage pool to any other storage pool
- Delete volumes from any storage pool
- Audit volumes belonging to any storage pool

Note: However, an administrator with unrestricted storage privilege cannot define or delete storage pools.

To grant unrestricted storage privilege to administrator COYOTE, enter:

```
grant authority coyote classes=storage
```

Restricted Storage Privilege

Administrators with *restricted storage privilege* can manage only those storage pools to which they are authorized. They cannot manage the database or recovery log.

For those authorized storage pools, administrators with restricted storage privilege can:

- Define volumes to the storage pools
- Move data from one volume to another in a storage pool
- Delete volumes from the storage pools
- Audit volumes belonging to the storage pools

For example, assume that you have these tape storage pools: TAPEPOOL1, TAPEPOOL2, and TAPEPOOL3. To grant restricted storage privilege for these storage pools to administrator HOLLAND, you could enter:

```
grant authority holland stgpools=tape*
```

HOLLAND is restricted to managing storage pools beginning with “TAPE” that existed when the authority was granted. HOLLAND is not authorized to manage any storage pools that are defined after authority has been granted.

To add a new storage pool, TAPEPOOL4, to HOLLAND’s authority, enter:

```
grant authority holland stgpools=tapepool4
```

Operator Privilege

Administrators with *operator privilege* control the immediate operation of the ADSM server and the availability of storage media.

An administrator with operator privilege can:

- Disable the server to prevent clients from accessing the server
- Enable the server for access by clients
- Cancel client/server sessions
- Vary disk volumes on or off line to perform maintenance
- Reset the error status for tape volumes
- Manage tape mounts
- Halt the server, when necessary

To grant operator privilege to administrator BILL, enter:

```
grant authority bill classes=operator
```

Analyst Privilege

An administrator with *analyst privilege* can issue commands that reset the counters that track server statistics.

To grant analyst privilege to administrator MARYSMITH, enter:

```
grant authority marysmith classes=analyst
```

Changing Administrative Authority

Task	Required Privilege Class
Extend, revoke, or reduce administrative privilege classes	System

You can extend, revoke or reduce another administrator's authority.

Extending Administrative Privilege

Granting authority to an administrator adds to any existing privilege classes; it does not override those classes.

For example, JONES has restricted policy privilege for policy domain ENGPOLDOM. Enter the following command to extend JONES' authority to policy domain MKTPOLDOM and add operator privilege:

```
grant authority jones domains=mktpoldom classes=operator
```

Revoking One or More Administrative Privilege Classes

You can revoke part of an administrator's authority by specifying the administrator's user ID and one or more privilege classes.

Assume that rather than revoking all of the privilege classes for administrator JONES you wished only to revoke his operator authority and his policy authorization to policy domain MKTPOLDOM. You would enter:

```
revoke authority jones classes=operator domains=mktpoldom
```

JONES still has policy privilege to the ENGPOLDOM policy domain.

Revoking All Administrative Privilege Classes

To revoke all administrative privilege classes, do not specify any privilege classes, policy domains, or storage pools. For example, to revoke both the storage and operator privilege classes from administrator JONES enter:

```
revoke authority jones
```

Reducing Privilege Classes

You can reduce an administrator's authority simply by revoking one or more privilege classes and granting one or more other classes.

For example, administrator HOGAN has system authority. To reduce HOGAN to the operator privilege class do the following:

1. Revoke the system privilege class by entering:

```
revoke authority hogan classes=system
```

2. Grant operator privilege class by entering:

```
grant authority hogan classes=operator
```

Managing Administrator Access

An administrator can control access to the server by renaming or removing an administrator, or by locking and unlocking an administrator from the server.

Task	Required Privilege Class
Rename an administrator user ID	System
Remove other administrators from the server	
Temporarily prevent other administrators from accessing the system	
Display administrator information	Any administrator

Renaming an Administrator

You can rename an administrator ID when an employee wants to be identified by a new ID, or you want to assign an existing administrator ID to another person. You cannot rename an administrator ID to one that already exists on the system.

For example, if administrator HOLLAND leaves your organization, you can assign administrative privilege classes to another user by completing the following steps:

1. Assign HOLLAND’s user ID to WAYNESMITH by issuing the RENAME ADMIN command:

```
rename admin holland waynesmith
```

   By renaming the administrator ID, you remove HOLLAND as a registered administrator from the server. In addition, you register WAYNESMITH as an administrator with the password, contact information, and administrative privilege classes previously assigned to HOLLAND.

2. Change the password to prevent the previous administrator from accessing the server by entering:

```
update admin waynesmith new_password contact="development"
```

Note: The administrator SERVER_CONSOLE cannot be renamed.

Removing Administrators

You can remove administrators from the server so that they no longer have access to administrator functions. For example, to remove registered administrator ID SMITH, enter:

```
remove admin smith
```

Notes:

1. You cannot remove the last system administrator from the system.
2. You cannot remove the administrator SERVER_CONSOLE.

Locking and Unlocking Administrators from the Server

You can lock out administrators to temporarily prevent them from accessing ADSM.

For example, administrator MARYSMITH takes a leave of absence from your business. You can lock her out by entering:

```
lock admin marysmith
```

When she returns, any system administrator can unlock her administrator ID by entering:

```
unlock admin marysmith
```

MARYSMITH can now access ADSM to complete administrative tasks.

Note: You cannot lock or unlock the SERVER_CONSOLE ID from the server.

Requesting Information about Administrators

Any administrator can query the server to view administrator information. You can also query all administrators authorized with a specific privilege class.

For example, to query the system for a detailed report on administrator ID DAVEHIL, issue the QUERY ADMIN command:

```
query admin davehil format=detailed
```

Figure 53 displays a detailed report.

```
       Administrator Name: DAVEHIL
    Last Access Date/Time: 02/09/1996 19:49:46
   Days Since Last Access: 1
   Password Set Date/Time: 02/08/1996 19:49:31
  Days Since Password Set: 1
                  Locked?: No
                  Contact: backup team
         System Privilege:
         Policy Privilege: ENGPOLDOM
        Storage Privilege:
        Analyst Privilege:
       Operator Privilege:
        Registration Date: 02/09/1996 19:00:00
Registering Administrator: REES
```

Figure 53. A Detailed Administrator Report

Managing Client Nodes

Task	Required Privilege Class
Set registration to open or closed	System
Register client nodes to any policy domain	System or unrestricted policy
Register client nodes to specific policy domains	System, unrestricted policy, or restricted policy for those domains
Update, rename, lock, or unlock any client nodes	System, unrestricted policy
Update, rename, lock, or unlock client nodes assigned to specific policy domains	System, unrestricted policy, or restricted policy for those domains
Request information about client nodes or file spaces	Any administrator
Delete any file space from storage pools	System or unrestricted policy

Task	Required Privilege Class
Delete file spaces defined for client nodes assigned to specific policy domains	System, unrestricted policy, or restricted policy for those domains
Remove any client nodes	System or unrestricted policy
Remove client nodes assigned to specific policy domains	System, unrestricted policy, or restricted policy for those domains

Managing client node registration includes:

- Setting client node registration to open or closed
- Registering client nodes
- Updating client node information
- Managing client node access
- Requesting information about client nodes
- Requesting information about file spaces
- Deleting file spaces and client nodes

Setting Client Node Registration

Before a user can request backup or archive services, the workstation, or client node, must be registered with the server.

ADSM provides two methods for registering client nodes with an ADSM server:

Open registration
Users register their own client nodes.

Closed registration
An administrator must register each client node.

At installation, registration is set to closed. To set registration to open, enter:

```
set registration open
```

Note: Existing registered client nodes are not affected by changes to the registration process.

User Registration of Client Nodes

Under open registration, when a user accesses ADSM from an unregistered workstation, the server prompts the user for a password and contact information and registers the workstation as a client node with the server. On UNIX systems, only the root user can register a workstation as a client node with the server.

ADSM sets the following defaults:

- Assigns each client node to the policy domain STANDARD.
- Allows each client user to choose whether or not to compress files. On a UNIX system a root user can define whether compression is used by entering the COMPRESSION option in the **dsm.opt** client options file.

- Allows each client node user to delete archived copies (but not backed up files) from storage pools.

To change the defaults after the client node has been registered, you can update the client node (see “Updating Client Node Information” on page 295).

Administrator Registration of Client Nodes

To register a client node under open or closed registration, an administrator provides some or all of the following information:

- The node name. UNIX users should provide the value returned by the HOSTNAME command to the administrator.
- The node password.
- The policy domain to which the client node is assigned.
- Whether the user can compress files before they are backed up, archived, or space-managed directly to tape.

  Compression saves throughput time and server storage but requires more workstation memory and CPU cycles. Typically, a workstation with a slow processor connected to the server on a high-speed transmission line does not benefit from compression.

  Attention: Clients can use either client compression or drive compression but not both. For details, see “Using Data Compression” on page 104.

  To optimize performance or to ease memory constraints at the workstation, an ADSM administrator can restrict file compression. You can select one of three options:

  – Compress files

  – Do not compress files

  – Use the value in the COMPRESSION option in the client system options file or the API configuration file

- Whether the user is allowed to delete backed up or archived files from storage pools, by using the DSMC DELETE FILESPACE or DSMC DELETE ARCHIVE command.

  If users cannot delete archived or backed up files, an administrator must do so (see “Deleting File Spaces and Client Nodes” on page 299).

For example, you want to register three workstations from the engineering department and assign them to the *ENGPOLDOM* policy domain. (Before you can assign client nodes to a policy domain, the policy domain must exist. For how to define a policy domain, see Chapter 10, “Managing Policies” on page 185.)

You want to set file compression on and let the users delete backed up or archived files from storage pools. From an administrative client, you can use the macro facility to register more than one client node at a time. For this example, you create a macro file named REGENG.MAC, that contains the following REGISTER NODE commands:

```
register node ssteiner choir contact='department 21'
domain=engpoldom compression=yes archdelete=yes backdelete=yes

register node carolh skiing contact='department 21, second shift'
domain=engpoldom compression=yes archdelete=yes backdelete=yes

register node mab guitar contact='department 21, third shift'
domain=engpoldom compression=yes archdelete=yes backdelete=yes
```

Next, issue the MACRO command:

```
macro regeng.mac
```

For information on the MACRO command, see *ADSM Administrator's Reference*.

Managing Client Node Access

You can control client node access to ADSM by updating or renaming client nodes or by locking and unlocking client nodes from the server.

Updating Client Node Information

You can update the following client node information:

- The user password or contact information
- The policy domain to which the client node is assigned

  Note: An administrator with restricted policy privilege must be authorized to the current policy domain and to the new policy domain.

- Whether file compression is required
- Whether users can delete backed up or archived files from storage pools

For example, you can update client node TOMC to prevent him from deleting archived files from storage pools by entering:

```
update node tomc archdelete=no
```

Renaming Client Nodes

You can rename a client node if the workstation network name or host name changes.

For example, with UNIX systems, users define their ADSM node named based on the value returned by the HOSTNAME command. When users access the server, their ADSM user IDs match the host name of their workstations. If the host name changes, you can update a client node user ID to match the new host name.

For example, to rename CAROLH to ENGNODE enter:

```
rename node carolh engnode
```

ENGNODE retains the contact information and access to backup and archive data that belonged to CAROLH. All files backed up or archived by CAROLH now belong to ENGNODE.

Locking and Unlocking Client Nodes

You can prevent a client node from accessing the server and performing functions such as back up and restore or archive and retrieve. You can later let the client node reaccess the server. For example, to prevent client node MAB from accessing the server, enter:

```
lock node mab
```

To let client node MAB reaccess the server, enter:

```
unlock node mab
```

Requesting Information about Client Nodes

You can request information about client nodes. For example, as a policy administrator, you might query the server about all client nodes assigned to the policy domains for which you have authority. Or you might query the server for detailed information about one client node.

Client Nodes Assigned to Specific Policy Domains

You can display information about client nodes assigned to specific policy domains. For example, to view information about client nodes assigned to STANDARD and ENGPOLDOM policy domains, enter:

```
query node * domain=standard,engpoldom
```

The output from that command might look like this:

```
Node Name    Platform     Policy Domain   Days Since   Days Since   Locked?
                          Name                  Last     Password
                                              Access          Set
----------   --------     --------------  ----------   ----------   -------
DEBBYG         DOS        STANDARD                 2           12   No
ENGNODE        AIX        ENGPOLDOM               <1            1   No
HTANG          OS/2       STANDARD                 4           11   No
MAB            AIX        ENGPOLDOM               <1            1   No
PEASE          AIX        STANDARD                 3           12   No
SSTEINER       (?)        ENGPOLDOM               <1            1   No
```

A Specific Client Node

You can view information about specific client nodes. For example, to review the registration parameters defined for client node PEASE, enter:

```
query node pease format=detailed
```

The resulting report would look like this:

```
                     Node Name: PEASE
                      Platform: AIX
            Policy Domain Name: STANDARD
         Last Access Date/Time: 02/21/1996 10:58:36
        Days Since Last Access: 3
        Password Set Date/Time: 02/09/1996 10:02:00
       Days Since Password Set: 12
                       Locked?: No
                       Contact:
                   Compression: Yes
       Archive Delete Allowed?: No
        Backup Delete Allowed?: No
             Registration Date: 02/09/1996 10:02:00
     Registering Administrator: REES
Last Communication Method Used: Tcp/Ip
    Bytes Received Last Session: 1,719
        Bytes Sent Last Session: 602
Duration of Last Session (sec): 184.63
    Pct. Idle Wait Last Session: 99.69
   Pct. Comm. Wait Last Session: 0.00
   Pct. Media Wait Last Session: 0.00
```

Requesting Information about File Spaces

A *file space name* identifies a group of files that are stored as a logical unit in server storage. On registered client nodes, users can define file spaces for their workstation.

On client nodes such as OS/2 or DOS, a file space name identifies a logical partition, such as the volume label of a disk drive. For example, a volume with the label XYZ is a different file space from a volume with the label ABC.

On client nodes such as AIX or SunOS, a file space name identifies a file system or file space defined by a user with the VIRTUALMOUNTPOINT option. With this option, users can define a virtual mount point for a file system to back up or archive files beginning with a specific directory or subdirectory. For information on the VIRTUALMOUNTPOINT option, refer to the appropriate *ADSM Using the Backup-Archive Client*.

You can display file space information in order to:

- Identify file spaces defined to each client node, so that you can delete each file space from the server before removing the client node from the server
- Monitor the space used on workstation's disks
- Monitor whether backups are completing successfully for the file space
- Determine the date and time of the last backup

You display file space information by identifying the client node name and file space name.

Note: File space names are case-sensitive and must be entered exactly as known to the server.

For example, to view information about file spaces defined for client node PEASE, enter:

```
query filespace pease *
```

The following figure shows the output from this command. The report shows that client node ID PEASE:

- Has three file spaces on an AIX workstation
- Runs the *JFS* file system
- The amount of used and available space in each file space

```
Node Name                      Filespace   Platform Filespace Capacity %Util
                               Name                 Type          (MB)
------------------------------ ----------- -------- --------- -------- -----
PEASE                          /home/peas- AIX      JFS          196.0  91.7
                                e/dir
PEASE                          /home/peas- AIX      JFS          328.0  81.0
                                e/dir1
PEASE                          /home/peas- AIX      JFS           46.9  96.0
                                e/dir2
```

Deleting File Spaces and Client Nodes

You can delete a client node from a server, but first you must delete all of that client's data from server storage by deleting any file spaces belonging to the node.

Deleting a File Space

You may want to delete a file space when:

- Users are not authorized to delete backed up or archived files in storage pools

  The authority to delete backed up or archived files from server storage is set when a client node is registered. See "Setting Client Node Registration" on page 293 and "Administrator Registration of Client Nodes" on page 294 for information on allowing users to delete files in storage pools.

  For example, client node PEASE no longer needs archived files in file space */home/pease/dir2*. However, he does not have the authority to delete those files. You can delete them by entering:

```
delete filespace pease /home/pease/dir2 type=archive
```

- You want to remove a client node from the server

  You must delete a user's files from storage pools before you can remove a client node. For example, to delete all file spaces belonging to client node ID DEBBYG, enter:

```
delete filespace debbyg * type=any
```

- You want to delete files belonging to a specific owner

  For client nodes that support multiple users, such as UNIX, a file owner name is associated with each file on the server. The owner name is the user ID of the operating system, such as the UNIX user ID. When you delete a file space belonging to a specific owner, only files that have the specified owner name in the file space are deleted.

Removing Client Nodes

After all file spaces belonging to a client node have been deleted (see “Deleting a File Space” on page 299), you can delete the client node.

For example, to remove client node DEBBYG, enter:

```
remove node debbyg
```

Registering an Application Programming Interface to the Server

Workstation users can request ADSM services by using an application that uses the ADSM application programming interface (API). An administrator uses the REGISTER NODE command to register the workstation as an ADSM client.

Understanding How the Compression Option is Set

For applications that use the ADSM API, compression can be determined by:

- An administrator during registration who can:
  - Require that files are compressed
  - Restrict files from being compressed by the client
  - Allow the application or client user to determine the compression status
- The client options file. If an administrator does not set compression on or off, ADSM checks the compression status set in the client options file. The client options file is required, but the API user configuration file is optional.
- One of the object attributes. When an application sends an object to the server, some object attributes can be specified. One of the object attributes is a flag that indicates whether or not the data has already been compressed. If the application turns this flag on during either a backup or an archive operation, then ADSM does not compress the data a second time. This process overrides what the administrator sets during registration.

For more information on setting options for the API and on controlling compression, see *ADSM Using the Application Programming Interface*.

Understanding How the File Deletion Option is Set

For applications using the ADSM API, the file deletion option can be set by:

- An administrator during registration

  If an administrator does not allow the file deletion, then an ADSM administrator must delete any objects or file spaces associated with the workstation from server storage.

  If an administrator allows file deletion, then ADSM checks the client options file.

- An application using the ADSM API deletion program calls

  If the application uses the **dsmDeleteObj** or **dsmDeleteFS** program call, then objects or files are marked for deletion when the application is executed.

Chapter 15. Exporting and Importing Data

ADSM provides an export-import facility that allows you to copy all or part of a server to removable media (export) so that data can be transferred to another server (import).

Task	Required Privilege Class
Perform export and import operations	System
Display information about export and import operations	Any administrator

This chapter takes you through the export and import tasks. The sections listed in the following table begin at the indicated pages.

Section	Page
Concepts:	
Data that can be exported and imported	303
Tasks:	
Preparing to export or import data	304
Monitoring export and import processes	306
Exporting data to sequential media volumes	310
Importing data from sequential media volumes	314

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, "Interface Cross-Reference" on page 467 lists each command and shows if its function is also available on the administrative client GUI.

Data That Can Be Exported and Imported

Administrators can export or import the following types of ADSM data:

- Server control information, which includes:
  - Administrator definitions
  - Client node definitions
  - Policy and scheduling definitions
- File data from server storage, which includes file space definitions and authorization rules. You can request that file data be exported in any of the following groupings of files:
  - Active and inactive versions of backed up files, archive copies of files, and space-managed files

© Copyright IBM Corp. 1995, 1997

- Active versions of backed up files, archive copies of files, and space-managed files
- Active and inactive versions of backed up files
- Active versions of backed up files
- Archive copies of files
- Space-managed files

Attention: You can export information from ADSM Version 1 to ADSM Version 2, but not from ADSM Version 2 to ADSM Version 1.

Your decision on what information to export depends on why you are exporting that information:

- To copy information to a second server to balance the workload across servers, use the EXPORT NODE, EXPORT POLICY, and EXPORT ADMIN commands. For example, when many client nodes access the same server, users contend for communication paths, server resources, and tape mounts during a restore or retrieve operation.

  To relieve a server of some workload and improve its performance, you may want to take one or all of the following actions:

  - Move a group of client nodes to a second server
  - Move policy definitions associated with these client nodes
  - Move administrator definitions for administrators who manage these client nodes

  When you complete the import to the second server, you can delete file spaces, client nodes, policy objects, scheduling objects and administrators from the source server to reduce contention for server resources.

- To copy data for the purpose of installing a new server, use the EXPORT SERVER command to copy all data to sequential media volumes.

Preparing to Export or Import Data

Before you export or import data, complete the following tasks:

- Use the export or import command with the PREVIEW parameter to verify what data will be moved
- Prepare sequential media for exporting and importing data

Using Preview before Exporting or Importing Data

ADSM provides the PREVIEW option on the EXPORT and IMPORT commands. With PREVIEW=YES, the report shows how much data will be transferred without actually moving any data. With PREVIEW=NO, the export or import operation is performed.

Issue each EXPORT or IMPORT command with PREVIEW=YES to determine which objects and how much data will be moved. ADSM sends the following types of messages to the server console and to the activity log for each operation:

Export Reports the types of objects, number of objects, and number of bytes that would be copied to sequential media volumes. Use this information to determine how many sequential media volumes you need to prepare for an export operation.

Import Reports the number and types of objects found on the sequential media volumes that meet your import specifications, and reports information about any problems that it detects, such as corrupted data. Use this information to determine which data to move from sequential media volumes to the server and to determine if you have enough storage pool space allocated on the server for the import operation to succeed.

To determine how much space is required to export server definitions and all backup versions, archive copies, and space-managed files to sequential media volumes, enter:

```
export server filedata=all preview=yes
```

After you issue this command, the server starts a background process and issues a message similar to the following:

```
EXPORT SERVER started as Process 4
```

You can view the preview results on the server console and by querying the activity log.

You can request information about the background process, as described in “Requesting Information about an Export or Import Process” on page 307. If necessary, you can cancel an export or import process, as described in “Canceling Server Processes” on page 253.

Planning for Sequential Media Used to Export Data

To export data, you must specify a device class that supports sequential media and identify the volumes that will be used to store the exported data. Use this section to help you select the device classes and prepare sequential media volumes.

Selecting a Device Class

You can query the source and the target servers about device classes to select a device class on each server that supports the same device type. If you cannot find a device class on each server that supports a like device type, then define a new device class for a device type that is available to both servers. See Chapter 7, “Defining Device Classes” on page 91.

Note: If the mount limit for the device class selected is reached when you request an export (that is, if all the drives are busy), ADSM automatically cancels lower priority operations, such as reclamation, to make a mount point available for the export.

Estimating the Number of Removable Media Volumes to Label

To estimate the number of tapes or optical disks needed to store export data, divide the number of bytes to be moved by the estimated capacity of a volume.

For example, you have 8mm tapes with an estimated capacity of 2472MB. If the preview shows that you need to transfer 4GB of data, then label at least two tape volumes before you export the data.

Using Scratch Media

ADSM allows you to use scratch media to ensure that you have sufficient space on which to store all export data. If you use scratch media, be sure to record their label names and the order in which they were mounted.

Labeling Removable Media Volumes

During an import process, you must specify the order in which volumes will be mounted. This order must match the order in which tapes or optical disks have been mounted during the export process.

To ensure that tapes or optical disks are mounted in the correct order, label tapes or optical disks with information that identifies the order in which they are mounted during the import process. For example, label tapes as DSM001, DSM002, DSM003, and so on to indicate the order in which data is stored on the tape volumes.

When you export data, record the date and time for each labeled volume. Store this information in a safe location, because you will need the information when you import the data to the server.

Monitoring Export and Import Processes

ADSM provides you with a number of methods for monitoring export or import processes.

- You can view information about a process that is running on the server console or from an administrative client running in console mode.
- You can query the activity log for status information when a process has completed, from the server console or from an administrative client running in batch or interactive mode.

Requesting Information about an Export or Import Process

After you issue an EXPORT or IMPORT command, the server starts a background process, assigns a process ID to the operation, and displays the process ID when the operation starts.

You can query an export or import process by specifying the process ID number. For example, to request information about the EXPORT SERVER operation, which started as process 4, enter:

```
query process 4
```

If you issue a preview version of an EXPORT or IMPORT command and then query the process, ADSM reports the types of objects to be copied, the number of objects to be copied, and the number of bytes to be copied.

When you export or import data and then query the process, ADSM displays the number and types of objects copied so far, and the total number of bytes that have been transferred, along with information on any media mount requests that may be outstanding for the process.

For guidance information on querying background processes, see “Requesting Information about Server Processes” on page 253.

Viewing Information from the Server Console

When you issue an IMPORT or EXPORT command, either from the server console or from an administrative client, information is displayed on the server console. Figure 54 on page 308 shows an example of the information that is displayed after issuing an EXPORT SERVER command.

```
ANR0610I EXPORT SERVER started by SERVER_CONSOLE as process 1.
ANR0639I EXPORT SERVER: Processing domain ENGPOLDOM.
ANR0640I EXPORT SERVER: Processing policy set ACTIVE in policy domain
ENGPOLDOM.
ANR0640I EXPORT SERVER: Processing policy set STANDARD in policy domain
ENGPOLDOM.
ANR0641I EXPORT SERVER: Processing management class STANDARD in domain
ENGPOLDOM, set ACTIVE.
ANR0641I EXPORT SERVER: Processing management class STANDARD in domain
ENGPOLDOM, set STANDARD.
ANR0643I EXPORT SERVER: Processing archive copy group in domain
ENGPOLDOM, set STANDARD, management class ACTIVE.
ANR0643I EXPORT SERVER: Processing archive copy group in domain
ENGPOLDOM, set STANDARD, management class STANDARD.
ANR0643I EXPORT SERVER: Processing backup copy group in domain
ENGPOLDOM, set STANDARD, management class ACTIVE.
ANR0643I EXPORT SERVER: Processing backup copy group in domain
ENGPOLDOM, set STANDARD, management class STANDARD.
ANR0604I EXPORT SERVER: No schedules were found in policy domain * for
exporting.
ANR0635I EXPORT SERVER: Processing node TOMC.
ANR0605I EXPORT SERVER: No schedule associations were found in
policy domain * for exporting.
ANR0637I EXPORT SERVER: Processing file space DRIVED for node TOMC.
ANR0637I EXPORT SERVER: Processing file space OS2 for node TOMC.
ANR0637I EXPORT SERVER: Processing file space OS2VDISK for node TOMC.
ANR0617I EXPORT SERVER: Processing completed successfully.
ANR0620I EXPORT SERVER: Copied 1 domain(s).
ANR0621I EXPORT SERVER: Copied 2 policy set(s).
ANR0622I EXPORT SERVER: Copied 2 management class(es).
ANR0623I EXPORT SERVER: Copied 4 copy group(s).
ANR0626I EXPORT SERVER: Copied 1 node definition(s).
ANR0627I EXPORT SERVER: Copied 3 file space(s), 16 archive file(s)
and 0 backup file(s).
ANR0629I EXPORT SERVER: Copied 3045632 bytes of data.
ANR0611I EXPORT SERVER started by SERVER_CONSOLE as process 1 has ended.
```

Figure 54. Sample Export Server Output Sent to the Server Console

Viewing Information from an Administrative Client

Use the console mode from an administrative client to monitor export or import operations or to capture processing messages to an output file. For example, to start an administrative session in console mode on an OS/2 client, enter:

```
> dsmadmc -consolemode
```

While the system is running in console mode, you cannot enter any administrative commands from the client session. You can, however, start another administrative client session for entering commands (for example, QUERY PROCESS) if you are using a multitasking workstation, such as OS/2 or AIX.

If you want ADSM to write all terminal output to a file, specify the OUTFILE option with a destination. For example, to write output to the SAVE.OUT file, enter:

```
> dsmadmc -consolemode -outfile=save.out
```

For information about using the CONSOLE mode option and ending an administrative session in console mode, see *ADSM Administrator's Reference*.

Querying the Activity Log for Export or Import Information

After an export or import process has completed, you can query the activity log for status information and possible error messages.

To minimize processing time when querying the activity log for export or import information, restrict the search by specifying *export* or *import* in the SEARCH parameter of the QUERY ACTLOG command.

For example, to determine how much data will be moved after issuing the preview version of the EXPORT SERVER command, query the activity log by entering:

```
query actlog search=export
```

Figure 55 on page 310 displays a sample activity log report.

```
Date/Time            Message
-------------------- --------------------------------------------------
05/03/1996 10:50:28  ANR0610I EXPORT SERVER started by ADSMADMIN as
process 1.
05/03/1996 10:50:28  ANR0639I EXPORT SERVER: Processing domain ENGPOLDOM.
05/03/1996 10:50:28  ANR0640I EXPORT SERVER: Processing policy set
ACTIVE in policy domain ENGPOLDOM.
05/03/1996 10:50:28  ANR0640I EXPORT SERVER: Processing policy set
STANDARD in policy domain ENGPOLDOM.
05/03/1996 10:50:29  ANR0641I EXPORT SERVER: Processing management class
STANDARD in domain ENGPOLDOM, set ACTIVE.
05/03/1996 10:50:29  ANR0641I EXPORT SERVER: Processing management class
STANDARD in domain ENGPOLDOM, set STANDARD.
05/03/1996 10:50:29  ANR0643I EXPORT SERVER: Processing archive copy
group in domain ENGPOLDOM, set STANDARD, management class ACTIVE.
05/03/1996 10:50:29  ANR0643I EXPORT SERVER: Processing archive copy
group in domain ENGPOLDOM, set STANDARD, management class STANDARD.
05/03/1996 10:50:29  ANR0642I EXPORT SERVER: Processing backup copy
group in domain ENGPOLDOM, set STANDARD,  management class ACTIVE.
05/03/1996 10:50:29  ANR0642I EXPORT SERVER: Processing backup copy
group in domain ENGPOLDOM, set STANDARD, management class STANDARD.
05/03/1996 10:50:29  ANR0604I EXPORT SERVER: No schedules were found in policy
domain * for exporting.
05/03/1996 10:50:29  ANR0635I EXPORT SERVER: Processing node TOMC.
05/03/1996 10:50:29  ANR0605I EXPORT SERVER: No schedule associations were
found in policy domain * for exporting.
05/03/1996 10:50:29  ANR0637I EXPORT SERVER: Processing file space DRIVED for
node TOMC.
05/03/1996 10:50:29  ANR0637I EXPORT SERVER: Processing file space OS2 for node
TOMC.
05/03/1996 10:50:29  ANR0637I EXPORT SERVER: Processing file space OS2VDISK for
node TOMC.
05/03/1996 10:50:32  ANR0617I EXPORT SERVER: Processing completed successfully.
05/03/1996 10:50:32  ANR0620I EXPORT SERVER: Copied 1 domain(s).
05/03/1996 10:50:32  ANR0621I EXPORT SERVER: Copied 2 policy set(s).
05/03/1996 10:50:32  ANR0622I EXPORT SERVER: Copied 2 management class(es).
05/03/1996 10:50:32  ANR0623I EXPORT SERVER: Copied 4 copy group(s).
05/03/1996 10:50:32  ANR0626I EXPORT SERVER: Copied 1 node definition(s).
05/03/1996 10:50:32  ANR0627I EXPORT SERVER: Copied 3 file space(s),
16 export file(s) and 0 backup file(s).
05/03/1996 10:50:32  ANR0629I EXPORT SERVER: Copied 3045632 bytes of data.
05/03/1996 10:50:32  ANR0611I EXPORT SERVER started by ADSMADMIN as
process 1 has ended.
```

Figure 55. Sample Activity Log Report on Exported Data

Exporting Data to Sequential Media Volumes

You can export all server control information or a subset of server control information by specifying one or more of the following export commands:

- EXPORT SERVER
- EXPORT ADMIN
- EXPORT NODE
- EXPORT POLICY

When you export data, you must specify the device class to which export data can be written. You must also list the volumes in the order in which they are mounted when the data is imported. See “Labeling Removable Media Volumes” on page 306 for information on labelling tape volumes.

Deciding When to Export Data

When you issue an EXPORT command, the operation runs as a background process. This process allows you to continue performing administrative tasks. In addition, users can continue to back up, archive, migrate, restore, retrieve, or recall files from ADSM.

If you choose to perform an export operation during normal working hours, be aware that administrators can change server definitions and users may modify files that are in server storage. If administrators or users modify data shortly after it has been exported, then the information copied to tape may not be consistent with data stored on the source server.

If you want to export an exact point-in-time copy of server control information, you can prevent administrative and other client nodes from accessing the server. See “Preventing Administrative Clients from Accessing the Server” and “Preventing Client Nodes from Accessing the Server.”

Preventing Administrative Clients from Accessing the Server

Administrators can change administrator, policy, or client node definitions during an export process. To prevent administrators from modifying these definitions, you can lock out administrator access to the server and cancel any administrative sessions before issuing an EXPORT command. After the export process is complete, unlock administrator access.

For more information on canceling sessions, see “Canceling a Client Session” on page 250. For more information on locking or unlocking administrators from the server, see “Locking and Unlocking Administrators from the Server” on page 291.

Preventing Client Nodes from Accessing the Server

If client node information is exported while the same client is backing up, archiving, or migrating files, the latest file copies for the client may not be exported to tape. To prevent users from accessing the server during export operations, cancel existing client sessions as described in “Canceling a Client Session” on page 250. Then you can do one of the following:

- Disable server access to prevent client nodes from accessing the server, as described in “Disabling or Enabling Server Access” on page 251.

  This option is useful when you export all client node information from the source server and want to prevent all client nodes from accessing the server.

- Lock out particular client nodes from server access, as described in “Locking and Unlocking Client Nodes” on page 296.

This option is useful when you export a subset of client node information from the source server and want to prevent particular client nodes from accessing the server until the export operation is complete.

After the export operation is complete, allow client nodes to access the server again by:

- Enabling the server, as described in “Disabling or Enabling Server Access” on page 251.
- Unlocking client nodes, as described in “Locking and Unlocking Client Nodes” on page 296

Exporting Server Data

When you issue the EXPORT SERVER command, the server exports all server control information. You can also export file data information with the EXPORT SERVER command.

For example, you want to export definitions and all file data to four defined tape cartridges, which are supported by the TAPE8MM device class. You want ADSM to use scratch volumes if the four volumes are not enough, and so you use the default of SCRATCH=YES. To issue this command, enter:

```
export server devclass=tape8mm
volumenames=dsm001,dsm002,dsm003,dsm004 filedata=all
```

During the export process, ADSM exports definition information before it exports file data information. This ensures that definition information is stored on the first tape volumes. This process allows you to mount a minimum number of tapes during the import process, if your goal is to copy only control information to the target server.

In the example above, the server exports:

- Administrator definitions
- Client node definitions
- Policy domain, policy set, management class, and copy group definitions
- Schedule definitions and client node associations
- File space definitions
- File space authorization rules
- Backed up, archived, and space-managed files

Exporting Administrator Information

When you issue the EXPORT ADMIN command, the server exports administrator definitions. Each administrator definition includes:

- Administrator name, password and contact information
- Any administrative privilege classes the administrator has been granted
- Whether the administrator ID is locked from server access

You can specify a list of administrator names, or you can export all administrator names from the server to tape volumes.

In the following example, definitions for the DAVEHIL and PENNER administrator IDs will be exported to the DSM001 tape volume, which is supported by the TAPE8MM device class. Do not allow any scratch media to be used during this export process. To issue this command, enter:

```
export admin davehil,penner devclass=tape8mm
volumenames=dsm001 scratch=no
```

Exporting Client Node Information

When you issue the EXPORT NODE command, the server exports client node definitions. Each client node definition includes:

- User ID, password, and contact information
- Name of the policy domain to which the client is assigned
- File compression status
- Whether the user has the authority to delete backed up or archived files from server storage
- Whether the client node ID is locked from server access

Optionally, you can specify whether to export file data. File data includes file space definitions and authorization rules. You can request that file data be exported in any of the following groupings of files:

- Active and inactive versions of backed up files, archive copies of files, and space-managed files
- Active versions of backed up files, archive copies of files, and space-managed files
- Active and inactive versions of backed up files
- Active versions of backed up files
- Archive copies of files
- Space-managed files

When client file data is exported, ADSM copies files to export volumes in the order of their physical location in server storage. This process minimizes the number of mounts required during the export process.

If you do not explicitly specify that you want to export file data, then ADSM only exports client node definitions.

For example, suppose you want to do the following:

- Export definitions for client nodes and file spaces in the ENGPOLDOM policy domain
- Export any active backup versions of files belonging to these client nodes
- Export this information to scratch volumes in the TAPE8MM device class

To issue this command, enter:

```
export node filespace=* domains=engpoldom
filedata=backupactive devclass=tape8mm
```

In this example, the server exports:

- Definitions of client nodes assigned to the engineering policy domain
- File space definitions and backup authorizations for each client node in the engineering policy domain
- Active versions of backed up files belonging to the client nodes assigned to the engineering policy domain

Exporting Policy Information

When you issue the EXPORT POLICY command, the server exports the following information belonging to each specified policy domain:

- Policy domain definitions
- Policy set definitions, including the active policy set
- Management class definitions, including the default management class
- Backup copy group and archive copy group definitions
- Schedule definitions
- Associations between client nodes and schedules

For example, suppose you want to export policy and scheduling definitions from the policy domain named ENGPOLDOM. You want to use tape volumes DSM001 and DSM002, which belong to the TAPE8MM device class, but allow the server to use scratch tape volumes if necessary. To issue this command, enter:

```
export policy engpoldom
devclass=tape8mm volumenames=dsm001,dsm002
```

Importing Data from Sequential Media Volumes

Before you import data to a new target server, you must:

1. Install ADSM on the target server. This step includes defining disk space for the database and recovery log.

   For information on installing ADSM, see *ADSM Quick Start*.

2. Define server storage for the target server.

   Because each server operating system handles devices differently, ADSM does not export server storage definitions. Therefore, you must define initial server storage for the target server. ADSM must at least be able to use a drive that is compatible with the export media. This task can include defining libraries, drives, device classes, storage pools, and volumes. See the *ADSM Administrator's Guide* that applies to the target server.

After ADSM is installed and set up on the target server, a system administrator can import all server control information or a subset of server control information by specifying one or more of the following import commands:

- IMPORT SERVER
- IMPORT ADMIN
- IMPORT NODE
- IMPORT POLICY

This section guides you through the entire process of importing all server control information and file data from tape volumes to a new target server. This process includes:

- Previewing information before you import data
- Importing definitions
- Tailoring server storage definitions on the target server
- Importing file data

After you understand how to import server control information and file data information, you can import any subset of data to the target server.

Step 1: Previewing Information before You Import Data

Before you import any data to the target server, preview each import command to determine what data you want to import to the target server. You can import all or a subset of export data from tapes.

When you set PREVIEW=YES, tape operators must mount export tape volumes so that the target server can calculate the statistics reported by the use of this parameter.

For example, to preview information for the IMPORT SERVER command, enter:

```
import server devclass=tape8mm preview=yes
volumenames=dsm001,dsm002,dsm003,dsm004
```

Figure 56 on page 316 shows an example of the messages sent to the server console and the activity log.

```
ANR0402I Session 3 started for administrator SERVER_CONSOLE (Server).
ANR1363I Import volume DSM001 opened (sequence number 1).
ANR0610I IMPORT SERVER started by SERVER_CONSOLE as process 2.
ANR0612I IMPORT SERVER: Reading EXPORT SERVER data from server ADSM exported
05/07/1996 12:39:48.
ANR0639I IMPORT SERVER: Processing domain ENGPOLDOM.
ANR0640I IMPORT SERVER: Processing policy set ACTIVE in policy domain
ENGPOLDOM.
ANR0640I IMPORT SERVER: Processing policy set STANDARD in policy domain
ENGPOLDOM.
ANR0641I IMPORT SERVER: Processing management class STANDARD in domain
ENGPOLDOM, set ACTIVE.
ANR0641I IMPORT SERVER: Processing management class MCENG in domain
ENGPOLDOM, set STANDARD.
ANR0641I IMPORT SERVER: Processing management class STANDARD in domain
ENGPOLDOM, set STANDARD.
ANR0643I IMPORT SERVER: Processing archive copy group in domain
ENGPOLDOM, set ACTIVE, management class STANDARD.
ANR0643I IMPORT SERVER: Processing archive copy group in domain ENGPOLDOM,
set STANDARD, management class MCENG.
ANR0643I IMPORT SERVER: Processing archive copy group in domain ENGPOLDOM,
set STANDARD, management class STANDARD.
ANR0642I IMPORT SERVER: Processing backup copy group in domain ENGPOLDOM,
set ACTIVE, management class STANDARD.
ANR0642I IMPORT SERVER: Processing backup copy group in domain ENGPOLDOM,
set STANDARD, management class MCENG.
ANR0642I IMPORT SERVER: Processing backup copy group in domain ENGPOLDOM,
set STANDARD, management class STANDARD.
ANR0638I IMPORT SERVER: Processing administrator DAVEHIL.
ANR0638I IMPORT SERVER: Processing administrator PENNER.
ANR0635I IMPORT SERVER: Processing node TOMC.
ANR0636I IMPORT SERVER: Processing file space OS2 for node TOMC as file
space OS1.
ANR0636I IMPORT SERVER: Processing file space DRIVED for node TOMC as file
space DRIVE1.
ANR0636I IMPORT SERVER: Processing file space OS2VDISK for node TOMC as file
space OS2VDIS1.
ANR1365I Import volume DSM001 closed (end reached).
ANR1363I Import volume DSM002 opened (sequence number 2).
ANR1365I Import volume DSM002 closed (end reached).
ANR1363I Import volume DSM003 opened (sequence number 3).
ANR1365I Import volume DSM003 closed (end reached).
ANR1363I Import volume DSM004 opened (sequence number 4).
ANR1365I Import volume DSM004 closed (end reached).
ANR0617I IMPORT SERVER: Processing completed successfully.
ANR0620I IMPORT SERVER: Copied 1 domain(s).
ANR0621I IMPORT SERVER: Copied 2 policy set(s).
ANR0622I IMPORT SERVER: Copied 2 management class(es).
ANR0623I IMPORT SERVER: Copied 6 copy group(s).
ANR0625I IMPORT SERVER: Copied 2 administrator(s).
ANR0626I IMPORT SERVER: Copied 1 node definition(s).
ANR0627I IMPORT SERVER: Copied 3 file space(s), 0 archive file(s) and 462
backup file(s).
ANR0629I IMPORT SERVER: Copied 8856358 bytes of data.
ANR0611I IMPORT SERVER started by SERVER_CONSOLE as process 2 has ended.
```

Figure 56. Sample Report Created by Issuing Preview for an Import Server Command

Use the value reported for the total number of bytes copied to estimate if you have sufficient storage pool space on the server to store imported file data.

For example, Figure 56 shows that 8 856 358 bytes of data will be imported. Ensure that you have at least 8 856 358 bytes of available space in the backup storage pools defined to the server. You can use the QUERY STGPOOL and QUERY VOLUME commands to determine how much space is available in the server storage hierarchy.

In addition, the preview report shows that 0 archive files and 462 backup files will be imported. Because backup data is being imported, ensure that you have sufficient space in the backup storage pools used to store this backup data. See “Step 3: Tailoring Server Storage Definitions on the Target Server” on page 319 for information on identifying storage pools on the target server.

For information on specifying the PREVIEW parameter, see “Using Preview before Exporting or Importing Data” on page 304. For information on reviewing the results of a preview operation, see “Monitoring Export and Import Processes” on page 306.

Step 2: Importing Definitions

Next, you want to import server control information, which includes:

- Administrator definitions
- Client node definitions
- Policy domain, policy set, management class, and copy group definitions
- Schedule definitions and client node associations

However, do not import file data at this time, because some storage pools named in the copy group definitions may not exist yet on the target server.

Before you import server control information, do the following:

- Read and understand the following information:
  - “Determining Whether to Replace Existing Definitions”
  - “Understanding How ADSM Imports Active Policy Sets” on page 318
- Start an administrative client session in console mode to capture import messages to an output file. See “Directing Import Messages to an Output File” on page 318.

Then import the server control information from specified tape volumes. See “Importing Server Control Information” on page 319.

Determining Whether to Replace Existing Definitions

By using the REPLACEDEFS option, you can specify whether to replace existing definitions on the target server when ADSM encounters an object with the same name during the import process.

For example, if a definition exists for the ENGPOLDOM policy domain on the target server before you import policy definitions, then you must specify REPLACEDEFS=YES to have ADSM replace the existing definition with the data from the export tape.

Definitions that can be replaced include administrator, client node, policy, or schedule definitions. The default is to not replace existing definitions on the target server.

Understanding How ADSM Imports Active Policy Sets

When ADSM imports policy definitions, the following objects are imported to the target server:

- Policy domain definitions
- Policy set definitions, including the ACTIVE policy set
- Management class definitions
- Backup copy group definitions
- Archive copy group definitions
- Schedule definitions defined for each policy domain
- Client node associations, if the client node definition exists on the target server

If ADSM encounters a policy set named ACTIVE on the tape volume during the import process, it uses a temporary policy set named $$ACTIVE$$ to import the active policy set.

After $$ACTIVE$$ is imported to the target server, ADSM activates this policy set. During the activation process, the server validates the policy set by examining the management class and copy group definitions.

ADSM reports on the following conditions, which result in warning messages during validation:

- The storage destinations specified in the backup copy groups and the archive copy groups do not refer to defined storage pools.
- The default management class does not contain a backup or archive copy group.
- The current ACTIVE policy set contains management class names that are not defined in the policy set to be activated.
- The current ACTIVE policy set contains copy group names that are not defined in the policy set to be activated.

After each $$ACTIVE$$ policy set has been activated, ADSM deletes that $$ACTIVE$$ policy set from the target server. To view information about active policy on the target server, you can use the following commands:

- QUERY COPYGROUP
- QUERY MGMTCLASS
- QUERY POLICYSET

Results from issuing the QUERY DOMAIN command show the activated policy set as $$ACTIVE$$. ADSM uses the $$ACTIVE$$ name to show you that the policy set which is currently activated for this domain is the policy set that was active at the time the export was performed.

Directing Import Messages to an Output File

The information generated by the validation process can help you define a storage hierarchy that supports the storage destinations currently defined in the import data.

You can direct import messages to an output file to capture any error messages that are detected during the import process by starting an administrative client session in console mode before you invoke this import command.

For example, to direct messages to an output file named IMPSERV.OUT, enter:

```
> dsmadmc -consolemode -outfile=impserv.out
```

Importing Server Control Information

Now you are ready to import the server control information. Based on the information generated during the preview operation, you know that all definition information has been stored on the first tape volume named DSM001. Specify that this tape volume can be read by a device belonging to the TAPE8MM device class.

From an administrative client session or from the server console, enter:

```
import server filedata=none devclass=tape8mm
volumenames=dsm001
```

Step 3: Tailoring Server Storage Definitions on the Target Server

After you import definition information, use the reports generated by the import process to help you tailor storage for the target server.

To tailor server storage definitions on the target server, complete the following steps:

1. Identify any storage destinations specified in copy groups and management classes that do not match defined storage pools:

   - If the policy definitions you imported included an ACTIVE policy set, that policy set is validated and activated on the target server. Error messages generated during validation include whether any management classes or copy groups refer to storage pools that do not exist on the target server. You have a copy of these messages in a file if you directed console messages to an output file as described in “Directing Import Messages to an Output File” on page 318.
   - Query management class and copy group definitions to compare the storage destinations specified with the names of existing storage pools on the target server.

     To request detailed reports for all management classes, backup copy groups, and archive copy groups in the ACTIVE policy set, enter these commands:

```
query mgmtclass * active * format=detailed

query copygroup * active * standard type=backup format=detailed

query copygroup * active * standard type=archive format=detailed
```

2. If storage destinations for management classes and copy groups in the ACTIVE policy set refer to storage pools that are not defined, do one of the following:

   - Define storage pools that match the storage destination names for the management classes and copy groups, as described in “Defining or Updating Storage Pools” on page 137.

   - Change the storage destinations for the management classes and copy groups. Do the following:

     a. Copy the ACTIVE policy set to another policy set
     b. Modify the storage destinations of management classes and copy groups in that policy set, as required
     c. Activate the new policy set

     For information on copying policy sets, see “Defining and Updating a Policy Set” on page 203.

Depending on the amount of client file data that you expect to import, you may want to examine the storage hierarchy to ensure that sufficient storage space is available. Storage pools specified as storage destinations by management classes and copy groups may fill up with data. For example, you may need to define additional storage pools to which data can migrate from the initial storage destinations.

Step 4: Importing File Data Information

After you have defined the appropriate storage hierarchy on the target server, you can import file data from the tape volumes. File data includes file space definitions and authorization rules. You can request that file data be imported in any of the following groupings of files:

- Active and inactive versions of backed up files, archive copies of files, and space-managed files
- Active versions of backed up files, archive copies of files, and space-managed files
- Active and inactive versions of backed up files
- Active versions of backed up files
- Archive copies of files
- Space-managed files

Before you import file data information:

- Understand how ADSM handles duplicate file space names.

- Decide whether to keep the original creation date for backup versions and archive copies or to import file data using an adjusted date.

Then you can import file data to the target server.

Understanding How Duplicate File Spaces Are Handled

When ADSM imports file data information, it imports any file spaces belonging to each specified client node. If a file space definition already exists on the target server for the node, ADSM does *not* replace the existing file space name.

If ADSM encounters duplicate file space names when it imports file data information, it creates a new file space name for the imported definition by replacing the final character or characters with a number. A message showing the old and new file space names is written to the server console and to the activity log.

For example, if the C_DRIVE and D_DRIVE file space names reside on the target server for node FRED and on the tape volume for FRED, then the server imports the C_DRIVE file space as C_DRIV1 file space and the D_DRIVE file space as D_DRIV1 file space, both assigned to node FRED.

Deciding Whether to Use a Relative Date When Importing File Data

When you import file data, you can keep the original creation date for backup versions and archive copies, or you can specify that ADSM use an adjusted date.

Because tape volumes containing exported data might not be used for some time after the export operation that created them, the original dates defined for backup versions and archive copies may be old enough that files are expired immediately by policy when the data is imported to the target server.

To prevent backup versions and archive copies from being expired immediately, specify DATES=RELATIVE on the IMPORT NODE or IMPORT SERVER commands to adjust for the elapsed time since the files were exported to tape.

For example, assume that data exported to tape includes an archive copy archived five days prior to the export operation. If the tape volume resides on the shelf for six months before the data is imported to the target server, ADSM resets the archival date to five days prior to the import operation.

If you want to keep the original backup and archive dates set for backup versions and archive copies, then use DATES=ABSOLUTE, which is the default. If you use the absolute value, then any files whose retention period has passed will be expired shortly after they are imported to the target server.

Issuing an Import Server or Import Node Command

You can import file data, either by issuing the IMPORT SERVER or IMPORT NODE command. When you issue either of these commands, you can specify which type of files should be imported for all client nodes specified and found on the export tapes. You can specify any of the following values to import file data:

All

Specifies that all active and inactive versions of backed up files, archive copies of files, and space-managed files for specified client nodes are imported to the target server

None

Specifies that no files are imported to the target server; only client node definitions are imported

Archive

Specifies that only archive copies of files are imported to the target server

Backup

Specifies that only backup copies of files, whether active or inactive, are imported to the target server

Backupactive

Specifies that only active versions of backed up files are imported to the target server

Allactive

Specifies that only active versions of backed up files, archive copies of files, and space-managed files are imported to the target server

Spacemanaged

Specifies that only files that have been migrated from a user's local file system (space-managed files) are imported

For example, suppose you want to import all backup versions of files, archive copies of files, and space-managed files to the target server. You do not want to replace any existing server control information during this import operation. Specify the four tape volumes that were identified during the preview operation. These tape volumes can be read by any device in the TAPE8MM device class. To issue this command, enter:

```
import server filedata=all replacedefs=no
devclass=tape8mm volumenames=dsm001,dsm002,dsm003,dsm004
```

Considerations When Importing Data

You can use an import command to copy a subset of the information on export tapes to the target server. For example, if a tape was created with EXPORT SERVER, you can import only node information from the tape by using IMPORT NODE.

While ADSM allows you to issue any import command, data cannot be imported to the server if it has not been exported to tape. For example, if a tape is created with the EXPORT POLICY command, an IMPORT NODE command will not find any data on the tape because node information is not a subset of policy information.

Table 13 on page 323 shows the commands you can use to import a subset of exported information to a target server.

Table 13. Importing a Subset of Information from Tapes

If tapes were created with this export command:	You can issue this import command:	You cannot issue this import command:
EXPORT SERVER	IMPORT SERVER IMPORT ADMIN IMPORT NODE IMPORT POLICY	—
EXPORT NODE	IMPORT NODE IMPORT SERVER	IMPORT ADMIN IMPORT POLICY
EXPORT ADMIN	IMPORT ADMIN IMPORT SERVER	IMPORT NODE IMPORT POLICY
EXPORT POLICY	IMPORT POLICY IMPORT SERVER	IMPORT ADMIN IMPORT NODE

Recovering from Errors during the Import Process

During import processing, the server may encounter invalid data due to corruption during storage on tape or in the database prior to the export operation. If invalid data is encountered during an import operation, the server does the following:

- If a new object is being defined, the default value is used
- If the object already exists, the existing parameter is not changed

The server reports on the affected objects to the server console and the activity log during import and export operations. You should query these objects when the import process is complete to see if they reflect information that is acceptable to you.

Each time you run the IMPORT NODE or IMPORT SERVER command with the FILEDATA parameter equal to a value other than NONE, ADSM creates a new file space and imports data to it. This process ensures that the current import does not overwrite data from a previous import. For information on how ADSM handles duplicate file spaces, see “Understanding How Duplicate File Spaces Are Handled” on page 321.

A file space definition may already exist on the target server for the node. If so, an administrator with system privilege can issue the DELETE FILESPACE command to remove file spaces that are corrupted or no longer needed. For more information on the DELETE FILESPACE command, refer to the *ADSM Administrator’s Reference*.

Renaming a File Space

An imported file space can have the same name as a file space that already exists on a client node. In this case, the server does not overlay the existing file space, and the imported file space is given a new system generated file space name. This new name may match file space names that have not been backed up and are unknown to the server. In this case, you can use the RENAME FILESPACE command to rename the imported file space to the naming convention used for the client node.

Part 6. Protecting the Server

© Copyright IBM Corp. 1995, 1997

Chapter 16. Protecting and Recovering Your Data

If your ADSM database or recovery log are unusable, the entire ADSM server is unavailable. Failure, damage, or loss of the database, recovery log, or storage pools can cause the unrecoverable loss of client data. If a storage pool volume is lost and cannot be recovered, any client data on the volume is also lost. This chapter describes how ADSM can guard against these situations and helps you to choose the method that is best for your installation. The term *tape* is used often in the following descriptions. It refers to any kind of sequential access, removable media.

The sections listed in the following table begin on the indicated pages.

Section	Page
Concepts:	
Levels of data protection provided by ADSM	328
Protecting Data:	
How to back up storage pools	332
How to mirror the database and recovery log	332
How to back up the database	335
Recovering Data:	
How to recover the database and recovery log from mirrored copies	344
How to recover the database from backups	345
How to recover damaged files	344
Backup and recovery scenarios	352

In this chapter, most examples illustrate how to perform tasks by using the ADSM command line interface. For information about the ADSM commands, see *ADSM Administrator’s Reference*, or issue the HELP command from the command line of an ADSM administrative client.

Appendix C, “Interface Cross-Reference” on page 467 lists each command and shows if its function is also available on the administrative client GUI.

DRM

The Disaster Recovery Manager (DRM) feature (Chapter 17, “Using Disaster Recovery Manager” on page 359) automates many of the tasks associated with preparing for or recovering from a disaster. In this chapter, this icon identifies those tasks.

© Copyright IBM Corp. 1995, 1997

Levels of Protection

ADSM provides various methods for protecting ADSM data. For the most comprehensive coverage, they should be used together, They are:

- Storage pool backup
- Database and recovery log mirroring
- Database backup

This section describes each method and presents the benefits and costs of each.

Attention: ADSM Version 1 provided database salvage commands to re-establish the server database if a catastrophic error occurred. Although these commands are still available, the Version 2 database backup and recovery functions replace them and should be used to ensure the best level of protection for your server. Database salvage commands involve a lengthy process. You should not use them without help from your IBM service representative.

Storage Pool Protection

ADSM stores client data on volumes in storage pools. If one or more storage pool volumes is lost or damaged, the client data may be permanently lost. However, you can back up random or sequential access pools to sequential access copy storage pools and move the volumes offsite. Then if data is lost or damaged, you can restore individual volumes or entire storage pools from the data in the copy storage pools.

Database and Recovery Log Protection

In addition to all the information about your ADSM system, the database contains information (including pointers) about all the client data in your storage pools. The recovery log contains records of changes to the database. If you lose the recovery log, you lose the changes that have been made since the last database backup. If you lose the database, you lose all your client data.

You have several ways to protect this information:

- Mirror the database, or the recovery log, or both
- Back up the database to tape
- Back up the database to tape and in the recovery log save all the changes made to the database since that backup (this is called *roll-forward* mode)

Mirroring

You can prevent the loss of the database or recovery log due to a hardware failure, by mirroring them. Mirroring writes the same data to multiple disks simultaneously. However, mirroring does not protect against a disaster or a hardware failure that affects multiple drives or causes the loss of the entire system. While ADSM is running, you can dynamically start or stop mirroring and change the capacity of the database.

ADSM mirroring provides the following benefits:

- Protection against database and recovery log media failures

- Uninterrupted ADSM operations if a database or recovery log volume fails
- Avoidance of costly database recoveries

However, there are also costs:

- Mirroring doubles the required DASD for the mirrored volumes
- Mirroring results in decreased performance

Database Backup

ADSM can perform full and incremental backups of the database to tape while the server is running and available to clients. With ADSM in *normal* mode, the backup media can then be stored onsite or offsite and can be used to recover the database up to the point of the backup. You can run full or incremental backups as often as needed to ensure that the database can be restored to an acceptable point in time.

Note: You can run up to 32 incremental backups between full backups.

You can provide even more complete protection if you specify that ADSM run in *roll-forward* mode. With ADSM in *roll-forward* mode and with an intact recovery log, you can recover the database up to its most current state (the point at which the database was lost).

For the fastest recovery time and greatest availability of the database, mirror both the database and recovery log and periodically back up the database. When operating in roll-forward mode, mirroring better ensures that you have an intact recovery log, which is necessary to restore the database to its most current state.

Normal Mode versus Roll-Forward Mode: Roll-forward mode offers the highest level of protection for your data. However, there are costs to roll-forward mode. The following table describes the protection afforded by each mode and the requirements for each mode.

Level of Protection	
Normal Mode	**Roll-forward Mode**
Recover to a point in time of the latest full or incremental backup only	Recover to a point in time of the latest full or incremental backup or, with an intact recovery log, to the most current state
Recover with loss of client data that has been: • Backed up since the last database backup. • Moved due to storage pool migration, reclamation, or move data operations since the last database backup and then overwritten.	With an intact recovery log, recover to the most current state with no loss of client data
You must restore the entire database even if only one volume is damaged	You can restore a single volume
	Preferable if the server supports HSM clients (space-managed files should be protected as fully as possible from hardware failure)

Storage Requirements	
Normal Mode	**Roll-forward Mode**
Does not require a recovery log to restore to a point in time. The recovery log keeps only uncommitted transactions, and its size is not affected by normal mode.	Requires an intact recovery log to restore to the most current state. The recovery log keeps all transactions since the last database backup. In roll-forward mode you should significantly increase the recovery log size. However: • Frequent database backups reduce recovery log storage requirements (after a backup is completed, recovery log records preceding the backup are deleted). • Mirroring the recovery log requires much less space than mirroring the database.
For the greatest availability, you should mirror the database and recovery log or place them on devices that guarantee availability.	You should mirror the recovery log to recover to the most current state. **Note:** Unlike mirroring the database, roll-forward recovery does not provide continuous operations after a media failure. This is because the database must be brought down to perform the recovery.

The following table compares four typical ADSM data recovery configurations, two for roll-forward mode and two for normal mode. In all four cases, the storage pools and the database are backed up. The benefits and costs are:

Mirroring Whether the database and recovery log are mirrored. Mirroring costs additional disk space.

Coverage How completely you can recover your data. Roll-forward recovery cannot be done if the recovery log is not intact. However, roll-forward mode does support point-in-time recovery.

Speed to Recover How quickly data can be recovered

Mode	Mirroring	Coverage	Speed to Recover
Roll-Forward	Log and database	Greatest	Fastest
	Log Only	Medium	Moderate
Normal	Log and database	Medium	Moderate
	None	Least	Slowest

An Overview of the Process

Before you learn the details of protecting and recovering your data, read the following scenarios for protecting and recovering data. These scenarios are presented in detail in “Backup and Recovery Scenarios” on page 352.

DRM helps you perform the following tasks discussed in this section:

- Save the volume history and device configuration files
- Track the movement of storage pool and database backup volumes
- Restore the database
- Delete database volumes
- Define and restore primary storage pool volumes

Protecting Your Database and Storage Pool

1. Create a copy storage pool
2. Do a full backup of the primary storage pools to the copy storage pool
3. Do the following daily:
    a. Do incremental backups of the primary storage pools to copy storage pools
    b. Back up the database
    c. Save the volume history file (which describes ADSM volumes) the device configuration file (which describes the devices ADSM uses) and your server options
    d. Move offsite: copy storage pool volumes, database backup volumes, the volume history file, the device configuration file, and your server options

Recovering to a Point in Time

1. Install ADSM on a replacement processor
2. Move the database and storage pool backup volumes onsite
3. Restore the database from the latest backup level
4. Audit storage pool disk volumes and any tape volumes that were reused or added since the last backup.
5. Delete from the database any volumes in the copy storage pool that were onsite at the time of the disaster
6. Define new volumes in the primary storage pool
7. Restore the primary storage pool volumes from those in the copy storage pools

Recover a Lost or Damaged Storage Pool Volume

1. Identify the copy pool volumes containing backup copies of the files in the lost or damaged volume
2. Mark the storage pool backup volumes as unavailable
3. Bring the storage pool backup volumes onsite and mark them as read/write
4. Restore the destroyed files
5. Mark the storage pool backup volumes offsite and move them offsite

Backing Up Storage Pools

Task	Required Privilege Class
Define, back up, or restore storage pools Restore volumes	System, unrestricted storage, or restricted storage (only for those pools to which you authorized)
Update volumes	System or operator
Query volumes or storage pools	Any administrator

You can create backup copies of client files that are stored in your primary storage pools. The backup copies are stored in copy storage pools and can be used to restore the original files if they are damaged, lost, or unusable.

Primary storage pools should be backed up incrementally each day to the same copy storage pool (see Figure 57). Backing up to the same copy storage pool ensures that files do not need to be recopied if they have migrated to the next pool.

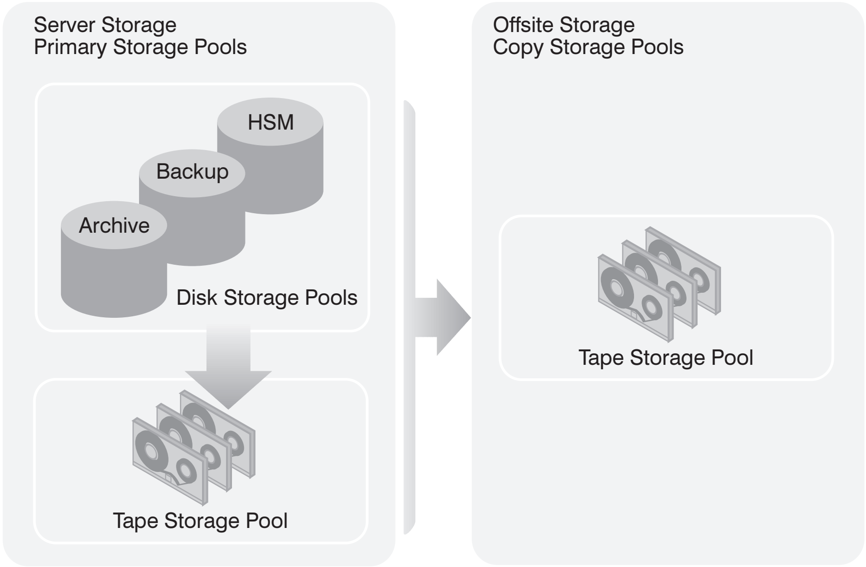


Figure 57. Copy Storage Pools

With scheduled storage pool backups and migrations and with sufficient disk storage, most copies can be made from the disk storage pool before the files are migrated to tape, thus avoiding unnecessary mounts. Here is the sequence:

1. Clients back up or archive data to disk
2. Back up the primary storage pools to copy storage pools
3. Data is migrated from disk storage pools to primary tape storage pools

Backing up storage pools requires an additional 200 bytes of space in the database for each file copy. As more files are added to the copy storage pools, reevaluate your database size requirements.

For recovery scenarios that involve backed up copies of storage pools, see “Recovering to a Point in Time from a Disaster” on page 354 and “Recovering a Lost or Damaged Storage Pool Volume” on page 356.

Mirroring the Database and Recovery Log

This section explains how to:

- Allocate disk volumes to mirror the database and recovery log
- Define ADSM mirrored volume copies
- Monitor ADSM mirrored volume copies

Task	Required Privilege Class
Define database and recovery log volumes	System or unrestricted storage
Query mirrored volumes	Any administrator

The following scenario shows the importance of mirroring in the recovery process: As the result of a sudden power outage, a partial page write occurs. The recovery log is now corrupted and not completely readable. Without mirroring, transaction recovery operations cannot complete when the server is restarted. However, if the recovery log is mirrored and a partial write is detected, a mirror volume can be used to construct valid images of the missing pages.

Allocating Volume Copies to Separate Physical Disks

By separating volume copies on different physical devices, you protect the server against media failure and increase the availability of the database and recovery log. If you cannot assign each volume copy to its own physical disk, then allocate them as shown in Table 14.

Table 14. Separating Volume Copies

Physical Disk	Volume	Volume
Physical Disk 1	Database volume copy 1	Recovery log volume copy 3
Physical Disk 2	Recovery log volume copy 1	Database volume copy 2
Physical Disk 3	Database volume copy 3	Recovery log volume copy 2

ADSM mirrored volumes must have at least the same capacity as the original volumes.

Defining Database or Recovery Log Mirrored Volumes

To mirror the database or recovery log, define a volume copy for each volume in the database or recovery log.

For example, the database consists of five volumes named VOL1, VOL2, VOL3, VOL4, and VOL5. To mirror the database, you must have five volumes that match the original volumes in size. Figure 58 on page 334 shows a mirrored database in which VOL1-VOL5 are mirrored by VOLA-VOLE.

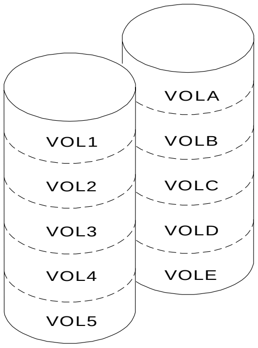


Figure 58. Mirrored Volumes

Format the space by using the DSMFMT command. For example, to format vola, a 25MB database volume, enter:

```
./dsmfmt -m -db vola 25
```

Then define the group of mirrored volumes by entering, for example, the following commands:

For example, the mirrored volumes are defined by entering the following commands:

```
define dbcopy vol1 vola

define dbcopy vol2 volb

define dbcopy vol3 volc

define dbcopy vol4 vold

define dbcopy vol5 vole
```

After a volume copy is defined, ADSM synchronizes the volume copy with the original volume. This process can range from minutes to hours, depending on the size of the volumes and performance of your system. After synchronization is complete, the volume copies are mirror images of each other.

Note: The mirror read and mirror write server options specify modes for reading and writing database and recovery log pages. See *ADSM Administrator's Reference* for details.

Requesting Information about Mirrored Volumes

You can request information about mirrored database or recovery log volumes by using the QUERY DBVOLUME and QUERY LOGVOLUME commands. For example:

```
query dbvolume
```

This command results in the following display:

```
Volume Name    Copy    Volume Name    Copy    Volume Name    Copy
(Copy 1)       Status  (Copy 2)       Status  (Copy 3)       Status
-----------    ------  -------------  ------  -------------  ------
VOL1           Sync'd  VOLA           Sync'd                 Undef-
VOL2           Sync'd  VOLB           Sync'd                 ined
VOL3           Sync'd  VOLC           Sync'd
VOL4           Sync'd  VOLD           Sync'd

VOL5           Sync'd  VOLE           Sync'd
```

- Each pair of vertical columns displays an image of the database or recovery log. For example, VOLA, VOLB, VOLC, VOLD, and VOLE (Copy 2) represent one image of the database.
- Each horizontal column displays a *group of mirrored volumes*. For example, VOL1, and VOLA represent two volume copies.

Backing Up the Database

Requesting a database backup (“Doing Full and Incremental Backups” on page 344) is a simple operation. However, before you do your first backup, you must take some or all of the following steps:

- Define device classes for backups (optional)
- Set the recovery log mode
- Adjust the recovery log size (optional)
- Set a database backup trigger (roll-forward mode only)

To restore your data, you must also save copies of the following information:

- Volume history file
- Device configuration file
- Server options file
- Database and recovery log set up (the output from detailed queries of your database and recovery log volumes)

DRM helps you save the previously listed information.

Defining Device Classes for Backups

You can use existing device classes for backups or define new ones. For incremental backups you can specify a device class different from the one used for full backups.

For example, you can write full backups to a tape device and incremental backups to a disk device. Specifying a device class with a device type of FILE is useful if an incremental backup is run based on a database backup trigger. You should do this only if you are also backing up the files to tape and taking them off site. Otherwise, in a disaster you can only restore the full backup.

You can also reserve one or more device classes and, therefore, mount points for automatic backups only. In this way, you avoid trying to run a backup based on the database backup trigger with no mount point available. If a database backup, which is a high priority operation, shares a device class with a low priority operation, such as reclamation, and all the mount points are in use, ADSM automatically cancels the lower priority operation. This frees a mount point for the database backup.

Note: Device class definitions are saved in the device configuration files (see “Saving the Device Configuration Backup File” on page 342).

Setting the Recovery Log Mode

You set the recovery log mode to either *normal* or *rollforward*. If you do not set the recovery log mode, ADSM runs in normal mode. See “Database Backup” on page 329 for a description of the two modes and for a comparison their benefits and costs.

To set the log mode to normal, enter:

```
set logmode normal
```

To set the log mode to roll-forward, enter:

```
set logmode rollforward
```

Note: The log mode is not in rollforward mode until you perform the first full database backup after entering this command.

Scheduling Database Backups

Database backups can tie up resources (mount points and tapes) and, depending on the type of backup and the size of your database, can take some time. You will probably want to schedule your backups to occur, when possible, after certain activities and at specific times of the day.

To ensure that you have the most recent database information, you might back up the database after activities such as:

- Significant backup or archive activities
- Migration between storage pools
- Reclamation
- MOVE DATA or DELETE VOLUME commands
- Storage pool backups

You would usually back up your storage pools daily and immediately back up the database. Depending on the amount of client data and frequency of the activities mentioned above, you may back up less often.

Consider the following when you decide what kind of backups to do and when to do them:

Full backups

- Take longer to run than incremental backups
- Have shorter recovery times than incremental backups (you must load only one set of volumes to restore the entire database)

Full backups are required:

- For the first backup
- If there have been 32 incremental backups since the last full backup
- After changing the log mode to roll-forward
- After changing the database size (an extend or reduce operation)

Incremental backups

- Take less time to run than full backups
- Have longer recovery times than full backups because a full backup must be loaded first

Estimating the Size of the Recovery Log

In both normal mode and roll-forward mode, the volume of ADSM transactions affects how large you should make your recovery log. As more clients are added and the volume of concurrent transactions increases, you can extend the size of the log. In roll-forward mode you must also consider how often you perform database backups. In this mode, the recovery log keeps all transactions since the last database backup and typically requires significantly more space than is required in normal mode.

How, then, do you determine how large your recovery log should be in roll-forward mode? You need to determine how much recovery log space is used between database backups. For example, if you plan daily incremental backups, you should check your daily usage over a period of time. You can use the following procedure to make your estimate:

1. Start by setting your log mode to normal. In this way you are less likely to exceed your log space if your initial setting is too low for roll-forward mode.
2. After a scheduled database backup, issue the following command to reset the statistic on the amount of recovery log space used since the last reset:

```
reset logconsumption
```

3. Just before the next scheduled database backup, issue the following command to display the current recovery log statistics:

```
query log format=detailed
```

   The **Cumulative Consumption** field contains the log space in megabytes used by the server since the statistic was last reset. Record the value.
4. Reiterate steps 2 and 3 over at least one week.
5. Increase the highest cumulative consumption value by 30 to 40 percent. Set your recovery log size to this increased value to account for periods of unusually high activity.

   For example, over a period of a week the highest cumulative consumption value was 500MB. If you set your recovery log to 650MB you should have sufficient space between daily backups.

For information on how to adjust the recovery log size, see "Adding Space to the Database or Recovery Log" on page 266 or "Deleting Space from the Database or Recovery Log" on page 271.

Note: If the recovery log runs out of space, you may not be able to start the server for normal operation. You can create an additional recovery log volume if needed to start the server and perform the needed database backup. For example, to create a 5MB volume A00, issue the following command:

```
> dsmserv extend log a00 5mb
```

Volume sizes are specified in multiples of 4MB plus 1 MB for overhead.

Setting a Database Backup Trigger

In roll-forward mode, a database backup trigger can cause ADSM to back up the database automatically. When the space occupied in the recovery log reaches a specified percentage, ADSM automatically runs a full or incremental backup of the database and deletes any unnecessary recovery log records.

Attention: The database backup trigger is intended to initiate a backup when you have scheduled a database backup but the recovery log utilization has grown faster than planned. It should not be used in place of coordinating your recovery log size and your scheduled backups. A database backup has a greater priority than many other operations. A backup based on a trigger could occur at a time of high activity and affect your other operations. To control the timing of scheduled database backups, adjust the recovery log size so that the trigger does not cause the database to be backed up at non-scheduled times.

Setting a database backup trigger is optional, but it is recommended to ensure that the recovery log does not run out of space before the next backup.

If the log mode is changed from normal to roll-forward, the next database backup must be a full backup. If a database backup trigger is defined when you set the log mode to roll-forward, the full backup is done automatically. The server does not start saving log records for roll-forward recovery until this full backup completes successfully.

In "Estimating the Size of the Recovery Log" on page 337 you determined the size of your recovery log. Your database backup trigger should be based on that procedure. For example, your recovery log typically consumes less than 500MB between backups, and your log size is 650MB. You do not want the trigger to initiate a backup except in unusual circumstances. Therefore you should set the trigger no lower than 75 percent (approximately 500MB).

To set the database backup trigger at 75 percent and run 20 incremental backups to every full backup, enter:

```
define dbbackuptrigger logfullpct=75 devclass=tape8mm
numincremental=20
```

If you do not specify the LOGFULLPCT and NUMINCREMENTAL parameters, the trigger defaults to 50 percent and ADSM runs 6 incremental backups to every full backup. Each incremental backup, whether automatic or by command, is added to the count of incremental backups run. Each full backup, whether automatic or by command, resets the count for incremental backups to zero. When you specify 0 for the NUMINCREMENTAL parameter, ADSM automatically runs only full backups.

Note: If you issue a BACKUP DB command with the TYPE=INCREMENTAL parameter, ADSM performs an incremental backup of the database regardless of the NUMINCREMENTAL setting. For example, you set NUMINCREMENTAL to 5, and there have been five incremental backups since the last full backup. If you then issue BACKUP DB TYPE=INCREMENTAL, an incremental backup is still taken, and the counter for the number of incremental backups since the last full backup is set to 6. This occurs if the BACKUP DB command is issued either by an administrator or through an administrative schedule.

After you set the database backup trigger, you might find that automatic backups occur too often. Check the backup trigger percentage by entering:

```
query dbbackuptrigger
```

ADSM displays the following information:

```
             Full Device Class: TAPE8MM
      Incremental Device Class: TAPE8MM
           Log Full Percentage: 75
    Incrementals Between Fulls: 6
Last Update by (administrator): SERVER_CONSOLE
         Last Update Date/Time: 03/06/1996 10:49:23
```

This information shows that the trigger is set to 75 percent. If automatic backups are occurring too often, you could increase the value to 80 percent by entering:

```
update dbbackuptrigger logfullpct=80
```

If the database backup trigger automatically runs backups more often than you want and the setting is high (for example, 90 percent or higher), you should probably increase the recovery log size. If you no longer want to use the database backup trigger, enter:

```
delete dbbackuptrigger
```

After you delete the database backup trigger, ADSM no longer runs automatic database backups.

Note: If you delete the trigger and stay in roll-forward mode, transactions fail when the log fills. Therefore, you should change the log mode to normal. Remember, however, that normal mode does not let you perform roll-forward recovery.

Saving the Volume History File

The volume history file contains information about the following:

- Sequential access storage pool volumes that have been added, reused (through reclamation or MOVE DATA operations), or deleted (during DELETE VOLUME operations or reclamation)
- Database backup volumes
- Export volumes for administrator, node, policy, and server data

ADSM updates the volume history file as volumes are added. However, you must periodically run a delete operation to discard outdated information about volumes (see “Deleting Volume History Information” on page 341 for details).

This information is stored in the database, but during a database restore, it is not available from there. To perform a restore, therefore, ADSM must get the information from the volume history file.

To ensure the availability of the volume history information, you can do any of the following:

- Store at least one copy of the volume history file on a disk separate from the database, or offsite
- Store a printout of the file offsite
- Store a copy of the file offsite with your database backups
- Store a remote copy of the file, for example, on an NFS-mounted file system

DRM saves snapshots of the volume history file in its disaster recovery plan file.

Note: You can recover the database without a volume history file. However, because you must examine every volume that may contain database backup information, this is a time consuming and error-prone task.

The VOLUMEHISTORY server option lets you specify backup volume history files (for details, see the *ADSM Administrator’s Reference*). After the server is restarted, whenever ADSM updates volume information in the database, it also updates the same information in the backup files specified by the VOLUMEHISTORY option.

You can also back up the volume history information at any time, by entering:

```
backup volhistory
```

If you do not specify file names, ADSM backs up the volume history information to *all* files specified with the VOLUMEHISTORY server option.

Deleting Volume History Information

You should periodically delete outdated information from the volume history file. For example, if you keep your backups for seven days, any information older than that is not needed (see the example below). When information about database backup volumes or export volumes is deleted, the volumes return to scratch status in the libraries attached to the server and may be reused. For scratch volumes with device type FILE, the files are deleted. When information about volumes in storage pools is deleted, the volumes themselves are not affected.

To display volume history information up to yesterday, enter:

```
query volhistory enddate=today-1
```

For example, to delete information that is seven days old or older, enter:

```
delete volhistory todate=today-7
```

DRM expires database backup series and deletes the volume history entries.

Saving the Device Configuration Backup File

The device configuration file contains information about the device classes, libraries, and drives needed to read backup data. Whenever ADSM updates device configuration information in the database, it updates the device configuration file.

This information is stored in the database, but during a database restore, it is not available from there. To perform a restore, therefore, ADSM must get the information from the device configuration file.

To ensure the availability of the device configuration information, you can do any of the following:

- Store at least one backup copy of the device configuration file on a disk separate from the database
- Store your device configuration file offsite with your volume history file and database backups
- Store a printout of the information stored offsite
- Store a remote copy, for example, on an NFS-mounted file system

DRM saves snapshots of the device configuration file in its disaster recovery plan file.

The DEVCONFIG server option lets you specify backup device configuration files (for details, see the *ADSM Administrator's Reference*). After the server is restarted, whenever ADSM updates device configuration information in the database, it also updates the same information in the backup files.

During a database restore operation, ADSM tries to open the first device configuration file. If it cannot open or read that file, ADSM tries to use any remaining device configuration files (in the order in which they occur in the server options) until it finds one that is usable. If none can be found, you must recreate the file. See “Recreating a Device Configuration File” on page 343 for details.

You can also back up the device configuration information at any time, by entering:

```
backup devconfig
```

If you do not specify file names, ADSM backs up the device configuration file to *all* files specified with the DEVCONFIG server option.

If you lose your device configuration file and need it to restore the database, you must recreate it manually. See “Recreating a Device Configuration File” for details.

If you are using automated tape libraries, ADSM also saves volume location information in the device configuration file. The file is updated whenever CHECKIN LIBVOLUME, CHECKOUT LIBVOLUME, and AUDIT LIBRARY commands are issued, and the information is saved as comments (/* ...... */). This information is used during restore or load operations to locate a volume in an automated library. If you must recreate the device configuration file, you will be unable to recreate the volume location information. Therefore, you must define your library as a manual library and manually mount the volumes during server processing.

Recreating a Device Configuration File

The following commands read and execute the device configuration file:

- DSMSERV RESTORE DB
- DSMSERV LOADDB
- DSMSERV DISPLAY DBBACKUPVOLUME

Note: The DSMSERV LOADDB command may increase the size of the database. The server packs data in pages in the order in which they are inserted. The DSMSERV DUMPDB utility does not preserve that order. Therefore, page packing is not optimimized, and the database may require additional space.

If no device configuration file is found, you must recreate it before you can start the restore operation. The device configuration file must follow these conventions:

- The commands must be in this order:
  - DEFINE DEVCLASS
  - DEFINE LIBRARY
  - DEFINE DRIVE

  You must provide those definitions needed to mount the volumes read by the ADSM command that you issued. If you are restoring or loading from a FILE device class, you will need only the DEFINE DEVCLASS command.

- You can use command defaults.

- The file can include blank lines.

- A single line can be up to 240 characters.
- The file can include continuation characters and comments as described in the *ADSM Administrator's Reference*.

The following figure shows an example of a device configuration file:

Doing Full and Incremental Backups

The first back up of your database must be a full backup. You can run up to 32 incremental backups between full backups.

To perform a full backup of your database to the TAPE8MM device class, for example, enter:

```
backup db type=full devclass=tape8mm
```

In this example, ADSM writes the backup data to scratch volumes. You can also specify volumes by name. After a full backup, you can perform incremental backups, which copy only the changes to the database since the previous backup.

To do an incremental backup of the database to the TAPE8MM device class, enter:

```
backup db type=incremental devclass=tape8mm
```

Recovering by Using Mirrored Volumes

If a mirrored volume fails due to media failure, you can recover the volume by taking the following steps:

1. View the status of the database and recovery log volumes (QUERY DBVOLUME or QUERY LOGVOLUME).
2. If necessary, place the failing volume offline from ADSM (DELETE DBVOLUME or DELETE LOGVOLUME). The server usually does this automatically.
3. Fix the failing physical device.
4. Allocate space to be used for a new volume (DSMFMT).
5. Bring the volume online (DEFINE DBCOPY or DEFINE LOGCOPY).

After a database or recovery log volume copy is defined, the server synchronizes the volume copy with its associated database or recovery log volume.

Recovering by Using Database and Storage Pool Backups

This section explains how to recover by using backups of the database and storage pools. Figure 59 shows the situation presented in the two scenarios in this section: an installation has lost its server, including the database and recovery log, and its onsite storage pools.

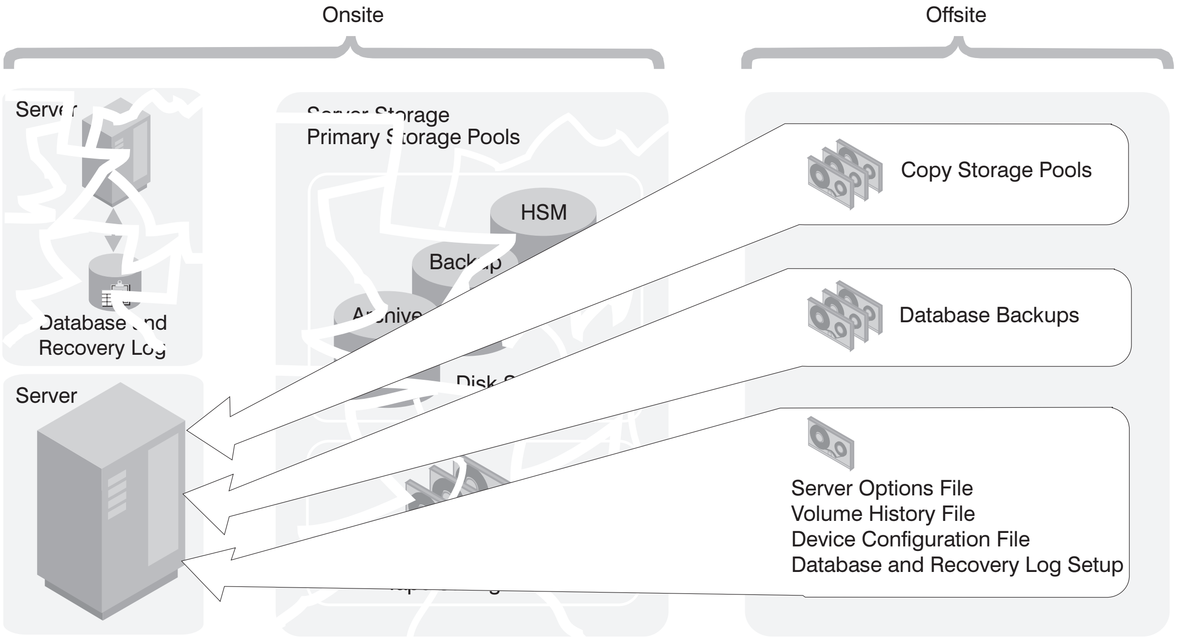


Figure 59. Recovery from a Disaster

The following topics are included:

- Restoring to a point in time
- Restoring to the most current state

To perform a restore, you should have the following information, preferably stored offsite (see Figure 59):

- A full database backup
- Any incremental database backups between the last full backup and the point in time to which you are recovering
- Copy storage pool volumes
- On tape or diskette, or as printouts:
  - Server options file
  - Volume history file
  - Device configuration file

– Database and recovery log setup (the output from detailed queries of your database and recovery log volumes)

DRM can query the ADSM server and generate a current, detailed disaster recovery plan for your installation.

Restoring a Database to a Point in Time

Point-in-time recovery is normally used for situations such as disaster recovery or to remove the effects of errors that can cause inconsistencies in the database.

Here is the procedure for restoring the database:

1. Rename and save a copy of the volume history file if it exists. After the database is restored, any volume history information pointed to by the server options is lost. You will need this information to identify the volumes to be audited.
2. If the device configuration file is unavailable, recreate it manually (see "Recreating a Device Configuration File" on page 343). Put the recreated file in the server work library. You can do the same with the server options file.
3. If the original database or recovery log volumes were lost, issue the DSMSERV INSTALL command to initialize the database and recovery log.

```
> dsmserv install 1 log1 1 dbvol1
```

   Attention: Do not start the server until *after* you restore the database (the next step). Starting the server before the restore would destroy any existing volume history files.

4. Issue the DSMSERV RESTORE DB command. For example, to restore the database to a backup series that was created on April 19, 1996, enter:

```
> dsmserv restore db todate=04/19/1996
```

   ADSM does the following:

   a. Reads the volume history file to locate the last full backup that occurred on or before the specified date and time.

      Note: If the volume history file is not available, you must mount tape volumes in the correct order or specify their order on the DSMSERV RESTORE DB command.

   b. Using the device configuration file, requests a mount of the first volume, which should contain the beginning of the full backup.
   c. Restores the backup data from the first volume.

d. Continues to request mounts and to restore data from the backup volumes that contain the full backup and any incremental backups that occurred on or before the date specified.

From the old volume history information (generated by the QUERY VOLHISTORY command) you need a list of all the volumes that were reused (STGREUSE), added (STGNEW), and deleted (STGDELETE) since the original backup. Use this list to perform the following steps.

5. Audit all disk volumes, all reused volumes, and any deleted volumes located by the AUDIT VOLUME command with the FIX=YES parameter.

   This process identifies files recorded in the database that can no longer be found on the volume. If a copy of the file is in a copy storage pool, the file on the audited volume is marked as damaged. Otherwise, the file is deleted from the database and is lost.

6. If the audit detects any damaged files, issue the RESTORE STGPOOL command to restore those files after you have audited the volumes in the storage pool. Include the FIX=YES parameter on the AUDIT VOLUME command to delete database entries for files not found in the copy storage pool.

7. Mark as destroyed any volumes that cannot be located, and recover those volumes from copy storage pool backups. If no backups are available, delete the volumes from the database by using the DELETE VOLUME command with the DISCARDDATA=YES parameter.

8. Redefine any storage pool volumes that were added since the database backup.

Some files may be lost if they were moved since the backup (due to migration, reclamation, or move data requests) and the space occupied by those files has been reused. You can minimize this loss by using the REUSEDELAY parameter when defining or updating sequential access storage pools. This parameter delays volumes from being returned to scratch or being reused.

By backing up your storage pool and your database, you reduce the risk of losing data. To further minimize loss of data, you can:

- Mark the backup volumes in the copy storage pool as OFFSITE and move them to an offsite location.

  In this way the backup volumes are preserved and are not reused or mounted until they are brought onsite. Ensure that you mark the volumes as OFFSITE before you back up the database.

- Back up the database immediately after you back up the storage pools.

- Turn off migration and reclamation while you back up the database.

- Do not perform any MOVE DATA operations while you back up the database.

- Use the REUSEDELAY interval to prevent your copy storage pool volumes from being reused or deleted before they might be needed.

If your old volume history file shows that any of the copy storage pool volumes needed to restore your storage pools have been reused (STGREUSE) or deleted (STGDELETE), you may not be able to restore all your files. You can avoid this problem by including the REUSEDELAY parameter when you define your copy storage pools.

After a restore, the volume inventories for ADSM and for your tape management system may be inconsistent. For example, after a database backup, a new volume is added to ADSM. The tape management system inventory records the volume as belonging to ADSM. If the database is restored from the backup, ADSM has no record of the added volume, but the tape management system does. You must synchronize these inventories.

Point-in-Time Restore without a Volume History File

If you are doing a point-in-time restore and a volume history file is not available, you must enter the volume names in the DSMSERV RESTORE DB command in the sequence in which they were written to. First, however, issue the DSMSERV DISPLAY DBBACKUPVOLUME command to read your backup volumes and display the information needed to arrange them in order (backup series, backup operation, and volume sequence):

```
> dsmserv display dbbackupvolume devclass=tape8mm
  volumenames=dsm012,dsm023,dsm037,dsm038,dsm058,dsm087
```

For example, the most recent backup series consists of three operations:

0 A full backup on three volumes in the sequence dsm023, dsm037, and dsm087

1 An incremental backup on one volume, dsm012

2 An incremental backup on two volumes in the sequence dsm038 and dsm058

You would issue three commands in the following order:

```
> dsmserv restore db volumenames=dsm023,dsm037,dsm087
  devclass=tape8mm commit=no

> dsmserv restore db volumenames=dsm012
  devclass=tape8mm commit=no

> dsmserv restore db volumenames=dsm038,dsm058
  devclass=tape8mm commit=no
```

Attention: If the original database or recovery log volumes are available, you issue only the DSMSERV RESTORE DB command. However, if those volumes have been lost, you must first issue the DSMSERV INSTALL command to initialize the database and recovery log, then issue the DSMSERV RESTORE DB command.

Storage Pool Backups in Point-of-Time Restore

The following example shows the importance of storage pool backups with a point-in-time restore. In this example, the storage pool was not backed up with the BACKUP STGPOOL command.

9:30 a.m. Client A backs up its data to Volume 1.

Noon The system administrator backs up the database.

1:30 p.m. Client A's files on Volume 1 (disk), is migrated to tape (Volume 2).

3:00 p.m. Client B backs up its data to Volume 1.

The server places Client B's files in the location that contained Client A's files prior to the migration.

3:30 p.m. The server goes down.

3:40 p.m. The system administrator reloads the noon version of the database by using the DSMSERV RESTORE DB command.

4:40 p.m. Volume 1 is audited. The following then occurs:

1. The server compares the information on Volume 1 and with the restored database (which matches the database at noon).
2. The audit does not find Client A's files on Volume 1 where the reloaded database indicates they should be. Therefore, the server deletes these Client A file references.
3. The database has no record that Client A's files are on Volume 2, and the files are, in effect, lost.
4. The database has no record that Client B's files are on Volume 1, and the files are, in effect, lost.

If roll-forward recovery had been used, the database would have been rolled forward to 3:30 p.m. when the server went down, and neither Client A's files nor Client B's files would have been lost. If a point-in-time restore of the database had been performed and the storage pool had been backed up, Client A's files would not have been deleted by the volume audit and could have been restored with a RESTORE VOLUME or RESTORE STGPOOL command. Client B's files would still have been lost, however.

Restoring a Database to its Most Current State

You can use roll-forward recovery to restore a database to its most current state if:

- ADSM has been in roll-forward mode continuously from the time of the last full backup to the time the database was damaged or lost.
- The last backup series created for the database is available. A backup series consists of a full backup, all applicable incremental backups, and all recovery log records for database changes since the last backup in the series was run.

To restore the database to its most current state, enter:

```
> dsmserv restore db
```

Attention: If the original database or recovery log volumes are available, you issue only the DSMSERV RESTORE DB command. However, if those volumes have been lost, you must first issue the DSMSERV INSTALL command to initialize the database and recovery log, then issue the DSMSERV RESTORE DB command.

Note: Roll-forward recovery does not apply if all recovery log volumes are lost. However, with the server running in roll-forward mode, you can still perform point-in-time recovery in such a case.

Correcting Damaged Files

A data-integrity error can be caused by such things as a tape deteriorating or being overwritten or by a drive needing cleaning. If a data-integrity error is detected when a client tries to restore, retrieve, or recall a file or during a volume audit, ADSM marks the file as damaged. If the same file is stored in other copy storage pools, the status of those file copies is not changed.

If a client tries to access a file that is marked as damaged and an undamaged copy is available on an onsite copy storage pool volume, ADSM sends the user the undamaged copy.

Files that are marked as damaged cannot be:

- Restored, retrieved, or recalled
- Moved by migration, reclamation, or the MOVE DATA command
- Backed up during a BACKUP STGPOOL operation if the primary file is damaged
- Restored during a RESTORE STGPOOL or RESTORE VOLUME operation if the backup copy in a copy storage pool is damaged

Maintaining the Integrity of Files

To maintain the data integrity of user files, you can:

1. Detect damaged files before the users do.

   The AUDIT VOLUME command marks a file as damaged if a data-integrity error is detected for the file. If an undamaged copy is in an onsite copy storage pool, it is used to provide client access to the file.

2. Reset the damaged status of files if the error that caused the change to damaged status was temporary.

   You can use the AUDIT VOLUME command to correct situations when files are marked damaged due to a temporary hardware problem, such as a dirty tape head. ADSM resets the damaged status of files if the volume in which the files are stored is audited and no data-integrity errors are detected.

3. Correct files that are marked as damaged.

   If a primary file copy is marked as damaged and a usable copy exists in a copy storage pool, the primary file can be corrected using the RESTORE VOLUME or RESTORE STGPOOL command. For an example, see “Restore Damaged Files.”

4. Regularly run commands to identify files that are marked as damaged:

   - The RESTORE STGPOOL command displays the name of each volume in the restored storage pool that contains one or more damaged primary files. Use this command with the preview option to identify primary volumes with damaged files without actually performing the restore operation.
   - The QUERY CONTENT command with the DAMAGED option lets you display damaged files on a specific volume.

For an example of how to use these commands, see “Restore Damaged Files.”

Restore Damaged Files

If you use copy storage pools, you can restore damaged client files. You can also check storage pools for damaged files and restore the files. This section explains how to restore damaged files based on the scenario in “Example: Simple Hierarchy with One Copy Storage Pool” on page 143.

If a client tries to access a file stored in 8MM-POOL and a data integrity error occurs, the file in 8MM-POOL is automatically marked as damaged. Future accesses to the file automatically use the copy in 8MM-COPYPOOL as long as the copy in 8MM-POOL is marked as damaged.

To restore any *damaged* files in 8MM-POOL, you can define a schedule that issues the following command periodically:

```
restore stgpool 8mm-pool
```

You can check for and replace any files that develop data-integrity problems in 8MM-POOL or in 8MM-COPYPOOL. For example, every three months, query the volumes in 8MM-POOL and 8MM-COPYPOOL by entering the following commands:

```
query volume stgpool=8mm-pool

query volume stgpool=8mm-copypool
```

Then issue the following command for each volume in 8MM-POOL and 8MM-COPYPOOL:

```
audit volume <volname> fix=yes
```

If a data integrity error occurs on a file in 8MM-POOL, that file is marked *damaged* and an error message is produced. If a data integrity error occurs on file in 8MM-COPYPOOL, that file is deleted and a message is produced.

Restore *damaged* primary files by entering:

```
restore stgpool 8mm-pool
```

Finally, create new copies in 8MM-COPYPOOL by entering:

```
backup stgpool 8mm-pool 8mm-copypool
```

Backup and Recovery Scenarios

This section presents scenarios for protecting and recovering an ADSM server. You can modify the procedures to meet your needs.

DRM can help you track your onsite and offsite volumes and query the ADSM server and generate a current, detailed disaster recovery plan for your installation.

These scenarios assume a storage hierarchy consisting of:

- The default random access storage pools (BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL)
- TAPEPOOL, a tape storage pool

Protecting Your Database and Storage Pool

A company's standard procedures include the following:

- Perform reclamation of its copy storage pool, once a week. Reclamation for the copy storage pools is turned off at other times.

  Note: In a copy storage pool definition, the REUSEDELAY parameter delays volumes from being returned to scratch or being reused. The value should be set high enough to ensure that the database can be restored to an earlier point in time and that the database references to files in the storage pool is still valid. For example, a user may want to retain database backups for seven days and, therefore, sets REUSEDELAY to 7.

- Back up its storage pools every night.
- Perform a full backup of the database once a week and incremental backups on the other days.

- Ship the database and copy storage pool volumes to an offsite location every day.

To protect client data, the administrator does the following:

1. Creates a copy storage pool named DISASTER-RECOVERY. Only scratch tapes are used, and the maximum number of scratch volumes is set to 100. The copy storage pool is defined by entering:

```
define stgpool disaster-recovery 8mm_class pooltype=copy
maxscratch=100
```

2. Performs the first backup of the primary storage pools.

   Note: The first backup of a primary storage pool is a full backup and, depending on the size of the storage pool, could take a long time.

3. Defines schedules for the following daily operations:

   a. Incremental backups of the primary storage pools each night by issuing:

```
backup stgpool backuppool disaster-recovery maxprocess=2

backup stgpool archivepool disaster-recovery maxprocess=2

backup stgpool spacemgpool disaster-recovery maxprocess=2

backup stgpool tapepool disaster-recovery maxprocess=2
```

   These commands use multiple, parallel processes to perform an incremental backup of each primary storage pool to the copy pool. Only those files for which a copy does not already exist in the copy pool are backed up.

   Note: Migration should be turned off during the rest of the day. You could add a schedule to migrate from disk to tape at this point. In this way, the backups are done while the files are still on disk.

   b. Change the access mode to OFFSITE for volumes that have read-write or read-only access, are onsite, and are at least partially filled. This is done by entering:

```
update volume * access=offsite location='vault site info'
wherestgpool=disaster-recovery whereaccess=readwrite,readonly
wherestatus=filling,full
```

   c. Back up the database by entering:

```
backup db type=incremental devclass=tape8mm scratch=yes
```

4. Does the following operations nightly after the scheduled operations have completed:

    a. Backs up the volume history, device configuration, and server options.

    b. Moves the volumes marked offsite, the database backup volumes, volume history files, device configuration files, and server options to the offsite location.

    c. Identifies offsite volumes that should be returned onsite by using the QUERY VOLUME command:

```
query volume stgpool=disaster-recovery access=offsite status=empty
```

    These volumes, which have become empty through expiration, reclamation, and file space deletion, have waited the delay time specified by the REUSEDELAY parameter. The administrator periodically returns outdated backup database volumes. These volumes are displayed with the QUERY VOLHISTORY command and can be released for reuse with the DELETE VOLHISTORY command.

5. Brings the volumes identified in step 4c onsite and updates their access to read-write.

Recovering to a Point in Time from a Disaster

In this scenario, the processor on which ADSM resides, the database, and all onsite storage pool volumes are destroyed by fire. An administrator must restore the server to the point in time of the last backup.

DRM can help you perform these steps.

Do the following:

1. Install the ADSM server on the replacement processor with the same server options and the same size database and recovery log as on the destroyed system. For example, to initialize the database and recovery log, enter:

```
> dsmserv install 1 log1 1 dbvol1
```

2. Move the latest backup and all of the DISASTER-RECOVERY volumes onsite from the offsite location.

    Note: Do not change the access mode of these volumes until after you have completed step 7 on page 355.

3. If a current, undamaged volume history file exists, save it.

4. Restore the volume history and device configuration files and the server options.

5. Restore the database from the latest backup level by issuing the DSMSERV RESTORE DB command (see “Recovering by Using Database and Storage Pool Backups” on page 345).

6. Change the access mode of all the existing primary storage pool volumes in the damaged storage pools to DESTROYED by entering:

```
update volume * access=destroyed wherestgpool=backuppool

update volume * access=destroyed wherestgpool=archivepool

update volume * access=destroyed wherestgpool=spacemgpool

update volume * access=destroyed wherestgpool=tapepool
```

7. Issue the QUERY VOLUME command to identify any volumes in the DISASTER-RECOVERY storage pool that were onsite at the time of the disaster. Any volumes that were onsite would have been destroyed in the disaster and could not be used for restore processing. Delete each of these volumes from the database by using the DELETE VOLUME command with the DISCARDDATA option. Any files backed up to these volumes cannot be restored.

8. Change the access mode of the remaining volumes in the DISASTER-RECOVERY pool to READWRITE by entering:

```
update volume * access=readwrite wherestgpool=disaster-recovery
```

   Note: Clients can get files from ADSM at this point. If a client tries to get a file that was stored on a destroyed volume, the retrieval request goes to the copy storage pool. In this way, clients can access their files without waiting for the primary storage pool to be restored. When you update volumes brought from offsite to change their access, you greatly speed recovery time.

9. Define new volumes in the primary storage pool so the files on the damaged volumes can be restored to the new volumes. The new volumes also allow clients to backup, archive, or migrate files to the server. You do not need to perform this step if you use only scratch volumes in the storage pool.

10. Restore files in the primary storage pool from the copies located in the DISASTER-RECOVERY pool by entering:

```
restore stgpool backuppool maxprocess=2

restore stgpool archivepool maxprocess=2

restore stgpool spacemgpool maxprocess=2

restore stgpool tapepool maxprocess=2
```

These commands use multiple parallel processes to restore files to primary storage pools. After all the files have been restored for a destroyed volume, that volume is automatically deleted from the database. See “When a Storage Pool Restoration is Incomplete” on page 156 for what to do if one or more volumes cannot be fully restored.

11. To ensure against another loss of data, immediately back up all storage volumes and the database. Then resume normal activity, including weekly disaster backups and movement of data to the offsite location.

Recovering a Lost or Damaged Storage Pool Volume

If a company makes the preparations described in “Protecting Your Database and Storage Pool” on page 352 it can recover from a media loss by using ADSM features.

In this scenario, an operator inadvertently destroys a tape volume (DSM087) belonging to the TAPEPOOL storage pool. An administrator performs the following actions to recover the data stored on the destroyed volume by using the offsite copy storage pool:

1. Determine the copy pool volumes that contain the backup copies of the files that were stored on the volume that was destroyed by entering:

```
restore volume dsm087 preview=volumesonly
```

   This command produces a list of offsite volumes that contain the backed up copies of the files that were on tape volume DSM087.

2. Set the access mode of the storage pool backup volumes identified as UNAVAILABLE to prevent reclamation.

   Note: This precaution prevents the movement of files stored on these volumes until volume DSM087 is restored.

3. Bring the identified volumes to the onsite location and set their access mode to READWRITE.

4. Restore the destroyed files by entering:

```
restore volume dsm087
```

   This command sets the access mode of the DSM087 to DESTROYED and attempts to restore all the files that were stored on volume DSM087. The files are not actually restored to volume DSM087, but to another volume in the TAPEPOOL storage pool. All references to the files on DSM087 are deleted from the database and the volume itself is deleted from the database.

5. Set the access mode of the volumes used to restore DSM087 to OFFSITE using the UPDATE VOLUME command.

6. Return the volumes to the offsite location.

Chapter 17. Using Disaster Recovery Manager

Disaster Recovery Manager (DRM) is an optional feature of ADSM. This feature offers assistance with preparing a disaster recovery plan and facilitates an ADSM-based recovery of business applications. With DRM recovery can potentially be performed at an alternate site, on replacement computer hardware, by people not familiar with the backed up applications.

The disaster recovery plan is an essential document for any administrator recovering from a disaster. This document is very useful for audit purposes, such as certifying the recoverability of the ADSM server. DRM provides automated generation of the server disaster recovery plan file, offsite recovery media management, and storage of client recovery information.

The sections listed in the following table begin at the indicated pages.

Section	Page
Concepts:	
Comparing Availability Management to Disaster Recovery Management	359
Features of Disaster Recovery Manager	360
About the Disaster Recovery Plan File	370
Offsite Recovery Media Management	363
Tasks:	
Enabling Disaster Recovery Manager	362
Creating a backup copy of server primary storage pools and database	363
Sending server backup volumes offsite	365
Moving reclaimed or expired volumes back onsite	367
Creating the ADSM server disaster recovery plan file	369
Storing client machine information	399
Defining and tracking client recovery media	402
Customizing disaster recovery manager	410

Comparing Availability Management to Disaster Recovery Management

This section compares the definitions of availability management with disaster recovery management to show how DRM works with existing backup features of ADSM to provide disaster recovery.

© Copyright IBM Corp. 1995, 1997

Availability Management

Recovery from incidental computer system outages such as disk drive crashes. Down time is often minimized by using disk mirroring and other forms of RAID technology or by maintaining onsite backup copies of data.

Availability management for the ADSM server can be accomplished with ADSM by:

- Mirroring the server database and recovery log
- Backing up storage pools and storing them onsite

Disaster Recovery Management

A disaster is a catastrophic interruption of business processing that destroys the ADSM server or clients, or both. Backup data is located offsite to protect it from damage.

Disaster recovery management is accomplished with ADSM by:

- Backing up client data to the ADSM server
- Backing up the server database to removable media and storing the media offsite
- Backing up the primary storage pools and storing the media offsite
- Using the disaster recovery plan file to assist with the ADSM server recovery

Features of Disaster Recovery Manager

Disaster Recovery Manager provides the following features:

- Automated generation of a server disaster recovery plan
- Offsite recovery media management
- Storage of client recovery information

Automated Generation of a Server Disaster Recovery Plan

The PREPARE command automatically queries the required information from the ADSM server and generates a recovery plan file that is based on a pre-defined recovery strategy for the server. The PREPARE command can be scheduled using the ADSM central scheduling capabilities to maintain an up-to-date recovery plan.

The recovery plan file contains the information and procedures necessary to assist with the recovery of the ADSM server. The information in the plan file includes:

- Site-specific server recovery instructions as defined by the administrator (for example, contact names and telephone numbers).
- The sequence of steps necessary to recover an ADSM server.
- List of ADSM database backup and copy storage pool volumes required to perform the recovery. The offsite location where the volumes reside is included.
- Devices required to read the database backup and copy storage pool volumes.
- Space requirements for the ADSM database and recovery log.

- Copy of ADSM server options file, device configuration file, and volume history information file.
- Commands for performing server database recovery and primary storage pool recovery.
- Machine and recovery media information as defined by the administrator (for example, location of the machine that contains the server and boot media requirements).

Offsite Recovery Media Management

Knowing the location of offsite recovery media is critical to successful disaster recovery. You can perform the following with DRM's offsite recovery media management:

- Determine what database backup volumes and copy storage pool volumes need to be moved offsite and back onsite.
- Track the media location in the ADSM database.

Database backup volumes and copy storage pool volumes can be treated as logical collections that are selected to move offsite for safekeeping and onsite for reuse or disposal. The reclamation of offsite volumes includes the capability to perform expiration of an ADSM database backup series.

Storage of Client Recovery Information

DRM allows the following client recovery information to be saved in the ADSM database:

- Business priority
- Machine location, machine characteristics, and machine recovery instructions
- Boot media requirements

In the event of a disaster, DRM query commands provide assistance to help you determine:

- What client machines were lost in the disaster and need to be recovered.
- The priority of the client machines to identify the order to recover machines.
- The machine requirements and boot media requirements.

Overview of Disaster Recovery Manager Setup

This section provides an overview of the tasks involved to begin using DRM. Additional details are provided in subsequent sections, and a checklist is provided, see “ADSM DRM Project Plan” on page 420.

Enabling Disaster Recovery Manager

1. Enable the ADSM server to support DRM by registering the DRM license.

Set Up for Server Recovery

1. Optionally define information about the machine that contains the ADSM server.

2. Create backup copies of the server primary storage pools.
3. Create a backup copy of the database.
4. Track the movement of server backup volumes offsite using the MOVE DRMEDIA commands.
5. Create the disaster recovery plan file for the ADSM server by using the PREPARE command.

Set Up for Storage of Client Recovery Information

1. Identify and prioritize ADSM clients based on application or business needs, and establish automatic schedules for backing up client data.
2. Define your disaster recovery information for the clients by saving machine information in the ADSM database to include:
   - Business priority and machine location.
   - Associate one or more nodes with a machine.
   - Machine characteristics.
   - Recovery instructions.
3. Define the boot media requirements for the client machines in the ADSM database.
4. Associate one or more machines with the recovery media.

Enabling Disaster Recovery Manager

Task	Required Privilege Class
Register license for Disaster Recovery Manager	System

To enable DRM you must register the DRM license, please, see “Registering the Disaster Recovery Manager (DRM) Feature” on page 282 for details.

Defining Machine Information for the ADSM Server

You can *optionally* store details about the machine on which an ADSM server resides. The PREPARE command automatically retrieves this information stored in the ADSM database and places it in the plan file. This information can help you rebuild the replacement machine.

To store information about the machine that contains the ADSM server, issue the DEFINE MACHINE command (see “Defining Machine Information” on page 399) and set the MACHINE ADSMSERVER parameter to YES.

You can also include additional details about hardware, software, and boot media information by following the steps in “Defining Machine Information” on page 399 and “Defining and Tracking Recovery Media” on page 402.

Creating a Backup Copy of Server Primary Storage Pools and Database

Before using DRM to create an ADSM server disaster recovery plan file, you must create a backup copy of your primary storage pools and database.

The following table lists the required privilege classes for performing the tasks in this section.

Task	Required Privilege Class
Backing up your primary storage pools	System, unrestricted storage, or restricted storage
Backing up your database	System or unrestricted storage

Use the following backup commands to create a backup copy of the server primary storage pools and database volumes.

1. Back up your primary storage pools. For example:

```
backup stgpool backuppool cstoragepf
```

   For more information on backing up your storage pools, see "Backing Up Storage Pools" on page 332.

2. Back up your database. For example:

```
backup db devclass=lib8mm type=full volumename=bk06
```

For more information on backing up your database, see "Backing Up the Database" on page 335.

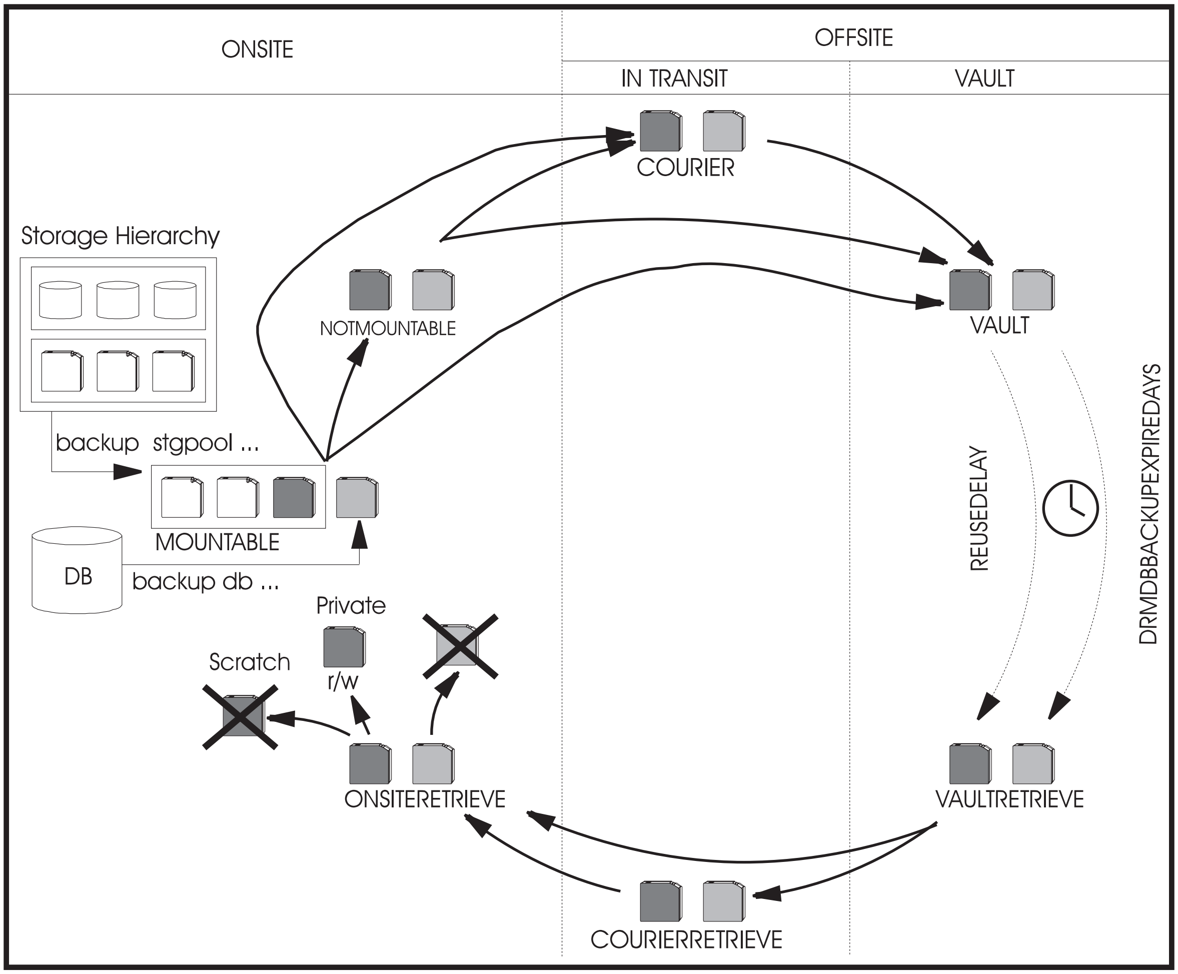
After you create your backup media, send it offsite for safekeeping. For more information, see "Sending Server Backup Volumes Offsite" on page 365.

When your backup media has been marked offsite, you are ready to create a disaster recovery plan file. For more information, see "Creating the ADSM Server Disaster Recovery Plan File" on page 369.

Offsite Recovery Media Management

Task	Required Privilege Class
Sending backup volumes offsite and back onsite	Unrestricted storage or operator

The following diagram is an overview of the recovery media life cycle:



Offsite recovery media management is used during routine operations and defines a process for the following:

- Moving ADSM database backup and copy storage pool volumes offsite for disaster recovery protection.
- Moving ADSM database backup and copy storage pool volumes onsite when they no longer contain valid data.

You can indicate the movement of the volumes with the MOVE DRMEDIA command and display and track their location with the QUERY DRMEDIA command.

Backup volume location information is included in the disaster recovery plan file that is generated by the PREPARE command. In the event of an actual disaster, for example the ADSM server is destroyed, the disaster recovery plan file can be used to provide a list of offsite volumes required at the recovery site. Refer to *ADSM Administrator's Reference* for a description of the PREPARE command.

Sending Server Backup Volumes Offsite

DRM uses the following states for database backup and copy storage pool volumes that are sent offsite for disaster recovery protection. The location of a volume is known at each state.

Volume State **Description**

MOUNTABLE The volume contains valid data and is accessible to the ADSM server.

NOTMOUNTABLE The volume contains valid data and is unavailable to the ADSM server, but is still onsite.

COURIER The volume contains valid data and is with the courier.

VAULT The volume contains valid data and is at the vault.

After you have created a backup copy of your primary storage pools and database, you can send your backup media offsite.

To send server backup media offsite, you must mark the volumes as unavailable for ADSM access, and then give the volumes to the courier. Use the following commands to identify the backup volumes written to by the ADSM server backup database and backup storage pool commands, and move these volumes offsite.

1. Issue the following command to identify the newly created copy storage pool and database backup volumes to be moved offsite:

```
query drmedia * wherestate=mountable
```

   ADSM displays information similar to the following:

```
Volume Name      State             Last Update          Automated
                                   Date/Time            LibName
---------------  ----------------  -------------------  -----------------
 TPBK05          Mountable         01/01/1997 12:00:31  LIBRARY
 TPBK99          Mountable         01/01/1997 12:00:32  LIBRARY
 TPBK06          Mountable         01/01/1997 12:01:03  LIBRARY
```

2. Indicate the movement of copy storage pool volumes and database backup volumes whose current state is MOUNTABLE by issuing the following command:

```
move drmedia * wherestate=mountable
```

This command automatically completes the following process for all volumes with a current state of MOUNTABLE:

- If the volume resides in an automated library, the volume is checked out of the library.
- Updates the volumes' state to NOTMOUNTABLE.
- Update the volumes' location according to the SET DRMNOTMOUNTABLENAME. If the SET command has not yet been issued, the default location is NOTMOUNTABLE. For more information, refer to the SET DRMNOTMOUNTABLENAME command.
- Updates the copy storage pool volumes' access mode to offsite.

3. Package the volumes and give them to the courier for transport to the offsite vault. Issue the following command to have ADSM select volumes whose current state is NOTMOUNTABLE, and record the fact that the volumes have been given to the courier.

```
move drmedia * wherestate=notmountable
```

This command automatically completes the following process for all volumes with a current state of NOTMOUNTABLE:

- Updates the volumes' state to COURIER.
- Updates the volumes' location according to the SET DRMCOURIERNAME. If the SET command has not yet been issued, the default location is COURIER. For more information, see "Specify the Courier Name" on page 413.

Your media containing backed up storage pools and database are now offsite.

4. When the vault location confirms receipt of the volumes, issue the MOVE DRMEDIA command with the WHERESTATE=COURIER parameter. For example:

```
move drmedia * wherestate=courier
```

This command automatically completes the following process for all volumes with a current state of COURIER:

- Updates the volumes' state to VAULT.
- Updates the volumes' location according to the SET DRMVAULTNAME command. If the SET command has not yet been issued, the default location is VAULT. For more information, see "Specify the Vault Name" on page 415.

5. To display a list of volumes that contain valid data at the vault, issue the following command:

```
query drmedia wherestate=vault
```

ADSM displays information similar to the following:

```
Volume Name       State           Last Update          Automated
                                  Date/Time            LibName
----------------- --------------  -------------------  -----------------
TAPE0P            Vault           01/05/1997 10:53:20
TAPE1P            Vault           01/05/1997 10:53:20
DBT02             Vault           01/05/1997 10:53:20
TAPE3S            Vault           01/05/1997 10:53:20
```

6. If you do not want to step through all the states, you can use TOSTATE parameter on the MOVE DRMEDIA command to specify the destination state. For example, to transition the volumes from NOTMOUNTABLE state to VAULT state, issue the following command:

```
move drmedia * wherestate=notmountable tostate=vault
```

   This command automatically completes the following process for all volumes with a current state of NOTMOUNTABLE:

   - Updates the volumes' state to VAULT.
   - Updates the volumes' location according to the SET DRMVAULTNAME command. If the SET command has not yet been issued, the default location is VAULT.

See "Example: Routine Operations Using Disaster Recovery Manager" on page 396 for an example that demonstrates sending server backup volumes offsite using MOVE DRMEDIA and QUERY DRMEDIA commands.

Moving Reclaimed or Expired Volumes Back Onsite

DRM uses the following states for backup volumes that are reclaimed or no longer contain valid data and are to be moved back onsite.

Volume State — **Description**

VAULTRETRIEVE
The volumes no longer contain valid data. These volumes are to be returned. They should be given to the courier by the vault operator. For more information on reclamation of offsite copy storage pool volumes, see "Reclamation of Offsite Volumes" on page 129. For information on expiration of database backup volumes, see step 1 on page 368 below.

COURIERRETRIEVE
The volumes no longer contain valid data and are in the process of being returned by the courier.

ONSITERETRIEVE The volumes no longer contain valid data and have been moved back to the onsite location. The volume records of ADSM database backup and scratch copy storage pool volumes are deleted from the ADSM database. For private copy storage pool volumes, the access mode is updated to READWRITE.

When backup volumes stored at the vault location no longer contain valid data, use the following procedure to move those volumes back onsite for reuse or disposal.

1. Use the SET DRMDBBACKUPEXPIREDAYS command to specify the number of days before a database backup series is expired. To ensure that the database can be returned to an earlier level and database references to files in the copy storage pool are still valid, specify the same value for the REUSEDELAY parameter in your copy storage pool definition.

   A database backup volume is considered eligible for expiration if all of the following conditions are true:

   - The last volume of the series has exceeded the expiration value specified with SET DRMDBBACKUPEXPIREDAYS. The expiration value specifies the number of days that must elapse since the volume was used by database backup.
   - The volume's state is VAULT.
   - The volume is not part of the most recent series (DRM will not expire the most recent database backup series).

   The following example sets the number of days to 30.

```
set drmdbbackupexpiredays 30
```

2. When a backup volume is reclaimed and the ADSM status for a copy storage pool volume is EMPTY or the database backup series is EXPIRED, the volume should be moved back onsite for reuse or disposal. To determine which volumes to retrieve, issue the following command:

```
query drmedia * wherestate=vaultretrieve
```

3. After you request the reclaimed volumes be moved back onsite, and the vault location acknowledges that the volumes have been given to the courier, issue the following command:

```
move drmedia * wherestate=vaultretrieve
```

This command automatically completes the following process for all volumes with a current state of VAULTRETRIEVE:

- The state of the volume is changed to COURIERRETRIEVE.
- The location of the volume is updated according to what is specified in the SET DRMCOURIERNAME command. For more information, see “Specify the Courier Name” on page 413.

4. When the courier delivers the volumes, issue the following command to acknowledge that the courier has returned the volumes onsite:

```
move drmedia * wherestate=courierretrieve
```

This command automatically completes the following process for all volumes with a current state of COURIERRETRIEVE:

- The volumes are now onsite and can be reused or disposed.
- The database backup volumes are deleted from the volume history table.
- For scratch copy storage pool volumes, the record in the ADSM database is deleted. For private copy storage pool volumes, the access is updated to read/write.

5. If you do not want to step through all the states, you can use the TOSTATE parameter on the MOVE DRMEDIA command to specify the destination state. For example, to transition the volumes from VAULTRETRIEVE state to ONSITERETRIEVE state, issue the following command:

```
move drmedia * wherestate=vaultretrieve tostate=onsiteretrieve
```

This command automatically completes the following process for all volumes with a current state of VAULTRETRIEVE:

- The volumes are now onsite and can be reused or disposed.
- The database backup volumes are deleted from the volume history table.
- For scratch copy storage pool volumes, the record in the ADSM database is deleted. For private copy storage pool volumes, the access is updated to read/write.

For an example scenario that demonstrates moving volumes back onsite, see “Example: Routine Operations Using Disaster Recovery Manager” on page 396.

Creating the ADSM Server Disaster Recovery Plan File

When the system administrator invokes the PREPARE command, DRM automatically queries the ADSM server for required information to generate a disaster recovery plan file.

To create a disaster recovery plan file, issue the PREPARE command.

In the following example, the PREPARE command is issued with the PLANPREFIX parameter to generate the recovery plan file in directory /u/server/recoveryplans/:

```
prepare planprefix=/u/server/recoveryplans/
```

The plan file name always includes the date and time (yyyymmdd.hhmmss) when the PREPARE command is issued. For example:

```
/u/server/recoveryplans/19950925.120532
```

For details about specifying the location of the disaster recovery plan file, see "Prefix for Recovery Plan File" on page 412, and also refer to the PREPARE command in the *ADSM Administrator's Reference*.

DRM creates one copy of the disaster recovery plan file. It is recommended that you create multiple copies of your disaster recovery plan for safekeeping. For example, keep copies in print, on diskettes, or on NFS-mounted disk space that is physically located offsite.

The PREPARE command should be issued or scheduled to run after back up of your storage pools and database, and the volumes have been marked offsite. This ensures that your disaster recovery plan file is kept up-to-date.

Each time the PREPARE command generates a new disaster recovery plan file, the previous file is not deleted. It is recommended that you periodically delete downlevel recovery plan files.

About the Disaster Recovery Plan File

The disaster recovery plan file contains the information required for recovery of an ADSM server to the point in time represented by the last database backup operation that is completed before the PREPARE command is issued.

The recovery information is organized into stanzas within the disaster recovery plan file. Each stanza in the recovery plan file has a unique name. These names are listed in Table 15 on page 380.

In the event of a disaster, the administrator can use the recovery plan as a guide to recovering the ADSM server. Optionally, the administrator can use an editor or a locally written procedure (for example, a modified version of the awk script *planexpl.awk.smp* that is shipped with DRM) to break out the recovery plan file stanzas into multiple useful files.

About the Recovery Plan File Stanzas

This section describes the stanzas in the recovery plan file and how to use the stanzas.

These stanza files can be categorized as follows:

Command stanzas
Consist of scripts and ADSM macros. These stanzas can be viewed, printed, updated, or executed as part of the disaster recovery process.

Site-specific instruction stanzas
These stanzas include recovery instructions specific to your site. They can be printed, updated, and used during server recovery.

Server requirements stanzas
These stanzas include the database and recovery log requirements, device and volume requirements, and license information. You can view or print these stanzas.

Configuration file stanzas
Consist of the volume history, device configuration, and server options files.

Machine and recovery media stanzas
These stanzas include machine general information, machine hardware and software characteristics, recovery media information, and machine recovery instructions. They can be printed, updated, and used during server recovery. The information provided in the plan file is solely determined by you in planning for rebuilding the machine where the ADSM server is running. The PREPARE command retrieves the information stored in the ADSM database for the machine that contains the server.

Note: The RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE and RECOVERY.SCRIPT.NORMAL.MODE stanzas contain the scripts that *invoke* the scripts and macros contained in the other stanzas.

The following are descriptions of the stanzas:

- PLANFILE.DESCRIPTION

  Identifies the server for this recovery plan, and the date and time the recovery plan is created.

- PLANFILE.TABLE.OF.CONTENTS

  Provides a list of the stanzas in this recovery plan.

- SERVER.REQUIREMENTS

  Identifies the database and recovery log storage requirements for this server. At the recovery site, you will need a replacement server machine that has enough disk space to install the database and recovery log volumes.

  This stanza also identifies the directory where the server executable resided when the server was started. Note that if the server executable is in a different directory on the replacement machine, then the plan file will need to be edited to account for this change.

Note: Use of links to locate the server executable file is not recommended. If you use links to locate this file, you will have to create the links on the replacement machine or modify the following plan file stanzas:

- RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE
- LOGANDDB.VOLUMES.CREATE
- LOGANDDB.VOLUMES.INSTALL
- PRIMARY.VOLUMES.REPLACEMENT.CREATE

- RECOVERY.INSTRUCTIONS.GENERAL

  Identifies site specific instructions the server administrator has manually edited in the source text file identified by *instructionsprefix*RECOVERY.INSTRUCTIONS.GENERAL. It is recommended that the instructions include the overall recovery strategy, key contact names, overview of key applications backed up by this server and so on.

  Note: *Instructionsprefix* is the prefix portion of the file name; see “Prefix for Recovery Instructions” on page 411.

  For more information on editing the text source file, see “Customizing the Site Specific RECOVERY.INSTRUCTIONS” on page 416.

- RECOVERY.INSTRUCTIONS.OFFSITE

  Identifies site specific instructions the server administrator has manually edited to the source text file identified by *instructionsprefix*RECOVERY.INSTRUCTIONS.OFFSITE. It is recommended that the instructions include the name and location of the offsite vault and how to contact the vault.

  Note: *Instructionsprefix* is the prefix portion of the file name; see “Prefix for Recovery Instructions” on page 411.

  For more information on editing the text source file, see “Customizing the Site Specific RECOVERY.INSTRUCTIONS” on page 416.

- RECOVERY.INSTRUCTIONS.INSTALL

  Identifies site specific instructions the server administrator has manually edited to the source text file identified by *instructionsprefix*RECOVERY.INSTRUCTIONS.INSTALL. It is recommended that the instructions include how to rebuild the base server machine and where backup copies of the system image are located.

  Note: *Instructionsprefix* is the prefix portion of the file name; see “Prefix for Recovery Instructions” on page 411.

  For more information on editing the text source file, see “Customizing the Site Specific RECOVERY.INSTRUCTIONS” on page 416.

- RECOVERY.INSTRUCTIONS.DATABASE

  Identifies site specific instructions the server administrator has manually edited to the source text file identified by *instructionsprefix*RECOVERY.INSTRUCTIONS.DATABASE. Information in this stanza should include how to prepare for the ADSM server database recovery. For example, if the backup device is an 8mm library, you may want to provide instructions on how to initialize or load the backup volumes.

Note: *Instructionsprefix* is the prefix portion of the file name; see “Prefix for Recovery Instructions” on page 411.

For more information on editing the text source file, see “Customizing the Site Specific RECOVERY.INSTRUCTIONS” on page 416.

- RECOVERY.INSTRUCTIONS.STGPOOL

  Identifies site specific instructions the server administrator has manually edited to the source text file identified by *instructionsprefix*RECOVERY.INSTRUCTIONS.STGPOOL. It is recommended that the instructions include what applications are backed up in what copy storage pools.

  Note: *Instructionsprefix* is the prefix portion of the file name; see “Prefix for Recovery Instructions” on page 411.

  For more information on editing the text source file, see “Customizing the Site Specific RECOVERY.INSTRUCTIONS” on page 416.

- RECOVERY.VOLUMES.REQUIRED

  Provides a list of the database backup and copy storage pool volumes required to recover the ADSM server. The location and device class names for the required volumes are also displayed. If you are using the MOVE DRMEDIA command for offsite recovery media management, a blank location field means that the volumes are onsite and available to the ADSM server. This list can be used as the basis of periodic audits for the inventory of volumes at the courier and offsite vault. In the event of a disaster, this list would be used to collect the required volumes before recovery of the server is started.

- RECOVERY.DEVICES.REQUIRED

  Provides details about the devices required to read the backup volumes.

- RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE

  Contains a script with the command required to restore the server database and restart the server. Use it as a guide and execute the commands as needed from a command line, or optionally copy it to a file, modify it and the files it references, and execute the script. At the completion of these steps, client requests for file restores will be satisfied directly from copy storage pool volumes.

  The RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE script will often need modification at the recovery site because of differences between the original and the replacement systems. This script provides the following:

  – Restores the server options, volume history, and device configuration information files.

  – Invokes the scripts contained in the following stanzas:

    LOGANDDB.VOLUMES.CREATE
    LOGANDDB.VOLUMES.INSTALL

Note: You should be aware that any log volumes or database volumes which currently exist and have the same name as those identified within the plan file will be *removed* when this script is executed (see LOGANDDB.VOLUMES.CREATE). In most disaster recovery situations a new machine is acquired and then the required operating system and the ADSM server product are installed. At the time this script is executed to begin the restoration of the ADSM Server, it is not expected that this new machine contains any pre-existing ADSM data in a log volume or database volume. If there are special circumstances where, for some reason (for example, testing), you have created a log volume or a database volume and want to preserve the contents, you then must take some action such as renaming the volume or copying the contents elsewhere before executing this script.

– Invokes the ADSM macros contained in the following stanzas:

  LICENSE.REGISTRATION
  COPYSTGPOOL.VOLUMES.AVAILABLE
  COPYSTGPOOL.VOLUMES.DESTROYED
  PRIMARY.VOLUMES.DESTROYED.

To help understand the operations being performed in this script, see “Backup and Recovery Scenarios” on page 352.

To invoke this script, the following three positional parameters must be specified:

– $1 (the administrator ID)
– $2 (the administrator password)
– $3 (the servername)

For example, to invoke this script using an administrator ID of *don*, password of *mox*, server name of *prodadsm*, enter the following command:

```
planprefix/RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE don mox prodadsm
```

For additional information on planprefix, see “Prefix for Recovery Plan File” on page 412.

- RECOVERY.SCRIPT.NORMAL.MODE

  Contains a script with the commands required to restore the server primary storage pools. Use it as a guide and execute the commands as needed from a command line, or optionally copy it to a file, modify it and the files it references, and execute it.

  At the completion of these steps, client requests for file restores are satisfied from primary storage pool volumes. Clients should also be able to resume file backup, archive, and migration functions.

  This script will often need modification at the recovery site because of differences between the original and the replacement systems.

  This script invokes the script contained in the following stanza:

  PRIMARY.VOLUMES.REPLACEMENT.CREATE

  This script also invokes the ADSM macros contained in stanzas:

PRIMARY.VOLUMES.REPLACEMENT
STGPOOLS.RESTORE

To help understand the operations being performed in this script, see “Backup and Recovery Scenarios” on page 352.

To invoke this script, the following three positional parameters must be specified:

- $1 (the administrator ID)
- $2 (the administrator password)
- $3 (the servername)

For example, to invoke this script using an administrator ID of *don*, password of *mox*, server name of *prodadsm*, enter the following command:

```
planprefix/RECOVERY.SCRIPT.NORMAL.MODE don mox prodadsm
```

For additional information on *planprefix*, see “Prefix for Recovery Plan File” on page 412.

- LOGANDDB.VOLUMES.CREATE

  Contains a script with the command required to recreate the ADSM server database and log volumes that existed before the disaster. You can use it as a guide and execute the commands as needed from a command line, or optionally copy it to a file, modify it, and execute it.

  If you used raw partitions for log or database volumes, you must create these volumes before running the LOGANDDB.VOLUMES.INSTALL script. If you want to use regular files instead of raw partitions, you will have to modify this script to include the appropriate *dsmfmt* commands.

  The PREPARE command assumes that the volume formatting program (dsmfmt) resides in the same directory as the server executable indicated in the stanza SERVER.REQUIREMENTS.

  This script is invoked by the script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

- LOG.VOLUMES

  Contains the names of the log volumes to be installed.

  The contents of this stanza must be placed into a separate file which will be used by LOGANDDB.VOLUMES.INSTALL.

- DB.VOLUMES

  Contains the names of the database volumes to be installed.

  The contents of this stanza must be placed into a separate file which will be used by LOGANDDB.VOLUMES.INSTALL.

- LOGANDDB.VOLUMES.INSTALL

  Contains a script with the commands required to install the ADSM server database and log volumes that existed before the disaster.

This script is invoked by the script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

- LICENSE.REGISTRATION

  Contains an ADSM macro to register your ADSM server licenses.

  This macro is invoked by the script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

- COPYSTGPOOL.VOLUMES.AVAILABLE

  Contains an ADSM macro to mark copy storage pool volumes that were moved offsite as moved back onsite. You can use it as a guide and execute the ADSM administrative commands as needed from a command line or graphical user interface, or optionally copy it to a file, modify it, and execute it.

  This macro is invoked by the script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

  In the event of a disaster, compare the copy storage pool volumes listed in this stanza with the volumes you have obtained from the courier and the offsite vault. If you have not physically obtained all volumes, you should remove the entries for the missing volumes from this stanza.

- COPYSTGPOOL.VOLUMES.DESTROYED

  Contains an ADSM macro to mark copy storage pool volumes as unavailable that were onsite at the time of the disaster. You can use it as a guide and execute the ADSM administrative commands as needed from a command line or graphical user interface, or optionally copy it to a file, modify it, and execute it.

  This macro is invoked by the script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

  In the event of a disaster, compare the copy storage pool volumes listed in this stanza with the volumes that were left onsite. If you have physically obtained any of the volumes, you should remove their entries from this stanza.

- PRIMARY.VOLUMES.DESTROYED

  Contains an ADSM macro to mark primary storage pool volumes as destroyed that were onsite at the time of disaster. You can use it as a guide and execute the ADSM administrative commands as needed from a command line or graphical user interface, or optionally copy it to a file, modify it, and execute it.

  This macro is invoked by the script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

  In the event of a disaster, compare the primary storage pool volumes listed in this stanza with the volumes that were onsite. If you have physically obtained any of the volumes and have determined they are useable, you should remove their entries from here.

- PRIMARY.VOLUMES.REPLACEMENT.CREATE

  Contains a script with the commands required to recreate the ADSM server primary storage pool volumes that existed with device class DISK. You can use it as a guide and execute the commands as needed from a command line, or optionally copy it to a file, modify it, and execute it.

If you used raw partitions for the primary storage pool volumes, you must create these replacement volumes by running the PRIMARY.VOLUMES.REPLACEMENT macro. If you want to use regular files instead of raw partitions, modify this script to include the appropriate *dsmfmt* commands.

The PREPARE command assumes that the volume formatting program (dsmfmt) resides in the same directory as the server executable indicated in the stanza SERVER.REQUIREMENTS.

This script is invoked by the script RECOVERY.SCRIPT.NORMAL.MODE.

The SET DRMPLANVPOSTFIX character is appended to the end of the names of the original volumes listed in this stanza. This appended character serves two alternative strategies:

– Makes it easy to find volume names that require renaming in the stanzas. Before using the volume names, change these names to new names that are valid for the device class and valid on the replacement system.
– Automatically generate a new name that can be used by the replacement server. This strategy requires that a previously planned naming convention take into account the appended post fix character.

Notes:

1. Replacement primary volume names must be different from any other original volume name or replacement name.

2. The ADSM server RESTORE STGPOOL command restores storage pools on a logical basis. This means that there is no one-to-one relationship between an original volume and its replacement.

3. There will be entries for the same volumes in PRIMARY.VOLUMES.REPLACEMENTS.

- PRIMARY.VOLUMES.REPLACEMENT

Contains an ADSM macro to define primary storage pool volumes to the ADSM server. You can use it as a guide and execute the ADSM administrative commands as needed from a command line or graphical user interface, or optionally copy it to a file, modify it, and execute it.

This macro is invoked by the script RECOVERY.SCRIPT.NORMAL.MODE.

Primary storage pool volumes that get entries in this stanza have at least one of the following three characteristics:

1. Original volume in a storage pool whose device class was DISK.
2. Original volume in a storage pool with MAXSCRATCH=0.
3. Original volume in a storage pool and volume scratch attribute=no.

The SET DRMPLANVPOSTFIX character is appended to the end of the names of the original volumes listed in this stanza. This appended character serves two alternative strategies:

- Makes it easy to find volume names that require renaming in the stanzas. Before using the volume names, change these names to new names that are valid for the device class and on the replacement system.
- Automatically generate a new name that can be used by the replacement server. This strategy requires that a previously planned naming convention take into account the appended post fix character.

  Notes:

  1. Replacement primary volume names must be different from any other original volume name or replacement name.
  2. The ADSM server RESTORE STGPOOL command restores storage pools on a logical basis. This means that there is no one-to-one relationship between an original volume and its replacement.
  3. There could be entries for the same volume in PRIMARY.VOLUMES.REPLACEMENT.CREATE and PRIMARY.VOLUMES.REPLACEMENT if the volume has a device class of DISK.

- STGPOOLS.RESTORE

  Contains an ADSM macro to restore the primary storage pools. You can use it as a guide and execute the ADSM administrative commands as needed from a command line or graphical user interface, or optionally copy it to a file, modify it, and execute it.

  This macro is invoked by the script RECOVERY.SCRIPT.NORMAL.MODE.

- VOLUME.HISTORY.FILE

  Contains a copy of the server volume history information that existed at the time PREPARE was run. The volume history file is very important to server recovery because the DSMSERV RESTORE DB command uses the volume history file to determine what volumes are required to restore the database. It is referenced by the script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

  The following rules are used to determine where the volume history file is placed at restore time:

  - If VOLUMEHISTORY entries are defined in the server options file, PREPARE uses the fully qualified file name associated with the first entry. If the specified file name does not begin with a directory specification (for example, '.' or '/'), PREPARE adds the prefix *volhprefix*.
  - If a VOLUMEHISTORY entry is not defined in the server options file, PREPARE uses the default name *volhprefix* followed by drmvolh.txt. For example, if *volhprefix* is /opt/IBMadsm-s/bin/ the file name used by PREPARE is /opt/IBMadsm-s/bin/drmvolh.txt.

  Note on the volhprefix:

  The *volhprefix* is set based on the following:

    - If the environmental variable DSMSERV_DIR has been defined, it is used as the *volhprefix*.
    - If the environmental variable DSMSERV_DIR has not been defined, the directory where the ADSM server is started from is used as the *volhprefix*.

  If a fully qualified file name was not specified for the VOLUMEHISTORY option in the server options file, PREPARE adds it to the stanza DSMSERV.OPT.FILE.

- DEVICE.CONFIGURATION.FILE

  Contains a copy of the server device configuration information that existed at the time PREPARE was run. The device configuration file is very important to server recovery because the DSMSERV RESTORE DB command uses this file to read the database backup volumes. It is referenced by the script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

  At recovery time the contents of this stanza may need to be modified if, for example, the original site uses an automated tape library and the recovery site does not use an automated tape library. For information about updating the device configuration file, see “Recreating a Device Configuration File” on page 343.

  The following rules are used to determine where the device configuration file is placed at restore time:

  - If DEVCONFIG entries are defined in the server options file, PREPARE uses the fully qualified file name associated with the first entry. If the specified file name does not begin with a directory specification (for example, ‘.’ or ‘/’), PREPARE adds the prefix *devcprefix*.
  - If a DEVCONFIG entry is not defined in the server options file, PREPARE uses the default name *devcprefix* followed by drmdevc.txt. For example, if *devcprefix* is /opt/IBMadsm-s/bin/ the file name used by PREPARE is /opt/IBMadsm-s/bin/drmdevc.txt.

  Note on the devcprefix:

  The *devcprefix* is set based on the following:

    - If the environmental variable DSMSERV_DIR has been defined, it is used as the *devcprefix*.
    - If the environmental variable DSMSERV_DIR has not been defined, the directory where the ADSM server is started from is used as the *devcprefix*.

  If a fully qualified file name was not specified for the DEVCONFIG option in the server options file, PREPARE adds it to the stanza DSMSERV.OPT.FILE.

- DSMSERV.OPT.FILE

  Contains a copy of the server options file used when the server was started. The server options file sets various server operating characteristics.

  This stanza is referenced by the script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

- LICENSE.INFORMATION

  Contains a copy of the latest license audit results and the server license terms.

- MACHINE.GENERAL.INFORMATION

  Provides information for the machine that contains the ADSM server (for example, machine location).

  This stanza is included in the plan file if the machine information was saved in the ADSM server database using the DEFINE MACHINE with the ADSMSERVER parameter set to YES.

- MACHINE.RECOVERY.INSTRUCTIONS

  Provides the recovery instructions for the machine that contains the ADSM server.

  This stanza is included in the plan file if the machine recovery instructions are saved in the ADSM server database.

- MACHINE.RECOVERY.CHARACTERISTICS

  Provides the hardware and software characteristics for the machine that contains the ADSM server.

  This stanza is included in the plan file if the machine characteristics are saved in the ADSM server database.

- MACHINE.RECOVERY.MEDIA

  Provides information about the media (for example, boot media) needed for rebuilding the machine that contains the ADSM server.

  This stanza is included in the plan file if recovery media information is saved in the ADSM server database and it was associated with the machine that contains the ADSM server.

Table 15 lists the recovery plan file stanzas, and indicates what type of administrative processing is required during set up of DRM, during routine operations, and during disaster recovery. The table also indicates whether the stanza contains a macro, a script, or a configuration file.

Table 15 (Page 1 of 3). Administrative Tasks Associated with the Disaster Recovery Plan File

Stanza Name	**Admin. Action During Setup or Periodic Updates**	**Recommended Admin. Action During Routine Processing**	**Admin. Action During Disaster Recovery**
PLANFILE.DESCRIPTION	—	—	—
PLANFILE.TABLE.OF.CONTENTS	—	—	—
SERVER.REQUIREMENTS	—	—	—
RECOVERY.INSTRUCTIONS.GENERAL	Optionally edit source file associated with stanza	—	—
RECOVERY.INSTRUCTIONS.OFFSITE	Optionally edit source file associated with stanza	—	—

Table 15 (Page 2 of 3). Administrative Tasks Associated with the Disaster Recovery Plan File

Stanza Name	Admin. Action During Setup or Periodic Updates	Recommended Admin. Action During Routine Processing	Admin. Action During Disaster Recovery
RECOVERY.INSTRUCTIONS.INSTALL	Optionally edit source file associated with stanza	—	—
RECOVERY.INSTRUCTIONS.DATABASE	Optionally edit source file associated with stanza	—	—
RECOVERY.INSTRUCTIONS.STGPOOL	Optionally edit source file associated with stanza	—	—
RECOVERY.VOLUMES.REQUIRED	—	MOVE DRMEDIA	—
RECOVERY.DEVICES.REQUIRED	—	—	—
RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE script	—	—	Optionally edit/execute
RECOVERY.SCRIPT.NORMAL.MODE script	—	—	Optionally edit/execute
LOGANDDB.VOLUMES.CREATE script	—	—	Optionally edit/execute
LOG.VOLUMES	—	—	Optionally edit/copy
DB.VOLUMES	—	—	Optionally edit/copy
LOGANDDB.VOLUMES.INSTALL script	—	—	Optionally edit/execute
LICENSE.REGISTRATION macro	—	—	Optionally edit/execute
COPYSTGPOOL.VOLUMES.AVAILABLE macro	—	MOVE DRMEDIA	Optionally edit/execute
COPYSTGPOOL.VOLUMES.DESTROYED macro	—	MOVE DRMEDIA	Optionally edit/execute
PRIMARY.VOLUMES.DESTROYED macro	—	—	Optionally edit/execute
PRIMARY.VOLUMES.REPLACEMENT.CREATE script	—	—	Optionally edit/execute
PRIMARY.VOLUMES.REPLACEMENT macro	—	—	Optionally edit/execute
STGPOOLS.RESTORE macro	—	—	Optionally edit/execute
VOLUME.HISTORY.FILE configuration file	—	—	Optionally copy
DEVICE.CONFIGURATION.FILE configuration file	—	—	Optionally edit/copy
DSMSERV.OPT.FILE configuration file	—	—	Optionally edit/copy
LICENSE.INFORMATION	—	—	—
MACHINE.GENERAL.INFORMATION	Optionally issue DEFINE MACHINE ADSMSERVER=YES	—	—

Table 15 (Page 3 of 3). Administrative Tasks Associated with the Disaster Recovery Plan File

Stanza Name	Admin. Action During Setup or Periodic Updates	Recommended Admin. Action During Routine Processing	Admin. Action During Disaster Recovery
MACHINE.RECOVERY.INSTRUCTIONS	Optionally issue INSERT MACHINE RECOVERYINSTRUCTIONS	—	—
MACHINE.RECOVERY.CHARACTERISTICS	Optionally issue INSERT MACHINE CHARACTERISTICS	—	—
MACHINE.RECOVERY.MEDIA	Optionally issue DEFINE RECOVERYMEDIA and DEFINE RECMEDMACHASSOCIATION	—	—

Note: In the column “Admin. Action During Setup or Periodic Updates,” the “—” means that DRM automatically collects this information for the file.

Example of a Disaster Recovery Plan File

The following is an example of a disaster recovery plan file generated by PREPARE.

```
begin PLANFILE.DESCRIPTION

Recovery Plan for ADSM Server ADSM
Created by DRM PREPARE on 02/01/1997 10:20:34
DRM PLANPREFIX /prepare/
ADSM Server for Sun Solaris - Version 2, Release 6, Level x.x/x.x

end PLANFILE.DESCRIPTION

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin PLANFILE.TABLE.OF.CONTENTS

PLANFILE.DESCRIPTION
PLANFILE.TABLE.OF.CONTENTS

Server Recovery Stanzas:
  SERVER.REQUIREMENTS
  RECOVERY.INSTRUCTIONS.GENERAL
  RECOVERY.INSTRUCTIONS.OFFSITE
  RECOVERY.INSTRUCTIONS.INSTALL
  RECOVERY.VOLUMES.REQUIRED
  RECOVERY.DEVICES.REQUIRED
  RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE script
  RECOVERY.SCRIPT.NORMAL.MODE script
  LOGANDDB.VOLUMES.CREATE script
  LOG.VOLUMES
  DB.VOLUMES
  LOGANDDB.VOLUMES.INSTALL script
  LICENSE.REGISTRATION macro
  COPYSTGPOOL.VOLUMES.AVAILABLE macro
  COPYSTGPOOL.VOLUMES.DESTROYED macro
  PRIMARY.VOLUMES.DESTROYED script
  PRIMARY.VOLUMES.REPLACEMENT.CREATE script
  PRIMARY.VOLUMES.REPLACEMENT macro
  STGPOOLS.RESTORE macro
  VOLUME.HISTORY.FILE
  DEVICE.CONFIGURATION.FILE
  DSMSERV.OPT.FILE
  LICENSE.INFORMATION

Machine Description Stanzas:
  MACHINE.GENERAL.INFORMATION
  MACHINE.RECOVERY.INSTRUCTIONS
  MACHINE.CHARACTERISTICS
  MACHINE.RECOVERY.MEDIA.REQUIRED

end PLANFILE.TABLE.OF.CONTENTS

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 1 of 19). Example of a Disaster Recovery Plan File

```
begin SERVER.REQUIREMENTS

Database Requirements Summary:

    Available Space (MB): 20
  Assigned Capacity (MB): 20
        Pct. Utilization: 2.2
Maximum Pct. Utilization: 2.2
        Physical Volumes: 2

Recovery Log Requirements Summary:

    Available Space (MB): 20
  Assigned Capacity (MB): 20
        Pct. Utilization: 4.4
Maximum Pct. Utilization: 4.8
        Physical Volumes: 2

ADSM Server Executable Directory: /opt/adsmserv/bin

end SERVER.REQUIREMENTS

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin RECOVERY.INSTRUCTIONS.GENERAL

 This ADSM server contains the backup and archive data for FileRight Company
 accounts receivable system. It also is used by various end users in the
 finance and materials distribution organizations.
 The storage administrator in charge of this server is Jane Doe 004-001-0006.
 If a disaster is declared, here is the outline of steps that must be completed.
 1. Determine the recovery site. Our alternate recovery site vendor is IBM
    BRS in Tampa, Fl, USA 213-000-0007.
 2. Get the list of required recovery volumes from this recovery plan file
    and contact our offsite vault so that they can start pulling the
    volumes for transfer to the recovery site.
 3. etc...

end RECOVERY.INSTRUCTIONS.GENERAL

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 2 of 19). Example of a Disaster Recovery Plan File

```
begin RECOVERY.INSTRUCTIONS.OFFSITE

 Our offsite vaulting vendor is OffsiteVault Inc.
 Their telephone number is 514-555-2341. Our account rep is Joe Smith.
 Our account number is 1239992. Their address is ...
 Here is a map to their warehouse ...
 Our courier is ...

end RECOVERY.INSTRUCTIONS.OFFSITE

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin RECOVERY.INSTRUCTIONS.INSTALL

 The base ADSM server system is SunOS 5.3 running on a
 SPARCstation 10.  The SunOS 5.3 operating system
 and ADSM product installation media is stored at the vault.
 There is also a copy in bldg 24 room 4 cabinet a.  The system
 administrator responsible for the SunOS 5.3 and ADSM installation
 is Fred Myers.


end RECOVERY.INSTRUCTIONS.INSTALL

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 3 of 19). Example of a Disaster Recovery Plan File

```
begin RECOVERY.VOLUMES.REQUIRED

Volumes required for data base restore

 Location = OffsiteVault Inc.
  Device Class = LIB8MM
  Volume Name =
   TPBK06
   TPBK08

Volumes required for storage pool restore

 Location = OffsiteVault Inc.
  Copy Storage Pool = CSTORAGEPF
  Device Class = LIB8MM
  Volume Name =
   TPBK05
   TPBK07

end RECOVERY.VOLUMES.REQUIRED

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin RECOVERY.DEVICES.REQUIRED

 Purpose: Description of the devices required to read the
          volumes listed in the recovery volumes required stanza.

             Device Class Name: LIB8MM
        Device Access Strategy: Sequential
            Storage Pool Count: 2
                   Device Type: 8MM
                        Format: DRIVE
         Est/Max Capacity (MB): 4.0
                   Mount Limit: 2
              Mount Wait (min): 60
         Mount Retention (min): 10
                  Label Prefix: ADSM
                       Library: RLLIB
                     Directory:
Last Update by (administrator): BOB
         Last Update Date/Time: 08/11/1995 10:18:34

end RECOVERY.DEVICES.REQUIRED

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 4 of 19). Example of a Disaster Recovery Plan File

```
begin RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE script

#!/bin/ksh
set -x

 # Purpose: This script contains the steps required to recover the server
 #   to the point where client restore requests can be satisfied
 #   directly from available copy storage pool volumes.
 # Note: This script assumes that all volumes necessary for the restore have
 #   been retrieved from the vault and are available. This script assumes
 #   the recovery  environment is compatible (essentially the same) as the
 #   original.  Any deviations require modification to this script and the
 #   macros and shell scripts it runs.  Alternatively, you can use this
 #   script as a guide, and manually execute each step.

if
[ -z "$1" -o -z "$2" -o -z "$3" ]
then
  print "Specify the following positional parameters:"
  print "administrative client ID, password, and server ID."
  print "Script stopped."
  exit
fi

 # Set the ADSM server working directory
cd /opt/IBMadsm-s/bin/

 # Restore server options, volume history, device configuration files.
cp /prepare/DSMSERV.OPT.FILE \
    /opt/IBMadsm-s/bin/dsmserv.optx
cp /prepare/VOLUME.HISTORY.FILE \
    /opt/IBMadsm-s/bin/volhistory.txtx
cp /prepare/DEVICE.CONFIGURATION.FILE \
    /opt/IBMadsm-s/bin/devconfig.txtx

export DSMSERV_CONFIG=/opt/IBMadsm-s/bin/dsmserv.optx

export DSMSERV_DIR=/opt/adsmserv/bin

 # Create and format log and database files.
/prepare/LOGANDDB.VOLUMES.CREATE 2>&1 \
| tee /prepare/LOGANDDB.VOLUMES.CREATE.log
```

Figure 60 (Part 5 of 19). Example of a Disaster Recovery Plan File

```
 # Install the log and database files.
/prepare/LOGANDDB.VOLUMES.INSTALL 2>&1 \
| tee /prepare/LOGANDDB.VOLUMES.INSTALL.log

 # Restore the ADSM server database to latest version backed up per the
 # volume history file.
/opt/IBMadsm-s/bin/dsmserv restore db todate=08/11/1995 totime=10:20:22

 # Start the server.
nohup /opt/IBMadsm-s/bin/dsmserv &
print Please start new ADSM server console with command dsmadmc -CONSOLE.
print Press enter to continue recovery script execution.
read pause

 # Register ADSM Server Licenses
dsmadmc -id=%1 -pass=%2 -serv=%3 -ITEMCOMMIT \
 -OUTFILE=/prepare/license.registration.log \
    macro /prepare/license.registration.mac

 # Tell ADSM Server these copy storage pool volumes are available for use.
 # Recovery Administrator: Remove from macro any volumes not obtained from vault.
dsmadmc -id=$1 -pass=$2 -serv=$3 -ITEMCOMMIT \
  -OUTFILE=/prepare/COPYSTGPOOL.VOLUMES.AVAILABLE.log \
     macro /prepare/COPYSTGPOOL.VOLUMES.AVAILABLE

 # Volumes in this macro were not marked as 'offsite' at the time
 # PREPARE ran. They were likely destroyed in the disaster.
 # Recovery Administrator: Remove from macro any volumes not destroyed.
dsmadmc -id=$1 -pass=$2 -serv=$3 -ITEMCOMMIT \
  -OUTFILE=/prepare/COPYSTGPOOL.VOLUMES.DESTROYED.log \
     macro /prepare/COPYSTGPOOL.VOLUMES.DESTROYED

 # Mark primary storage pool volumes as ACCESS=DESTROYED.
 # Recovery administrator: Remove from macro any volumes not destroyed.
dsmadmc -id=$1 -pass=$2 -serv=$3 -ITEMCOMMIT \
  –OUTFILE=/prepare/PRIMARY.VOLUMES.DESTROYED.log \
     macro /prepare/PRIMARY.VOLUMES.DESTROYED

end RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE script

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 6 of 19). Example of a Disaster Recovery Plan File

```
begin RECOVERY.SCRIPT.NORMAL.MODE script

#!/bin/ksh
set -x

 # Purpose: This script contains the steps required to recover the server
 #        primary storage pools. This mode allows you to return the
 #        copy storage pool volumes to the vault and to run the
 #        server as normal.
 # Note: This script assumes that all volumes necessary for the restore
 # have been retrieved from the vault and are available. This script
 # assumes the recovery  environment is compatible (essentially the
 # same) as the original. Any deviations require modification to this
 # script and the macros and shell scripts it runs. Alternatively,
 # you can use this script as a guide, and manually execute each step.

if
[ -z "$1" -o -z "$2" -o -z "$3" ]
then
  print "Specify the following positional parameters:"
  print "administrative client ID, password, and server ID."
  print "Script stopped."
  exit
fi

 # Create replacement volumes for primary storage pools that use
 # device class DISK.
 # Recovery administrator: Edit script for your replacement volumes.
/prepare/PRIMARY.VOLUMES.REPLACEMENT.CREATE 2>&1 \
| tee /prepare/PRIMARY.VOLUMES.REPLACEMENT.CREATE.log

 # Define replacement volumes in the primary storage pools. Must
 # have different name than original.
 # Recovery administrator: Edit macro for your replacement volumes.
dsmadmc -id=$1 -pass=$2 -serv=$3 -ITEMCOMMIT \
  -OUTFILE=/prepare/PRIMARY.VOLUMES.REPLACEMENT.log \
     macro /prepare/PRIMARY.VOLUMES.REPLACEMENT

 # Restore the primary storage pools from the copy storage pools.
dsmadmc -id=$1 -pass=$2 -serv=$3 -ITEMCOMMIT \
  -OUTFILE=/prepare/STGPOOLS.RESTORE.log \
     macro /prepare/STGPOOLS.RESTORE
```

Figure 60 (Part 7 of 19). Example of a Disaster Recovery Plan File

```
end RECOVERY.SCRIPT.NORMAL.MODE script

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*


begin LOGANDDB.VOLUMES.CREATE script

#!/bin/ksh
set -x

 # Purpose:  Create log and database volumes.
 # Recovery Administrator: Run this to format ADSM server log and
 #  database volumes.

  print Remove database volume /opt/IBMadsm-s/bin/db01x.
 rm -f /opt/IBMadsm-s/bin/db01x

  print Create ADSM database volume /opt/IBMadsm-s/bin/db01x 12M
 /opt/IBMadsm-s/bin/dsmfmt -m -db /opt/IBMadsm-s/bin/db01x 12M

  print Remove database volume /opt/IBMadsm-s/bin/db02x.
 rm -f /opt/IBMadsm-s/bin/db02x

  print Create ADSM database volume /opt/IBMadsm-s/bin/db02x 8M
 /opt/IBMadsm-s/bin/dsmfmt -m -db /opt/IBMadsm-s/bin/db02x 8

  print Remove log volume /opt/IBMadsm-s/bin/lg01x.
 rm -f /opt/IBMadsm-s/bin/lg01x

  print Create ADSM log volume /opt/IBMadsm-s/bin/lg01x 12M
 /opt/IBMadsm-s/bin/dsmfmt -m -log /opt/IBMadsm-s/bin/lg01x 12M

  print Remove log volume /opt/IBMadsm-s/bin/lg02x.
 rm -f /opt/IBMadsm-s/bin/lg02x

  print Create ADSM log volume /opt/IBMadsm-s/bin/lg02x 8M
 /opt/IBMadsm-s/bin/dsmfmt -m -log /opt/IBMadsm-s/bin/lg02x 8

end LOGANDDB.VOLUMES.CREATE script

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 8 of 19). Example of a Disaster Recovery Plan File

```
begin LOG.VOLUMES

/opt/IBMadsm-s/bin/lg01x
/opt/IBMadsm-s/bin/lg02x

end LOG.VOLUMES

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin DB.VOLUMES

/opt/IBMadsm-s/bin/db01x
/opt/IBMadsm-s/bin/db02x

end DB.VOLUMES

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin LOGANDDB.VOLUMES.INSTALL script

#!/bin/ksh
set -x

 # Purpose: Install the log and database volumes.
 # Recovery Administrator: Run this to initialize an ADSM server.

 /opt/IBMadsm-s/bin/dsmserv install \
   2 FILE:/prepare/LOG.VOLUMES \
   2 FILE:/prepare/DB.VOLUMES

end LOGANDDB.VOLUMES.INSTALL script

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 9 of 19). Example of a Disaster Recovery Plan File

```
begin LICENSE.REGISTRATION macro

 /* Purpose: Register the ADSM Server licenses by specifying the names of the */
 /*   enrollment certificate files necessary to re-create the licenses that   */
 /*   existed in the ADSM server.                                             */
 /* Recovery Administrator: Review licenses and add or delete licenses as     */
 /*   necessary.                                                              */
 register license file(desktop.lic)
 register license file(network.lic)
 register license file(drm.lic)
 register license file(devmod1.lic)
 register license file(devmod2.lic)
 register license file(50client.lic)
 register license file(50client.lic)
 register license file(50client.lic)
 register license file(10client.lic)
 register license file(10client.lic)
 register license file(5client.lic)
 register license file(1client.lic)
 register license file(1client.lic)
 register license file(1client.lic)
 register license file(1client.lic)

end LICENSE.REGISTRATION macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 10 of 19). Example of a Disaster Recovery Plan File

```
begin COPYSTGPOOL.VOLUMES.AVAILABLE macro

 /* Purpose: Mark copy storage pool volumes as available for use in recovery. */
 /* Recovery Administrator: Remove any volumes that have not been obtained    */
 /*   from the vault or are not available for any reason.                     */
 /* Note: It is possible to use the mass update capability of the ADSM        */
 /*   UPDATE command instead of issuing an update for each volume. However,   */
 /*   the 'update by volume' technique used here allows you to select         */
 /*   a subset of volumes to be processed.                                    */

 upd vol TPBK05 acc=READO wherestg=CSTORAGEPF
 upd vol TPBK07 acc=READO wherestg=CSTORAGEPF

end COPYSTGPOOL.VOLUMES.AVAILABLE macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin COPYSTGPOOL.VOLUMES.DESTROYED macro

 /* Purpose: Mark destroyed copy storage pool volumes as unavailable.      */
 /*   Volumes in this macro were not marked as 'offsite' at the time the   */
 /*   PREPARE ran. They were likely destroyed in the disaster.             */
 /* Recovery Administrator: Remove any volumes that were not destroyed.    */


end COPYSTGPOOL.VOLUMES.DESTROYED macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin PRIMARY.VOLUMES.DESTROYED macro

 /* Purpose: Mark primary storage pool volumes as ACCESS=DESTROYED.        */
 /* Recovery administrator: Delete any volumes listed here                 */
 /*   that you do not want to recover.                                     */
 /* Note: It is possible to use the mass update capability of the ADSM     */
 /*   UPDATE command instead of issuing an update for each volume. However */
 /*   the 'update by volume' technique used here allows you to select      */
 /*   a subset of volumes to be marked as destroyed.                       */

 upd vol /opt/IBMadsm-s/bin/bk02 acc=DESTROYED wherestg=BACKUPPOOL
 upd vol /opt/IBMadsm-s/bin/bk01x acc=DESTROYED wherestg=BACKUPPOOL
 upd vol /opt/IBMadsm-s/bin/bk03 acc=DESTROYED wherestg= BACKUPPOOLF
 upd vol BACK4X acc=DESTROYED wherestg=BACKUPPOOLT

end PRIMARY.VOLUMES.DESTROYED macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 11 of 19). Example of a Disaster Recovery Plan File

```
begin PRIMARY.VOLUMES.REPLACEMENT.CREATE script

#!/bin/ksh
set -x

 # Purpose: Create replacement volumes for primary storage pools that
 #   use device class DISK.
 # Recovery administrator: Edit this section for your replacement
 #   volume names. New name must be unique, i.e. different from any
 #   original or other new name.

   print Replace /opt/IBMadsm-s/bin/bk02 DISK 16M in BACKUPPOOL
 /opt/IBMadsm-s/bin/dsmfmt -m -data /opt/IBMadsm-s/bin/bk02@ 16

   print Replace /opt/IBMadsm-s/bin/bk01x DISK 5M in BACKUPPOOL
 /opt/IBMadsm-s/bin/dsmfmt -m -data /opt/IBMadsm-s/bin/bk01x@ 5

end PRIMARY.VOLUMES.REPLACEMENT.CREATE script

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 12 of 19). Example of a Disaster Recovery Plan File

```
begin PRIMARY.VOLUMES.REPLACEMENT macro

 /* Purpose: Define replacement primary storage pool volumes for either:   */
 /*   1. Original volume in a storage pool whose device class was DISK.    */
 /*   2. Original volume in a storage pool with MAXSCRATCH=0.              */
 /*   3. Original volume in a storage pool and volume scratch=no.          */
 /* Recovery administrator: Edit this section for your replacement         */
 /*   volume names. New name must be unique, i.e. different from any       */
 /*   original or other new name.                                          */

   /* Replace /opt/IBMadsm-s/bin/bk02 DISK 16M in BACKUPPOOL */
 def vol BACKUPPOOL /opt/IBMadsm-s/bin/bk02@ acc=READW

   /* Replace /opt/IBMadsm-s/bin/bk01x DISK 5M in BACKUPPOOL */
 def vol BACKUPPOOL /opt/IBMadsm-s/bin/bk01x@ acc=READW

   /* Replace /opt/IBMadsm-s/bin/bk03 FILES 4M in BACKUPPOOLF */
 def vol BACKUPPOOLF /opt/IBMadsm-s/bin/bk03@ acc=READW

   /* Replace BACK4X COOL8MM 0M in BACKUPPOOLT */
 def vol BACKUPPOOLT BACK4X@ acc=READW

end PRIMARY.VOLUMES.REPLACEMENT macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 13 of 19). Example of a Disaster Recovery Plan File

```
begin STGPOOLS.RESTORE macro

/* Purpose: Restore the primary storage pools from copy storage pool(s). */
/* Recovery Administrator: Delete entries for any primary storage pools  */
/*   that you do not want to restore.                                    */

 restore stgp ARCHIVEPOOL
 restore stgp BACKUPPOOL
 restore stgp BACKUPPOOLF
 restore stgp BACKUPPOOLT
 restore stgp SPACEMGPOOL

end STGPOOLS.RESTORE macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 14 of 19). Example of a Disaster Recovery Plan File

```
begin VOLUME.HISTORY.FILE

*****************************************************************************************************
*
*               IBM AdStar Distributed Storage Manager Sequential Volume Usage History
*                                     Updated 08/11/1995 10:20:34
*
*     Operation         Volume   Backup Backup Volume Device          Volume
*     Date/Time          Type    Series Oper.   Seq   Class Name      Name
*****************************************************************************************************
 1995/08/11 10:18:43  STGNEW          0      0      0 COOL8MM         BACK4X
 1995/08/11 10:18:43  STGNEW          0      0      0 FILES           /opt/IBMadsm-s/bin/bk03
* Location for volume TPBK05 is: 'OffsiteVault Inc.'
 1995/08/11 10:18:46  STGNEW          0      0      0 LIB8MM          TPBK05
* Location for volume TPBK06 is: 'OffsiteVault Inc.'
 1995/08/11 10:19:23  BACKUPFULL      1      0      1 LIB8MM          TPBK06
* Location for volume TPBK07 is: 'OffsiteVault Inc.'
 1995/08/11 10:20:03  STGNEW          0      0      0 LIB8MM          TPBK07
* Location for volume TPBK08 is: 'OffsiteVault Inc.'
 1995/08/11 10:20:22  BACKUPINCR      1      1      1 LIB8MM          TPBK08

end VOLUME.HISTORY.FILE

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin DEVICE.CONFIGURATION.FILE

/* IBM AdStar Distributed Storage Manager Device Configuration */
DEFINE DEVCLASS COOL8MM DEVTYPE=8MM FORMAT=DRIVE MOUNTLIMIT=1 MOUNTWAIT=60 MOUNTRETENTION=60 PREFIX=ADSM LIBRARY=ITSML
DEFINE DEVCLASS FILES DEVTYPE=FILE MAXCAPACITY=4096K MOUNTLIMIT=2 DIRECTORY=/opt/IBMadsm-s/bin/
DEFINE DEVCLASS FILESSM DEVTYPE=FILE MAXCAPACITY=100K MOUNTLIMIT=2 DIRECTORY=/opt/IBMadsm-s/bin/
DEFINE DEVCLASS LIB8MM DEVTYPE=8MM FORMAT=DRIVE MOUNTLIMIT=1 MOUNTWAIT=60 MOUNTRETENTION=60 PREFIX=ADSM LIBRARY=RLLIB

end DEVICE.CONFIGURATION.FILE

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin DSMSERV.OPT.FILE

* Server options file located in /opt/IBMadsm-s/bin/dsmserv.optx
TCPPort 1509
VOLUMEHISTORY /opt/IBMadsm-s/bin/volhistory.txtx
DEVCONFIG     /opt/IBMadsm-s/bin/devconfig.txtx

end DSMSERV.OPT.FILE

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 15 of 19). Example of a Disaster Recovery Plan File

```
begin LICENSE.INFORMATION

                                   Last License Audit: 08/21/1996 08:10:46
                              Registered Client Nodes: 102
                                Licensed Client Nodes: 180
                     Are network connections in use ?: Yes
                   Are network connections licensed ?: Yes
                        Are UNIX clients registered ?: No
                          Are UNIX clients licensed ?: No
                     Are desktop clients registered ?: Yes
                       Are desktop clients licensed ?: Yes
                       Are OEMVS clients registered ?: No
                         Are OEMVS clients licensed ?: No
           Is space management in use on the server ?: No
         Is space management licensed on the server ?: No
  Is disaster recovery manager in use on the server ?: Yes
Is disaster recovery manager licensed on the server ?: Yes
                   Device support module 1 licensed ?: Yes
                   Device support module 2 licensed ?: Yes
                   Device support module 3 licensed ?: No
                   Device support module 4 licensed ?: No
                            Server License Compliance: Valid

end LICENSE.INFORMATION

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

Figure 60 (Part 16 of 19). Example of a Disaster Recovery Plan File

```
begin MACHINE.GENERAL.INFORMATION

 Purpose: General information for machine ADSMSRV1.
          This is the machine that contains ADSM server ADSM.

         Machine Name: ADSMSRV1
     Machine Priority: 1
             Building: 21
                Floor: 2
                 Room: 2749
          Description: ADSM Server for Branch 51
  Recovery Media Name: ADSMSRVIMAGE

end MACHINE.GENERAL.INFORMATION

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-


begin MACHINE.RECOVERY.INSTRUCTIONS

 Purpose: Recovery instructions for machine ADSMSRV1.

Primary Contact:
   Jane Smith (wk 520-000-0000 hm 520-001-0001)
Secondary Contact:
   John Adams (wk 520-000-0001 hm 520-002-0002)

end MACHINE.RECOVERY.INSTRUCTIONS

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-
```

Figure 60 (Part 17 of 19). Example of a Disaster Recovery Plan File

```
begin MACHINE.CHARACTERISTICS

Purpose: Hardware and software characteristics of machine ADSMSRV1.

  Filesystem            kbytes    used   avail capacity  Mounted on
  /dev/dsk/c0t3d0s0      69191   33036   29245    53%    /
  /dev/dsk/c0t3d0s6     214143  162727   30006    84%    /usr
  /proc                      0       0       0     0%    /proc
  fd                         0       0       0     0%    /dev/fd
  swap                  225520      32  225488     0%    /tmp
  /dev/dsk/c0t3d0s7     288391  237438   22123    91%    /opt

end MACHINE.CHARACTERISTICS

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-
```

Figure 60 (Part 18 of 19). Example of a Disaster Recovery Plan File

```
begin MACHINE.RECOVERY.MEDIA.REQUIRED

 Purpose: Recovery media for machine ADSMSRV1.

  Recovery Media Name: ADSMSRV
                 Type: Other
         Volume Names:
             Location: IRONMNT
          Description: ADSM Server Installation CD
              Product:
  Product Information:

  Recovery Media Name: SUNOPSYS
                 Type: Other
         Volume Names:
             Location: IRONMNT
          Description: SunOS 5.3 Installation CD
              Product:
  Product Information:
end MACHINE.RECOVERY.MEDIA.REQUIRED

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-
```

Figure 60 (Part 19 of 19). Example of a Disaster Recovery Plan File

Example: Routine Operations Using Disaster Recovery Manager

The following scenario demonstrates the use of the PREPARE command and the disaster recovery plan for disaster recovery preparation.

1. ADSM server ADSMSERV contains the backups for the FileRight Company accounts receivable application.

   For *availability management*, FileRight uses ADSM server database mirroring and copy storage pools whose volumes are kept onsite.

   For *disaster recovery management*, FileRight uses ADSM server database backup and copy storage pool volumes which are immediately moved offsite after creation.

2. To prepare for a possible disaster, the administrator records the following site specific recovery steps in the RECOVERY.INSTRUCTIONS stanza source files:

    - Software license numbers
    - Sources of replacement hardware
    - The FileRight Company's specific recovery steps

3. ADSM storage pool backup and database backup processing is performed nightly using ADSM central scheduling.

4. At 8 a.m. the ADSM tape administrator issues the following command to create a list of volumes to be given to the courier:

```
query drmedia * wherestate=mountable
```

    These volumes were created last night by the ADSM server storage pool backup and database backup processing.

5. To check the volumes out of the library, the administrator issues the following command:

```
move drmedia * wherestate=mountable
```

    This command ejects the volumes from the library and marks them unavailable to ADSM.

6. The administrator then packages the volumes so that they can be given to the courier.

7. The courier arrives to pick up today's backup tapes. The administrator records that the volumes will be given to the courier by issuing the following command:

```
move drmedia * wherestate=notmountable
```

8. To generate a recovery plan file with the latest volume information, the administrator issues the following command:

```
prepare
```

    The administrator copies the recovery plan file to a diskette which will be given to the courier.

9. Several weeks ago during routine ADSM server processing copy storage pool volume CSP01 was reclaimed and its ADSM volume status was changed to PENDING. The volume is physically located at the offsite vault.

10. Last night the PENDING window passed for volume CSP01 and its ADSM volume status is now EMPTY. The volume no longer contains valid backup data and should be brought back onsite for reuse or disposal.

11. To determine if any tapes need to be returned from the vault, the administrator generates a list with the following command:

    ```
    query drmedia * wherestate=vaultretrieve
    ```

    These tapes no longer have valid backup data on them. Volume CSP01 is included in the list.

12. The administrator gives the database backup and copy storage pool tapes, the recovery plan file diskette, and the list of volumes to be returned from the vault to the courier.

13. The courier gives the tapes that were on the previous day's return from the vault list to the administrator. To update the state of these tapes as returned to onsite and to check these tapes into the library, the administrator issues the following command:

    ```
    move drmedia * wherestate=courierretrieve cmdf=/drm/checkin.libvol
         cmd="checkin libvol libauto vol1 status=scratch"
    ```

    After the above MOVE DRMEDIA command is executed, the volume records for the tapes that are in the COURIERRETRIEVE state are deleted from the ADSM database. The MOVE DRMEDIA command also generates the CHECKIN LIBVOL command for each tape processed in the file /drm/checkin.libvol:

    ```
    checkin libvol libauto tape01 status=scratch
    checkin libvol libauto tape02 status=scratch
        .
        .
        .
    ```

    You can run the ADSM MACRO command by specifying /drm/checkin.libvol as the macro name to process the CHECKIN LIBVOL commands:

    ```
    > dsmadmc -id=xxxxx -pa=yyyyyy MACRO /drm/checkin.libvol
    ```

14. The courier drives to the vault with today's database backup and copy storage pool tapes, the recovery plan diskette, and the volumes to return from the vault list.

15. At 4 p.m. the ADSM tape administrator calls the vault and verifies that today's backup database and storage pool tapes arrived and are secure. To set the

location of these volumes to VAULT, the administrator issues the following command:

```
move drmedia * wherestate=courier
```

16. The vault also tells the administrator that the volumes on today's return from the vault list have been given to the courier. The administrator issues the following command:

```
move drmedia * wherestate=vaultretrieve
```

This command changes the status for this volume to COURIERRETRIEVE.

17. Later that week, an audit team from headquarters arrives unannounced and asks the administrator for a copy of the disaster recovery plan for this server. The administrator gives the auditors a copy of the recovery plan file generated two hours earlier with up-to-date information that includes the volumes required for recovery, their location, and the commands required to use them to restore the server. The auditors are impressed by the plan's timeliness and completeness.

Storage of Client Recovery Information

DRM allows you to store recovery information for client machines backed up by the ADSM server.

Task	Required Privilege Class
Defining machine information	System
Associating client nodes with machines	
Defining and tracking machine recovery media	
Associating recovery media with machines	

Defining Machine Information

Machine information is used to store details about the machine on which a client node resides. In the event of a disaster, this information can help you identify what you need to rebuild or restore the replacement machines.

To assist with the recovery of an ADSM client machine, define the following information in the ADSM database:

- Machine location and business priority
- The ADSM client nodes associated with a machine
- Machine characteristics
- Machine recovery instructions

Note: The machine characteristics and machine recovery instructions do not have to be defined during the set up process. You can return to this step later.

1. To store information about the machine that contains the client, issue the DEFINE MACHINE command and specify the client's location and business priority.

   The following example defines machine mach22 in building 021, 2nd floor, in room 2929, and has a priority value of 1:

```
define machine mach22 building=021 floor=2 room=2929 priority=1
```

2. To associate one or more ADSM client nodes with a machine, issue the DEFINE MACHNODEASSOCIATION command.

   During disaster recovery, this association information is used to determine what ADSM client nodes resided on machines that have been destroyed. The file spaces associated with these client nodes should be restored. The following example associates node CAMPBELL with machine mach22:

```
define machnodeassociation mach22 campbell
```

   You can query your machine definitions by issuing the QUERY MACHINE command. For an example, see the query machine output in "Example: Recovering ADSM Clients" on page 407.

3. To insert machine characteristics and recovery instructions into the ADSM database, issue the INSERT MACHINE command. You must insert machine characteristics or recovery instructions line by line; therefore, you may want to create an awk script to do this process for you, see Figure 61 on page 402 for an example.

   The following two examples display how to insert machine characteristics and recovery instructions using a line-by-line method, and using an awk script. You must first use an operating system query command or utility to identify the characteristics for your client machine.

   - **INSERT MACHINE Command from an ADSM Prompt**

     The following shows partial output from a query on an AIX client machine operating system. For our example, we want to save the information from lines 1 and 4 with the INSERT MACHINE command.

```
--1  Host Name: mach22 with 8 MB Memory Card
---    16 MB Memory Card
---
--4  Operating System: AIX Version 3 Release 2
---
---  Hardware Address: 10:00:5x:a8:6a:46
```

     – The following example inserts the text "Host Name: mach22 with 8 MB Memory Card" as line 1 and "Operating System: AIX Version 3 Release 2" as line 2 into the ADSM database for machine mach22.

```
insert machine mach22 1 characteristics="Host Name: mach22 with 8 MB Memory Card"
insert machine mach22 2 characteristics="Operating System: AIX Version 3 Release 2"
```

– To specify recovery instructions for your client machine, use this same command but with the RECOVERYINSTRUCTIONS parameter. Characteristics and recovery instructions cannot be specified on the same command.

```
insert machine mach22 1 -
  recoveryinstructions="Recover this machine for accounts receivable dept."
```

- **INSERT MACHINE Command Using an Awk Script**

  To help automate the insertion of client machine information into the ADSM server database, an example awk script named machchar.awk.smp is shipped with the DRM feature. The following is an example procedure to show how you can write a local procedure to insert machine characteristics.

  – The output from the AIX operating system commands *lsdev, lsvg,* and *df* are written to the file clientinfo.txt. These commands will list the devices, logical volumes by volume group, and file systems.

    The file, clientinfo.txt, is then processed by the awk script, which builds an ADSM macro of INSERT commands (one INSERT command for each line in clientinfo.txt).

    The macro is then executed to load the data into the ADSM database.

    From an AIX prompt, the following commands are issued:

```
echo "devices"                          > clientinfo.txt
lsdev -C | sort -d -f                  >> clientinfo.txt
echo "logical volumes by volume group" >> clientinfo.txt
lsvg -o | lsvg -i -l                   >> clientinfo.txt
echo "file systems"                    >> clientinfo.txt
df                                     >> clientinfo.txt
```

    Figure 61 on page 402 is an example procedure named *machchar* to insert machine characteristics.

    The machchar.awk.smp script is shipped with the DRM feature and is located in the /opt/IBMadsm-s/bin directory.

```
# Read machine characteristics from a file and build ADSM macro commands
# to insert the information into the machine characteristics table.
# Invoke with:
#   nawk -f machchar.awk -v machine=acctrcv filewithinfo
BEGIN {
      print "delete machine "machine" type=characteri"
      }
      {
      print "insert machine "machine" "NR" characteri=\""$0"\""
      }
END   {
      }
```

Figure 61. Example of Awk Script File to Insert Machine Characteristics

The machchar.awk script is then executed from a AIX prompt as follows:

```
nawk -f machchar.awk -v machine=acctrcv clientinfo.txt > clientinfo.mac
```

– To insert the machine characteristics, start an administrative client and execute the macro. For example:

```
> dsmadmc -id=xxx -pw=xxx macro clientinfo.mac
```

You can view your machine characteristics by issuing the QUERY MACHINE command with FORMAT=CHARACTERISTICS parameter.

– To specify recovery instructions for your client machine, you can use this same awk script process but with the RECOVERYINSTRUCTIONS parameter.

Your client recovery information is now saved in the ADSM database.

Defining and Tracking Recovery Media

Use the following commands to save a description of the bootable media required to reinitialize or reinstall an operating system on a client machine, and associate one or more machines with this media. You can also use these commands to associate non-executable media such as application user guides with client machines.

1. Define your boot media needed for recovering one or more machines by issuing the DEFINE RECOVERYMEDIA command. In the following example, the boot recovery media name is tellerwrkstnimage, the volume list includes aix001, aix002, and aix003, for product AIX 4.1. The location of the recovery media is Building 21.

```
define recoverymedia tellerwrkstnimage volumenames=aix001,aix002,aix003

  type=boot product="AIX 4.1" location="Building 21"
```

   This command is usually only needed when a client machine configuration changes. For example, if you install a new level of AIX on the client machine and create a bootable image with **mksysb**, issue the DEFINE RECOVERYMEDIA command to create a new recovery media definition that can be used to track the new mksysb volumes.

To query your recovery media definitions, issue the QUERY RECOVERYMEDIA command with the FORMAT=DETAILED parameter.

2. Use the DEFINE RECMEDMACHASSOCIATION command to associate one or more machines with a recovery media. Before you associate a machine with a recovery media, the specified machine must exist and the recovery media must exist.

   During disaster recovery, this association information can be used to determine what boot media to use in the replacement machines.

   The following example associates machine MACH255 with recovery media tellerwrkstnimage:

```
define recmedmachassociation tellerwrkstnimage mach255
```

3. When the boot media is moved offsite, update the location with the UPDATE RECOVERYMEDIA command.

   The following example updates the location of boot media tellerwrkstnimage to Ironvault:

```
update recoverymedia tellerwrkstnimage location=ironvault
```

In a *recovery* scenario, you may want to have softcopy manuals that are on a CD-ROM. You can define this to DRM with the DEFINE RECOVERYMEDIA command.

The following example defines the AIX 4.1 manuals, a volume identifier of cd0001, and a type of OTHER because this is a manual:

```
define recoverymedia aix41manuals description="AIX 4.1 Bookshelf" -
  volumes=cd0001 type=other
```

Recovering the Server

The following list is a guideline to recovering your ADSM server using DRM's disaster recovery plan file.

- Obtain the latest disaster recovery plan file
- Break out the disaster recovery plan file to view, update, print, or execute as ADSM macros or scripts
- Obtain the backup volumes from the vault

- Locate a suitable replacement machine
- Restore the Sun Solaris operating system and ADSM product to your replacement machine
- Review the RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE RECOVERY.SCRIPT.NORMAL.MODE shell because they are important for restoring the server to a point where clients can be recovered (see “About the Recovery Plan File Stanzas” on page 370).

Example: Recovering the ADSM Server

The following scenario demonstrates how an administrator uses the recovery plan file to recover the ADSM server.

1. A disaster is declared for the building that houses the distributed systems server facility at the FileRight Company. Complete recovery of the ADSM server is required.
2. The recovery administrator views the latest recovery plan file for the ADSM server. The recovery plan file stored all the required recovery information in one place. Following the predefined notes in the RECOVERY.INSTRUCTIONS.GENERAL stanza, the administrator reviews the sequence of steps required to recover the server.
3. Step one is to begin the process of obtaining the backup tapes for the server. Fortunately, the backup tapes were stored offsite.
4. The administrator views the RECOVERY.INSTRUCTIONS.OFFSITE stanza for the name and telephone number of the courier the company uses to move tapes between the data center and the offsite vault.
5. The administrator uses a locally written procedure to break out the recovery plan file stanzas into multiple files. (See “About the Disaster Recovery Plan File” on page 370). These files can be optionally viewed, updated, printed, or executed as ADSM macros or scripts.
6. The administrator prints out the RECOVERY.VOLUMES.REQUIRED file. The printout is handed to the courier who goes to the offsite vault to obtain the backup volumes.
7. In the meantime, the administrator must find a suitable replacement machine. Stanza RECOVERY.DEVICES.REQUIRED specifies the required tape drive type that will be needed to read the backup tapes. Stanza SERVER.REQUIREMENTS summarizes the required amount of disk space.
8. The administrator restores the Sun Solaris operating system on the replacement machine as well as the ADSM server software. The media and its location were specified in the RECOVERY.INSTRUCTIONS.INSTALL stanza (as well as in the MACHINE.RECOVERY.MEDIA.REQUIRED stanza). The administrator ensures the environment is the same as when the disaster recovery plan file was created. The environment includes:
   - the directory structure of the ADSM server executable and disk formatting utility

- the directory structure for ADSM server configuration files, that is, disk log, volume history file, device configuration file and server options file
- the directory structure for database, log, and storage pool volume
- the directory structure as well as the individual files created when the disaster recovery plan file was split into multiple files

9. The administrator reviews the ADSM macros contained in the recovery plan. At the time of the disaster, the courier had not picked up the database and storage pool incremental backup volumes created the previous night. However, they were not destroyed by the water. The administrator removes the entry for the storage pool backup volume from the COPYSTGPOOL.VOLUMES.DESTROYED file.

10. The courier returns with the required volumes. Somehow, the vault could not find one of the copy storage pool volumes. There is not enough time to wait for the vault location to find the lost volume. The administrator removes the entry for that volume from the COPYSTGPOOL.VOLUMES.AVAILABLE file.

11. All of the server's primary volumes were destroyed. The administrator decides there are no changes required to the PRIMARY.VOLUMES script and ADSM macro files.

12. To restore the server database to a point where clients can be recovered, the administrator invokes the RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE script file by entering the script file name at the command prompt.

    Note: Alternatively the administrator could have used the steps in the recovery script as a guide, and manually executed each step.

    The following are the steps executed in this recovery script:

    a. Copy the ADSM server options file from the DSMSERV.OPT file to its original location.

    b. Copy the volume history file required by ADSM DSMSERV RESTORE DB processing from the VOLUME.HISTORY.FILE file to its original location.

       Note: Use this copy of the volume history file unless you have a more recent copy (after the disaster occurred).

    c. Copy the device configuration file required by ADSM DSMSERV RESTORE DB processing from the DEVICE.CONFIGURATION.FILE file to its original location.

    d. Create the ADSM server recovery log and database volumes using DSMFMT.

    e. Issue DSMSERV INSTALL for the recovery log and database files.

    f. Issue the DSMSERV RESTORE DB command.

    g. Start the server.

    h. Register ADSM server licenses.

    i. Mark copy storage pool volumes retrieved from vault as available.

    j. Mark copy storage pool volumes which cannot be obtained as unavailable.

k. Mark primary storage pool volumes as destroyed.

13. The administrator invokes the RECOVERY.SCRIPT.NORMAL.MODE script file to restore the server primary storage pools.

    Note: This action is optional at this time because ADSM can access the copy storage pool volumes directly to restore client data. Using this feature, the administrator can minimize client recovery time because server primary storage pools do not have to be restored first. However, in this scenario, the client machines were not damaged, so the focus of the administrator is to restore full ADSM server operation.

    If client machines are damaged, you may want to delay this action until after all clients are recovered.

    Alternatively, the administrator could have used the steps in the recovery script as a guide, and manually executed each step.

    The steps executed in this recovery script are as follows:

    a. Create replacement primary volumes.

    b. Define the replacement primary volumes to ADSM.

    c. Restore the primary storage pools.

14. The administrator collects the database backup and copy storage pool volumes used in the recovery so that they can be returned to the vault. For these database volumes to be returned to the vault using the routine MOVE DRMEDIA process, the administrator executes the following ADSM administrative commands:

    ```
    update volhist TPBK50 devcl=lib8mm ormstate=mountable
    update volhist TPBK51 devcl=lib8mm ormstate=mountable
    ```

    The copy storage pool volumes used in the recovery already have the correct ORMSTATE.

15. The administrator then runs the BACKUP DB command to back up the newly restored database.

16. The administrator issues the MOVE DRMEDIA * WHERESTATE=MOUNTABLE command to check the volumes out of the library.

17. To create a list of the volumes to be given to the courier, the administrator issues the QUERY DRMEDIA * WHERESTATE=NOTMOUNTABLE.

18. After the administrator packages the volumes and gives them to the courier, the MOVE DRMEDIA * WHERESTATE=NOTMOUNTABLE command is issued.

19. The administrator issues PREPARE and celebrates the successful disaster recovery.

Recovering the Clients

To recover ADSM clients, you must have the following information:

- Client machines that have been defined to ADSM, along with their location and restore priority value.
- The boot recovery media.
- Specific recovery instructions for the machine.
- Hardware requirements for the machine.

Example: Recovering ADSM Clients

The following scenario demonstrates how to use DRM's query commands to guide an administrator through the recovery of ADSM clients.

1. A week after the ADSM server was recovered, another water pipe burst in the building that houses distributed systems applications. Many machines that were backed up using ADSM clients are destroyed. A disaster is declared.

2. To view a list of client machines that were lost in building 21 and their restore priority, the administrator issues the following command:

```
query machine building=021 format=detailed
```

   ADSM displays information similar to the following:

```
          Machine Name: POLARIS
      Machine Priority: 1
              Building: 21
                 Floor: 2
                  Room: 1
          ADSM Server?: No
           Description: Payroll
             Node Name: POLARIS
   Recovery Media Name: MKSYSB1
      Characteristics?: Yes
Recovery Instructions?: Yes
```

3. For *each* machine, the administrator issues the following commands:

   a. To determine the location of the boot media, the administrator issues the QUERY RECOVERYMEDIA command. For example:

```
query recoverymedia mksysb1
```

ADSM displays the following information:

```
Recovery Media Name  Volume Names    Location    Machine Name
-------------------- -----------     ----------  ----------------
MKSYSB1              vol1 vol2       IRONVAULT    POLARIS
                      vol3
```

b. To determine the machine specific recovery instructions for the POLARIS machine, the administrator issues:

```
query machine polaris format=recoveryinstructions
```

ADSM displays the following:

```
Recovery Instructions for Polaris.
Primary Contact:
   Jane Smith (wk 520-000-0000 hm 520-001-0001)
Secondary Contact:
   John Adams (wk 520-000-0001 hm 520-002-0002)
```

c. To determine the machine hardware requirements for POLARIS, the administrator issues:

```
query machine polaris format=characteristics
```

ADSM displays information similar to the following:

```
devices
aio0         Defined             Asynchronous I/O
bus0         Available 00-00       Microchannel Bus
fd0          Available 00-00-0D-00  Diskette Drive
fda0         Available 00-00-0D     Standard I/O Diskette Adapter
fpa0         Available 00-00       Floating Point Processor
gda0         Available 00-04       Color Graphics Display Adapter
hd1          Defined             Logical volume
hd2          Defined             Logical volume
hd3          Defined             Logical volume
hdisk0       Available 00-01-00-00  400 MB SCSI Disk Drive
hdisk1       Available 00-01-00-40  Other SCSI Disk Drive
hft0         Available             High Function Terminal Subsystem
inet0        Available             Internet Network Extension
ioplanar0    Available 00-00       I/O Planar
kbd0         Defined   00-00-0K-00  United States keyboard
lb0          Available 00-02-00-20  ADSM Library
lo0          Available             Loopback Network Interface
loglv00      Defined             Logical volume
lp0          Available 00-00-0P-00  IBM 4201 Model 3 Proprinter III
lv03         Defined             Logical volume
lv04         Defined             Logical volume
lvdd         Available             N/A
mem0         Available 00-0B       8 MB Memory Card
mem1         Available 00-0C       16 MB Memory Card
mous0        Defined   00-00-0M-00  3 button mouse
mt0          Available 00-02-00-40  ADSM Tape Drive
ppa0         Available 00-00-0P     Standard I/O Parallel Port Adapter
pty0         Available             Asynchronous Pseudo-Terminal
rootvg       Defined             Volume group
sa0          Available 00-00-S1     Standard I/O Serial Port 1
sa1          Available 00-00-S2     Standard I/O Serial Port 2
scsi0        Available 00-01       SCSI I/O Controller
scsi1        Available 00-02       SCSI I/O Controller
sio0         Available 00-00       Standard I/O Planar
siokb0       Available 00-00-0K     Keyboard Adapter
sioms0       Available 00-00-0M     Mouse Adapter
siotb0       Available 00-00-0T     Tablet Adapter
sys0         Available 00-00       System Object
sysplanar0   Available 00-00       CPU Planar
sysunit0     Available 00-00       System Unit
tok0         Available 00-03       Token-Ring High-Performance Adapter
tr0          Available             Token Ring Network Interface
tty0         Available 00-00-S1-00  Asynchronous Terminal
tty1         Available 00-00-S2-00  Asynchronous Terminal
usrvice      Defined             Logical volume
veggie2      Defined             Volume group
logical volumes by volume group
veggie2:
LV NAME             TYPE       LPs   PPs  PVs  LV STATE      MOUNT POINT
hd2                 jfs        103   103  1    open/syncd    /usr
hd1                 jfs        1     1    1    open/syncd    /home
hd3                 jfs        3     3    1    open/syncd    /tmp
hd9var              jfs        1     1    1    open/syncd    /var
file systems
Filesystem    Total KB    free %used   iused %iused Mounted on
/dev/hd4          8192     420   94%     909    44% /
/dev/hd9var       4096    2972   27%      87     8% /var
/dev/hd2        421888   10964   97%   17435    16% /usr
/dev/hd3         12288   11588    5%      49     1% /tmp
/dev/hd1          4096    3896    4%      26     2% /home
```

d. With the necessary recovery information now available, the administrator successfully restores each client machine.

Customizing Disaster Recovery Manager

You can customize DRM with SET commands to specify the management of the following:

- Storage pools
- Path name prefixes where the recovery plan instructions and disaster recovery plan files should reside
- Replacement volume identifier
- Offsite recovery media

You can also customize site specific recovery instructions. The site specific recovery instructions are flat files that are manually edited using predetermined file names (for example, RECOVERY.INSTRUCTIONS.GENERAL). The site specific recovery instructions are used by the PREPARE command when a new disaster recovery plan is generated.

Task	Required Privilege Class
Specify copy storage pools to be managed	System
Specify primary storage pools to be managed	
Specify the character ID for replacement volume names	
Specify the prefix portion of the path name for recovery plan instructions	
Specify the prefix portion of the path name for recovery plan files	

Customizing the Management of Storage Pools, Path Name Prefixes, and Replacement Volume Identifiers

This section describes the SET commands to configure DRM. For more information, refer to *ADSM Administrator's Reference*.

Copy Storage Pools

Issue the SET DRMCOPYSTGPOOL command to specify the copy storage pools to be managed by DRM. Specify the name of the copy storage pools used for backing up the primary storage pools defined with the SET DRMPRIMSTGPOOL command. These copy storage pools will be processed by the PREPARE, MOVE DRMEDIA, and QUERY DRMEDIA commands. You can specify a list of copy storage pool names or a null string (“”) to indicate that all copy storage pools defined to the server are eligible for processing. At installation, all copy storage pools defined to the server are eligible for processing.

Copy storage pools that you may not want DRM to manage can include onsite copy storage pools used for recovery from media failures. The following example specifies that copy storage pools with the pattern-matching expression of COPY are to be managed by DRM:

```
set drmcopystgpool copy*
```

You can override this SET command using the COPYSTGPOOL parameter with the PREPARE command, for more information refer to *ADSM Administrator's Reference*.

Primary Storage Pools

You can specify the primary storage pools that you want to restore. Eligible primary storage pool volumes defined to these storage pools are included in the plan file stanzas generated by the PREPARE command.

Use the SET DRMPRIMSTGPOOL command to specify which primary storage pools should be processed by the PREPARE command. You can specify a list of primary storage pool names or a null string (“”) to indicate that all primary storage pools defined to the server are eligible for processing. At installation, all the primary storage pools defined to the server are eligible for processing.

The following example specifies that primary storage pools PRIM1 and PRIM2 are to be managed by DRM:

```
set drmprimstgpool prim1,prim2
```

You can override this setting using the PRIMSTGPOOL parameter with the PREPARE command, for more information refer to *ADSM Administrator's Reference*.

Character Identification for Replacement Volume Names

Issue the SET DRMPLANVPOSTFIX command to specify one character to be added to the end of the replacement volumes names in the disaster recovery plan. At installation, the character is set to @. After installation, use this command to change the character.

This command can be used to make the replacement primary storage pool volumes easy to find in the recovery plan stanzas, or to automatically generate replacement volume names.

The following example defines the character identification as r:

```
set drmplanvpostfix r
```

Prefix for Recovery Instructions

Issue the SET DRMINSTRPREFIX command to specify the prefix portion of the path name for the recovery instructions source files.

The following example specifies the prefix as /u/recovery/plans/rpp:

```
set drminstrprefix /u/recovery/plans/rpp
```

PREPARE processing will include the information from the following files in the disaster recovery plan file:

```
/u/recovery/plans/rpp.RECOVERY.INSTRUCTIONS.GENERAL
/u/recovery/plans/rpp.RECOVERY.INSTRUCTIONS.OFFSITE
/u/recovery/plans/rpp.RECOVERY.INSTRUCTIONS.INSTALL
/u/recovery/plans/rpp.RECOVERY.INSTRUCTIONS.DATABASE
/u/recovery/plans/rpp.RECOVERY.INSTRUCTIONS.STGPOOL
```

You can override this SET command using the INSTRPREFIX parameter with the PREPARE command, for more information refer to *ADSM Administrator’s Reference*.

Prefix for Recovery Plan File

Issue the SET DRMPLANPREFIX command to specify the prefix portion of the path name for the generated recovery plan file.

This prefix is used by ADSM to identify the location of the recovery plan file. The plan prefix is also used to generate the ADSM macros and script file names included in the RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE and RECOVERY.SCRIPT.NORMAL.MODE.

The following example specifies the prefix as /u/server/recoveryplans/:

```
set drmplanprefix /u/server/recoveryplans/
```

The disaster recovery plan file name created by PREPARE processing will be in the following format:

```
/u/server/recoveryplans/19950603.013030
```

You can override this SET command using the PLANPREFIX parameter with the PREPARE command, for more information refer to *ADSM Administrator’s Reference*.

Customizing the Management of Offsite Recovery Media

This section describes the SET commands to configure DRM with information necessary for offsite recovery media management. For more information, refer to *ADSM Administrator’s Reference*.

Task	Required Privilege Class
Specify the copy storage pools to be managed	System
Specify the name of the not mountable location name	
Specify the name of the courier	
Specify if ADSM should read sequential media labels of volumes checked out with MOVE DRMEDIA	
Specify expiration of a database backup series	
Specify processing of backup volumes	
Specify the name of the vault where volumes are stored	
Specify the file name for containing the executable commands	

Copy Storage Pools

The MOVE DRMEDIA or QUERY DRMEDIA command processes the volumes in the MOUNTABLE state that are associated with the list of copy storage pool names specified by the SET DRMCOPYSTGPOOL command. If the COPYSTGPOOL parameter is specified with the MOVE DRMEDIA or QUERY DRMEDIA command, the copy storage pool names specified with the command will override those specified with the SET DRMCOPYSTGPOOL command.

Specify the Not Mountable Name

Issue the SET DRMNOTMOUNTABLENAME command to specify the location name for storing your media that is ejected from the library and to be shipped to an offsite location. At installation, the name of this location is set to NOTMOUNTABLE. After installation, this command can be used to modify the name of the location. The location name is used by the MOVE DRMEDIA command to set the location of volumes that are transitioned to the NOTMOUNTABLE state.

The following example specifies the location name as Local.

```
set drmnotmountablename Local
```

Specify the Courier Name

Issue the SET DRMCOURIERNAME command to specify the courier name. At installation, the name of the courier is set to COURIER. After installation, this command can be used to modify the name of the courier. The courier name is used by the MOVE DRMEDIA command to set the location of volumes that are changing from the NOTMOUNTABLE state to the COURIER state.

The following example specifies the courier name as Joe's Courier Service:

```
set drmcouriername "Joe's Courier Service"
```

Sequential Media Labels for Checked Out Volumes

Issue the SET DRMCHECKLABEL command to specify whether ADSM should read sequential media labels of volumes checked out with the MOVE DRMEDIA command. At installation, the value is set to YES. After installation, use this command to modify the value.

The following example specifies that DRM should not read the volume labels:

```
set drmchecklabel no
```

Expiration of a Database Series

Issue the SET DRMDBBACKUPEXPIREDAYS command to specify the number of days before a database backup series is expired. At installation, the number of days before expiration is set to 60. After installation, use this command to modify the number of days that must elapse before a database is expired. A volume is considered eligible for expiration if all of the following conditions are true:

- The last volume of the series exceeds the expiration value specified with SET DRMDBBACKUPEXPIREDAYS. The expiration value specifies the number of days that must elapse since the volume was used by database backup.
- The volume's state is VAULT.
- The volume is not part of the most recent series (DRM will not expire the most recent database backup series).

The following example specifies that 30 days should pass before a database is expired:

```
set drmdbbackupexpiredays 30
```

Processing of Backup Volumes

Issue the SET DRMFILEPROCESS command to specify whether the MOVE DRMEDIA or QUERY DRMEDIA commands should process database backup volumes and copy storage pool volumes that are associated with a device class with a DEVTYPE=FILE. This command is useful for testing of the DRM environment. At installation, the value is set to No. After installation, you can modify this value.

```
set drmfileprocess yes
```

Specify the Vault Name

Issue the SET DRMVAULTNAME command to specify the vault name where volumes are stored. At installation, the name of the vault is set to VAULT. After installation, you can modify this value.

The vault name is used by the MOVE DRMEDIA command to set the location of volumes that are in transition from the COURIER state to the VAULT state.

The following example specifies the vault name as Ironvault with a contact name of D. Lastname, at telephone number 1-000-000-0000:

```
set drmvaultname "Ironvault, D. Lastname, 1-000-000-0000"
```

Specify the File Name that Contains Executable Commands

Issue the SET DRMCMDFILENAME command to specify the file name that will contain the executable commands generated by the MOVE DRMEDIA or QUERY DRMEDIA command. At installation, the file name is not set. After installation, you can modify the file name.

The following example specifies the file name as '/drm/orm/exec.cmds':

```
set drmcmdfilename /drm/orm/exec.cmds
```

Querying the Disaster Recovery Manager System Parameters

To query the settings defined for DRM, issue the QUERY DRMSTATUS command. For example:

```
query drmstatus
```

ADSM displays information similar to the following:

```
            Recovery Plan Prefix: /u/recovery/plans/rpp
        Plan Instructions Prefix: /u/recovery/plans/source/
      Replacement Volume Postfix: @
           Primary Storage Pools: PRIM1 PRIM2
              Copy Storage Pools: COPY*
     Not Mountable Location Name: Local
                    Courier Name: Joe’s Courier Service
                 Vault Site Name: Ironvault, D. Lastname, 1-000-000-0000
DB Backup Series Expiration Days: 30 Day(s)
                    Check Label?: Yes
       Process FILE Device Type?: No
               Command File Name: /drm/orm/exec.cmds
```

Customizing the Site Specific RECOVERY.INSTRUCTIONS

The PREPARE command includes site specific recovery instructions as stanzas in the disaster recovery plan.

Using the following file names, you can create and edit files with specific recovery instructions for your site:

- instructionsprefixRECOVERY.INSTRUCTIONS.GENERAL
- instructionsprefixRECOVERY.INSTRUCTIONS.OFFSITE
- instructionsprefixRECOVERY.INSTRUCTIONS.INSTALL
- instructionsprefixRECOVERY.INSTRUCTIONS.DATABASE
- instructionsprefixRECOVERY.INSTRUCTIONS.STGPOOL

When you create and edit these files and then issue the PREPARE command, the information in these files is included in the disaster recovery plan as stanzas. The following examples show sample entries for these files.

instructionsprefixRECOVERY.INSTRUCTIONS.GENERAL

Include information such as administrator names, telephone numbers, location of passwords, and so on.

The following is example text for this file:

```
Recovery Instructions for ADSM Server ACMESRV on system ZEUS.
Joe Smith  (wk 002-000-1111 hm 002-003-0000)is the primary system programmer.
Salley Doe (wk 002-000-1112 hm 002-005-0000) is primary recovery administrator.
Jane Smith (wk 002-000-1113 hm 002-004-0000) is the responsible manager.
Security Considerations:
 Joe Smith has the password for the Admin ID ACMEADM.  If Joe is unavailable,
 you will need to either issue SET AUTHENTICATION OFF or define a new
 administrative user ID at the replacement ADSM server console.
```

instructionsprefixRECOVERY.INSTRUCTIONS.OFFSITE

Include information such as the offsite vault location, courier's name, and telephone numbers.

The following is example text for this file:

```
Our offsite vault location is Ironvault, Safetown, Az.
The phone number is 1-800-000-0008. You need to contact them directly
to authorize release of the tapes to the courier.
Our courier's name is Fred Harvey.  You can contact him at 1-800-444-0000.
Since our vault is so far away, be sure to give the courier a list
of both the database backup and copy storage pool volumes required. Fred
is committed to returning these volumes to us in less than 12 hours.
```

instructionsprefixRECOVERY.INSTRUCTIONS.INSTALL

Include information about how to install the ADSM server and where the installation volumes are located.

The disaster recovery plan file issues commands using the ADSM administrative client, for example, dsmadmc. Ensure the proper path to the administrative client is established prior to executing the scripts RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE and RECOVERY.SCRIPT.NORMAL.MODE. For example, set the shell variable PATH or update the scripts with the appropriate path specification to find the ADSM administrative client. The following is example text for this file:

```
You will need to reinstall the ADSM server and administrative
client after installing the HP-UX operating system.
The install volume for the ADSM server is INS001. If that is lost, you
will need to contact Copy4You Software, at 1-800-000-0000, and obtain
a new copy. Another possibility is the local IBM Branch office at
555-7777.
```

instructionsprefixRECOVERY.INSTRUCTIONS.DATABASE

Include information about how to recover the database along with how much hardware space is needed.

The following is example text for this file:

```
You will need to find replacement disk space for the server database.  We have
an agreement with Joe Replace that in the event of a disaster, he will
provide us with disk space.
```

instructionsprefixRECOVERY.INSTRUCTIONS.STGPOOL

Include information on primary storage pool recovery instructions.

The following is example text for this file:

```
Do not worry about the archive storage pools during this disaster recovery.
Focus on migration and backup storage pools.
The most important storage pool is XYZZZZ.
```

Using an Awk Script to Break Out a Disaster Recovery Plan File

If you want to restore the ADSM server, you can use an awk script procedure or an editor to break out the stanzas in the disaster recovery plan file into individual files as appropriate.

An example procedure, *planexpl.awk.smp*, is shipped with the DRM feature and is located in /opt/IBMadsm-s/bin/ or wherever the server resides. This is an example procedure that you can modify and use for your local installation.

Summarized Example of Disaster Recovery Manager Usage

This section is an example outline to show how you use DRM in normal routine processing and during a disaster recovery procedure.

Setup

1. Enable DRM by registering the license
2. Ensure the device configuration and volume history information files exist
3. Back up your storage pools and database
4. Define site specific ADSM server recovery instructions
5. Describe priority ADSM client machines

Daily Operations

Day 1

- Back up client files
- Back up ADSM server storage pools
- Back up ADSM server database (full backup)
- Determine what backup volumes have been created
- Eject the volumes from the library
- Hand the volumes to the courier
- Generate the disaster recovery plan with PREPARE command

Day 2

- Back up client files
- Back up ADSM server storage pools
- Back up ADSM server database (full backup)
- Move the new backup volumes offsite

- Acknowledge receipt of previously sent volumes at vault (from Day 1)
- Generate the disaster recovery plan with PREPARE command

Day 3

- Automatic storage pool reclamation processing occurs
- Back up ADSM server database (incremental)
- Move the new backup volumes offsite
- Acknowledge receipt of previously sent volumes at vault (from Day 2)
- Give courier a list of empty volumes to be returned from the vault.
- Generate the disaster recovery plan with PREPARE command

Disaster Occurs

Day 4

- The ADSM server machine and the client machines have been destroyed in the disaster.

Disaster Recovery

Day 4 (continued)

1. Restore ADSM server using the latest recovery plan
2. Identify the top priority client node in the building where disaster occurred
3. Restore client machine files from ADSM server copy storage pools
4. Restore ADSM server primary storage pools
5. Move database backup and copy storage pool volumes back to the vault

Daily Operations

Day 5

- Back up client files
- Back up ADSM server storage pools
- Back up ADSM server database (full backup)
- Determine what backup volumes have been created
- Eject the volumes from the library
- Hand the volumes to the courier
- Generate the disaster recovery plan with PREPARE command

ADSM DRM Project Plan

The following checklist will assist you with planning the tasks required for DRM implementation. In this checklist UNIX refers to the following platforms: AIX, HP-UX, and Sun Solaris.

Table 16 (Page 1 of 3). ADSM DRM Project Plan

Activity	Start Date	End Date	Status	Person Resp.	Backup Person
DRM Planning					
Evaluate your disaster recovery requirements • What are the business priorities for recovering your ADSM clients? • Where is the recovery site? • Is the recovery site hot, warm, or cold? • Do the clients have connectivity to recovery server? • Who are the system and ADSM administrators? • Will you need to return to the original site?					
Evaluate your vault requirements • Where are the offsite backups stored? • How does the vault handle the backup media? • How are the backups packaged or processed? • Who provides the courier service?					
Evaluate the current storage pool backup implementation • What primary storage pools are being backed up? • When are the backups performed? • Backup purpose: offsite or onsite • Backup media • Naming conventions for replacement volumes for primary storage pools					
Evaluate the current database backup implementation • When are the backups performed? • Backup purpose: offsite or onsite • Backup media • How many backup series do you want maintained and for how long? Review the copy storage pool REUSEDELAY value and verify that it is the same as the SET DRMDBBACKUPEXPIREDAYS value.					
Determine which primary storage pools are to be managed by DRM					
Determine which copy storage pools are to be managed by DRM • Offsite copy storage pools					
Plan for the Recovery Plan File (RPF) • Where should the RPF be created? – UNIX*,NT: What is the RPF pathname prefix? – MVS: What is the RPF dsname prefix? • How many RPFs should be kept? • How will RPFs be made available at the recovery site? – Print and store offsite – Tape/diskette copy stored offsite – Copy sent/NFS to recovery site					

Table 16 (Page 2 of 3). ADSM DRM Project Plan

Activity	Start Date	End Date	Status	Person Resp.	Backup Person
Determine where you want to create the user-specified recovery instructions • UNIX*,NT: What is the instructions pathname prefix? • MVS: What is the instructions dsname prefix?					
MVS: Evaluate the TMS/DRM interaction options • Will DRM drive the backup volume movement? • Will TMS drive the backup volume movement? • How to notify the other of volume movement? • What TMS policies need to be created?					
Analyze the sequence of steps related to the PREPARE command backup movement • Document the flow of activities and timings – Sending of volumes offsite – Return on empty volumes – PREPARE timing					
DRM Installation					
Receive the ADSM code					
Install the ADSM code					
Get licensed for DRM • REGISTER LICENSE or • Update the server options					
Customize the DRM options The administrator with system authority issues: • SET DRMBACKUPEXPIREDAYS to define the Database backup expiration • SET DRMPRIMSTGPOOL to specify the DRM-managed storage pools • SET DRMCOPYSTGPOOL to specify the DRM-managed copy storage pools • SET DRMPLANVPOSTFIX to specify 1 character to be appended to new storage pools • SET DRMPLANPREFIX to specify the RPF prefix • SET DRMINSTRPREFIX to specify the user instruction file prefix • SET DRMNOTMOUNTABLENAME to specify the default location for media to be sent offsite • SET DRMCOURIERNAME to specify the default courier • SET DRMVAULTNAME to specify the default vault • SET DRMCMDFILENAME to specify the default file name to contain the commands specified with the CMD parameter on MOVE and QUERY DRMEDIA • SET DRMCHECKLABEL (UNIX*,NT only) to specify whether volume labels are verified when checked out by the MOVE DRMEDIA command					

Table 16 (Page 3 of 3). ADSM DRM Project Plan

Activity	Start Date	End Date	Status	Person Resp.	Backup Person
Define the site-specific recovery instructions Identify: • Target disaster recovery server location • Target server software requirements (OS or ADSM) • Target server hardware requirements (storage devices) • ADSM administrator contact • Courier name and telephone number • Vault location and contact person Create: • Enter the site-specific recovery instructions data into files created in the same path/HLQ as specified by SET DRMINSTRPREFIX					
DRM Test					
Test the DRM installation and customization • Q DRMSTATUS to display the DRM setup • Back up the primary storage pools • Back up the ADSM database • Q DRMEDIA to list the backup volumes • MOVE DRMEDIA to move offsite • PREPARE to create the RPF					
Examine the RPF created					
Test the RPF break out • UNIX*: awk script planexpl.awk • NT: REXX exec planexpl.rex • MVS: REXX exec ANRPLANX • Locally written procedure					
DRM Production					
Set up the schedules for automated functions					
Implement the DRM procedures					

Appendix A. Supported Devices and Device Configuration Worksheets

ADSM supports a wide range of devices:

Disk devices
ADSM supports any disk storage device supported by the operating system.

Tape devices
Table 17 and Table 18 on page 426 show the tape drives and libraries supported by ADSM.

Optical devices
Table 17 and Table 18 on page 426 show the optical drives and libraries supported by ADSM.

The devices that ADSM supports may change. For current information about supported devices:

- Check with IBM or your authorized reseller
- Call the IBM Information Support Center at 1-800-IBM-3333 and ask for STAR 20
- Visit the ADSM page on the World Wide Web at this address:

  http://www.storage.ibm.com/adsm

Devices Supported by ADSM

Table 17 shows the drives supported by ADSM, and the formats supported. You can record the device names for tape drives in Table 19 and Table 20 on page 429, for medium changers in Table 21 on page 429, and for optical drives in Table 22 on page 429.

The IBM 3490E, 3570, and 3590 tape drives are supported through the device drivers supplied by IBM with the hardware. Other devices that are listed are supported through the ADSM device driver.

Table 17 (Page 1 of 3). Supported Devices and Formats

Product ID	Supported Formats	Estimated Capacity
	1/2-inch Tape Devices	
IBM 3490E (Models C10, C11, C1A, C22, C2A, E01, E11)	3490B, 3490C	720MB (Note 1 on page 425)
	IBM Magstar Tape Devices	
IBM Magstar MP	3570B, 3570C	5GB (Note 1 on page 425)
IBM 3590	3590B, 3590C	10GB (Note 1 on page 425)
	4mm Tape Devices	
ADIC DATa 8000	DDS1	2.0GB (Note 1 on page 425)

© Copyright IBM Corp. 1995, 1997

Table 17 (Page 2 of 3). Supported Devices and Formats

Product ID	**Supported Formats**	**Estimated Capacity**
ADIC DATa 8008	DDS1, DDS1C	4.0GB (Note 1 on page 425)
ADIC DATa 8008i	DDS1, DDS1C	4.0GB (Note 1 on page 425)
ADIC DATa 8008e	DDS1, DDS1C	4.0GB (Note 1 on page 425)
Hewlett-Packard 35470A	DDS1	2.0GB
Hewlett-Packard 35480A	DDS1, DDS1C	2.0GB (Note 1 on page 425)
Hewlett-Packard C1533A	DDS1, DDS1C, DDS2, DDS2C	4.0GB (Note 1 on page 425)
Hewlett-Packard Jetstore 2000e	DDS1	2.0GB
Hewlett-Packard Jetstore 5000e	DDS1, DDS1C	2.0GB (Note 1 on page 425)
Hewlett-Packard Jetstore 6000e	DDS1, DDS1C, DDS2, DDS2C	4.0GB (Note 1 on page 425)
IBM 7206-005	DDS1, DDS1C, DDS2, DDS2C	4.0GB (Note 1 on page 425)
Sony SDT-5000	DDS1, DDS1C, DDS2, DDS2C	4.0GB (Note 1 on page 425)
Sony SDT-7000	DDS2, DDS2C	4.0GB (Note 1 on page 425)
WangDAT 3300DX	DDS1, DDS2	2.0GB
WangDAT 3400DX	DDS1, DDS1C, DDS2, DDS2C	4.0GB (Note 1 on page 425)
8mm Tape Devices		
Andataco Encore 8205	8200, 8200C	2.3GB (Note 1 on page 425)
Andataco Encore 8505	8200, 8200C, 8500, 8500C	5.0GB (Note 1 on page 425)
Dynatek HSB-2300	8200, 8200C	2.3GB (Note 1 on page 425)
Dynatek HSB-5000	8200, 8200C, 8500, 8500C	5.0GB (Note 1 on page 425)
Dynatek HSB-10.0	8200, 8200C, 8500, 8500C	5.0GB (Note 1 on page 425)
Exabyte EXB-8205	8200, 8200C	2.3GB (Note 1 on page 425)
Exabyte EXB-8205XL	8200, 8200C	3.5GB with XL tape (Note 1 on page 425)
Exabyte EXB-8500	8200, 8500	5.0GB
Exabyte EXB-8500C	8200, 8200C, 8500, 8500C	5.0GB (Note 1 on page 425)
Exabyte EXB-8505	8200, 8200C, 8500, 8500C	5.0GB (Note 1 on page 425)
Exabyte EXB-8505XL	8200, 8200C, 8500, 8500C	7.0GB with XL tape (Note 1 on page 425)
Exabyte EXB-8700LT	8200, 8200C, 8500, 8500C	7.0GB
Exabyte EXB-8900	8500, 8500C, 8900	20.0GB
IBM 7208-011	8200, 8200C, 8500, 8500C	5.0GB (Note 1 on page 425)
Sun 8505XL	8200, 8200C, 8500, 8500C	7.0GB with XL tape (Note 1 on page 425)
Transitional Technologies, Inc. (TTi) CTS-8000H	8500, 8500C	5.0GB (Note 1 on page 425)
TTi CTS-8519H	8200, 8200C, 8500, 8500C	5.0GB and 10.0GB (Note 2 on page 425 and Table 42 on page 446)
DLT Tape Devices		

Table 17 (Page 3 of 3). Supported Devices and Formats

Product ID	Supported Formats	Estimated Capacity
ADIC DS9200	DLT10, DLT10C	10GB (Note 1 on page 425)
ADIC DS9300	DLT15, DLT15C	15GB
ADIC DS9400	DLT10, DLT10C, DLT20, DLT20C	20GB (Note 1 on page 425)
DEC DLT TZ87	DLT10, DLT10C	10GB (Note 1 on page 425)
DEC DLT TZ87n	DLT10, DLT10C	10GB (Note 1 on page 425)
DEC DLT TZ88	DLT10, DLT10C, DLT20, DLT20C	20GB (Notes 1 and 3 on page 425)
DEC DLT TZ88n	DLT10, DLT10C, DLT20, DLT20C	20GB (Notes 1 and 3 on page 425)
Quantum DLT 2000	DLT10, DLT10C	10GB (Note 1 on page 425)
Quantum DLT 2000XT	DLT15, DLT15C	15GB (Notes 1 and 4 on page 425)
Quantum DLT 4000	DLT10, DLT10C, DLT20, DLT20C	20GB (Notes 1 and 3 on page 425)
	Optical Drives	
Hewlett-Packard 1300T	650MB, 1300MB	1.19GB
IBM 0632-C1	650MB	650MB
IBM 0632-C2A	650MB, 1300MB	1.19GB
IBM 0632-C2B	650MB, 1300MB	1.19GB
IBM 0632-CHA	650MB, 1300MB	1.19GB
IBM 0632-CHB	650MB, 1300MB	1.19GB
IBM 3510-001	650MB, 1300MB	1.19GB
IBM 7209-001	650MB	650MB
IBM 7209-002	650MB, 1300MB	1.19GB
IBM 7209-003	1300MB, 2600MB	2.6GB
Quantum DLT 2000	DLT10, DLT10C	10GB
Quantum DLT 4000	DLT10, DLT10C, DLT20, DLT20C	20GB

Notes:

1. Greater capacity may be achieved with compression.
2. ADSM supports only mirrored operations.
3. DLT20 and DLT20C can only be used with CompacTape IV cartridges.
4. DLT15 and DLT15C can only be used with CompacTape III cartridges.

Libraries Supported by the ADSM Device Driver

Table 18 on page 426 shows the supported libraries and references to the worksheets to use with each. Use the worksheets to record SCSI IDs and device names for the devices that you are attaching to your ADSM server system.

Where needed, the worksheets also show the element numbers (addresses) for drives, slots, and robotics in libraries. The element address is a number that indicates a physical location within an automated library. ADSM needs the element address to connect the physical location of a drive to the drive's SCSI address. You need the device names and element numbers when:

- Defining or updating drives that are in a library, when there is more than one drive in the library

- Checking in volumes to a library, when the library does not have an input/output station
- Using a drive in a library to label volumes (DSMLABEL)

Verify the element numbers shown here with documentation that you should have received from the device manufacturer.

The IBM 3494, 7331, and 7336 tape libraries are supported through the device driver supplied by IBM with the hardware. Other devices that are listed are supported through the ADSM device driver.

Table 18 (Page 1 of 4). Supported Libraries and Associated Worksheets

Product ID (Device)	**Worksheet**
ADIC SCALAR 218	Table 32 on page 438
ADIC SCALAR 224/M	Table 32 on page 438
ADIC SCALAR 248	Table 32 on page 438
ADIC SCALAR 424/M	Table 32 on page 438
ADIC SCALAR 448	Table 32 on page 438
ADIC SCALAR 458	Table 32 on page 438
ADIC VLS 4mm	Table 33 on page 439
ADIC 1200D 4mm	Table 34 on page 439
AIWA AL-D210 4mm	Table 46 on page 448
AIWA AL-D220 4mm	Table 47 on page 448
Andataco Encore 10e 8mm	Table 24 on page 431
Andataco Encore 120 8mm	Table 26 on page 433
Andataco Encore 210 8mm	Table 25 on page 432
Andataco Encore 440 8mm	Table 35 on page 440
Andataco Encore 480 8mm	Table 36 on page 441
BoxHill BorgBox 8mm	Table 36 on page 441
BoxHill BreadBox 8mm	Table 24 on page 431
BoxHill CubeBox 8mm	Table 35 on page 440
BoxHill FreezerBox 8mm	Table 23 on page 430
BoxHill IceBox 8mm	Table 26 on page 433
BoxHill LightBox 8mm	Table 25 on page 432
Breece Hill Q7	Table 37 on page 442
Breece Hill Q47	Table 38 on page 443
DEC DLT TZ875	Table 43 on page 446
DEC DLT TZ875n	Table 43 on page 446
DEC DLT TZ877	Table 44 on page 447

Table 18 (Page 2 of 4). Supported Libraries and Associated Worksheets

Product ID (Device)	**Worksheet**
DEC DLT TZ877n	Table 44 on page 447
DEC DLT TZ885	Table 43 on page 446
DEC DLT TZ885n	Table 43 on page 446
DEC DLT TZ887	Table 44 on page 447
DEC DLT TZ887n	Table 44 on page 447
DEC DLT TL810	Table 50 on page 450
DEC DLT TL820	Table 53 on page 453
Diverse Logistics Libra 8 4mm	Table 27 on page 434
Diverse Logistics Libra 16 4mm	Table 28 on page 434
Exabyte EXB-018/218 4mm	Table 48 on page 449
Exabyte EXB-10e 8mm	Table 24 on page 431
Exabyte EXB-10h 8mm	Table 24 on page 431
Exabyte EXB-10i 8mm	Table 24 on page 431
Exabyte EXB-60 8mm	Table 23 on page 430
Exabyte EXB-120 8mm	Table 26 on page 433
Exabyte EXB-210 8mm	Table 25 on page 432
Exabyte EXB-220 8mm	Table 25 on page 432
Exabyte EXB-440 8mm	Table 35 on page 440
Exabyte EXB-480 8mm	Table 36 on page 441
Hewlett-Packard 20T	Table 60 on page 459
Hewlett-Packard 40T	Table 60 on page 459
Hewlett-Packard 120T	Table 61 on page 460
Hewlett-Packard 200T	Table 61 on page 460
Hewlett-Packard C1553A 4mm	Table 39 on page 443
Hewlett-Packard C1561A 4mm	Table 39 on page 443
Hewlett-Packard SureStore 12000e 4mm	Table 39 on page 443
IBM 3494 (3490E and 3590 tape drives)	Table 20 on page 429
IBM 3995-063	Table 59 on page 458
IBM 3995-163	Table 59 on page 458
IBM 3995-C60	Table 59 on page 458
IBM 3995-C62	Table 59 on page 458
IBM 3995-C64	Table 59 on page 458
IBM 3995-C66	Table 59 on page 458
IBM 3995-C68	Table 59 on page 458
IBM 7331-205 8mm	Table 29 on page 435

Table 18 (Page 3 of 4). Supported Libraries and Associated Worksheets

Product ID (Device)	Worksheet
IBM 7336-205 4mm	Table 31 on page 437
Lago LS-380L 8mm	Table 40 on page 444
MountainGate D-28	Table 37 on page 442
MountainGate D-60	Table 37 on page 442
MountainGate D-360	Table 41 on page 445
MountainGate D-480	Table 41 on page 445
Odetics ACL 2/28 DLT	Table 49 on page 449
Odetics ACL 4/52 DLT	Table 50 on page 450
Odetics ACL 6/176 DLT	Table 51 on page 451
Odetics ACL 9/88 DLT	Table 52 on page 452
Odetics ACL 2640 DLT	Table 53 on page 453
OverlandData LXB 2110	Table 54 on page 454
OverlandData LXB 2210	Table 54 on page 454
OverlandData LXB 4110	Table 54 on page 454
OverlandData LXB 4210	Table 54 on page 454
Qualstar TLS 2218/A	Table 57 on page 456
Qualstar TLS 2236/A	Table 57 on page 456
Qualstar TLS 2436	Table 57 on page 456
Qualstar TLS 2472	Table 57 on page 456
Qualstar TLS 24144	Table 57 on page 456
Qualstar TLS 4210	Table 58 on page 457
Qualstar TLS 4220	Table 58 on page 457
Qualstar TLS 4420	Table 58 on page 457
Qualstar TLS 4440	Table 58 on page 457
Qualstar TLS 4480	Table 58 on page 457
Qualstar TLS 4660	Table 58 on page 457
Qualstar TLS 46120	Table 58 on page 457
Quantum DLT 2500	Table 43 on page 446
Quantum DLT 2500XT	Table 45 on page 447
Quantum DLT 2700	Table 44 on page 447
Quantum DLT 2700XT	Table 45 on page 447
Quantum DLT 4500	Table 43 on page 446
Quantum DLT 4700	Table 44 on page 447
Spectralogic 4000/20 4mm	Table 55 on page 455
Spectralogic 4000/40 4mm	Table 55 on page 455

Table 18 (Page 4 of 4). Supported Libraries and Associated Worksheets

Product ID (Device)	Worksheet
Spectralogic 4000/60 4mm	Table 55 on page 455
StorageTek 9704 4mm	Table 56 on page 455
StorageTek 9711 8mm	Table 40 on page 444
Transitional Technologies, Inc. (TTi) Series-10 8mm	Table 24 on page 431
TTi Q7 DLT	Table 37 on page 442
TTi Q47 DLT	Table 38 on page 443

Recording SCSI IDs and Device Names

Use the tables that follow to record SCSI IDs and device names.

Table 19. Manual Tape Drive (4mm, 8mm, or DLT)

Device	SCSI ID	Device Name
Tape drive	__________	/dev/rmt/____mt

Table 20. Tape Drives (IBM 3490E, 3570, or 3590)

Device	SCSI ID	Device Name
Tape drive	__________	/dev/rmt/____st

Table 21. Medium Changer (IBM 3570 or 3590 with ACF)

Device	SCSI ID	Device Name
Medium changer	__________	/dev/rmt/____smc

Table 22. Manual Optical Drive

Device	SCSI ID	Device Name
Optical drive	__________	/dev/rmt/____op

Table 23. BoxHill FreezerBox or Exabyte EXB-60

Device	SCSI ID	Device Name
Tape drive 1 (element 116) **1**	__________	/dev/rmt/____mt
Tape drive 2 (element 117) **2**	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

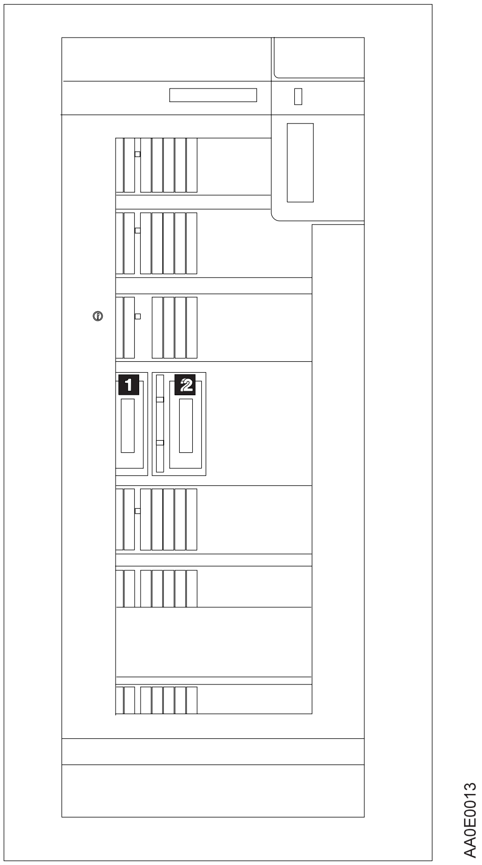


Table 24. Andataco Encore 10e, BoxHill BreadBox, Exabyte EXB-10e, EXB-10h, EXB-10i, or TTi Series-10

Device	**SCSI ID**	**Device Name**
Tape drive (element 0)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

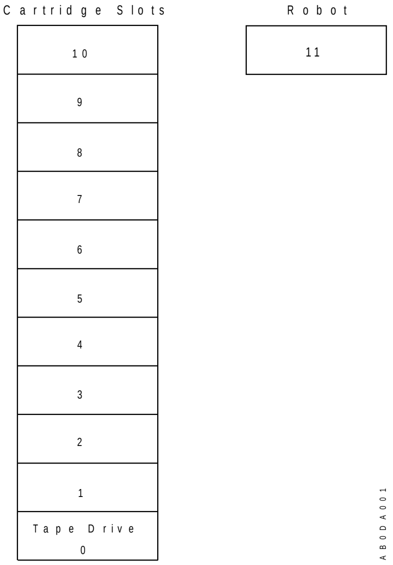


Table 25. Andataco Encore 210, BoxHill LightBox, Exabyte EXB-210, or EXB-220

Device	SCSI ID	Device Name
Tape drive 1 (element 82)	__________	/dev/rmt/____mt
Tape drive 2 (element 83)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

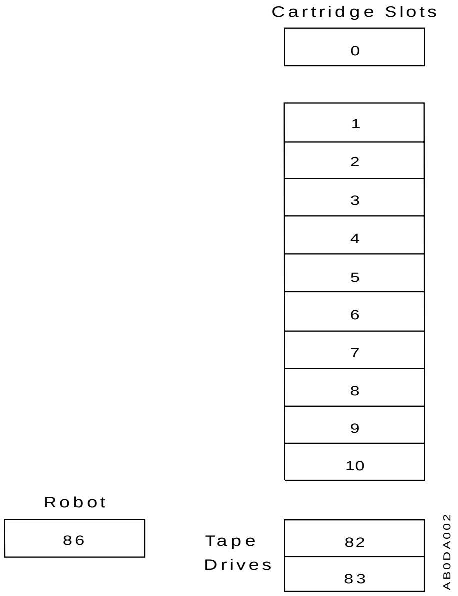


Table 26. Andataco Encore 120, BoxHill IceBox, or Exabyte EXB-120

Device	SCSI ID	Device Name
Tape drive (element 116) **1**	__________	/dev/rmt/____mt
Tape drive (element 117) **2**	__________	/dev/rmt/____mt
Tape drive (element 118) **3**	__________	/dev/rmt/____mt
Tape drive (element 119) **4**	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

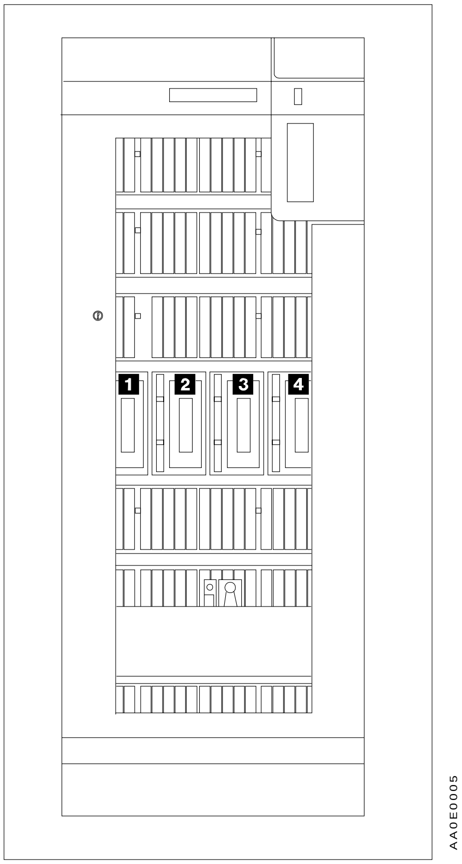


Table 27. Diverse Logistics Libra 8

Device	**SCSI ID**	**Device Name**
Tape drive (element 1)	__________	/dev/rmt/____mt

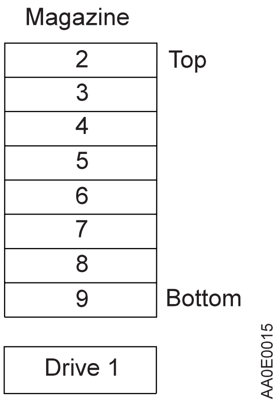


Table 28. Diverse Logistics Libra 16

Device	**SCSI ID**	**Device Name**
Tape drive (element 1)	__________	/dev/rmt/____mt

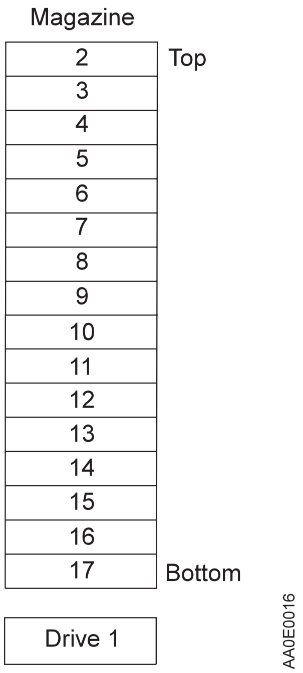


Table 29. IBM 7331-205

Device	**SCSI ID**	**Device Name**
Tape drive 1 (element 23)	__________	/dev/rmt/____mt
Tape drive 2 (element 24)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

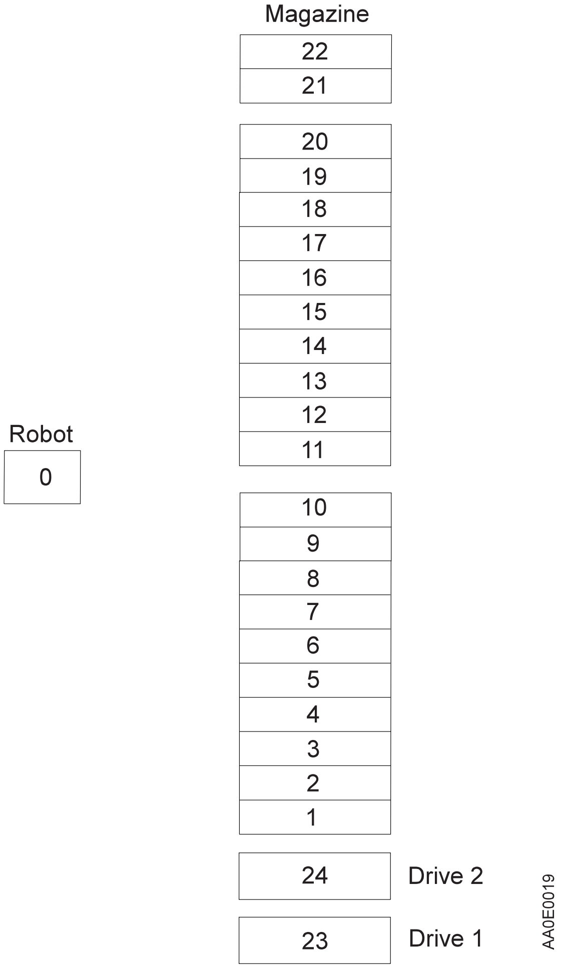


Table 30. IBM 7332-005

Device	SCSI ID	Device Name
Tape drive (element 1)	__________	/dev/rmt/____mt

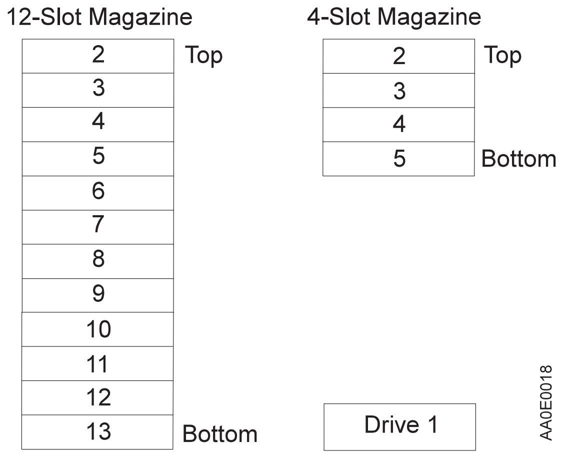


Table 31. IBM 7336-205

Device	SCSI ID	Device Name
Tape drive 1 (element 23)	__________	/dev/rmt/____mt
Tape drive 2 (element 24)	__________	/dev/rmt/____mt
Autochanger (element 0)	__________	/dev/rmt/____lb

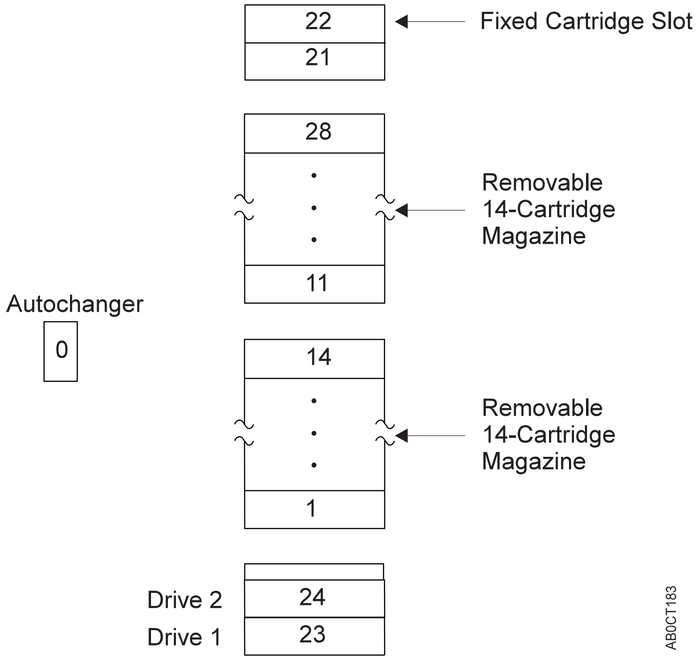
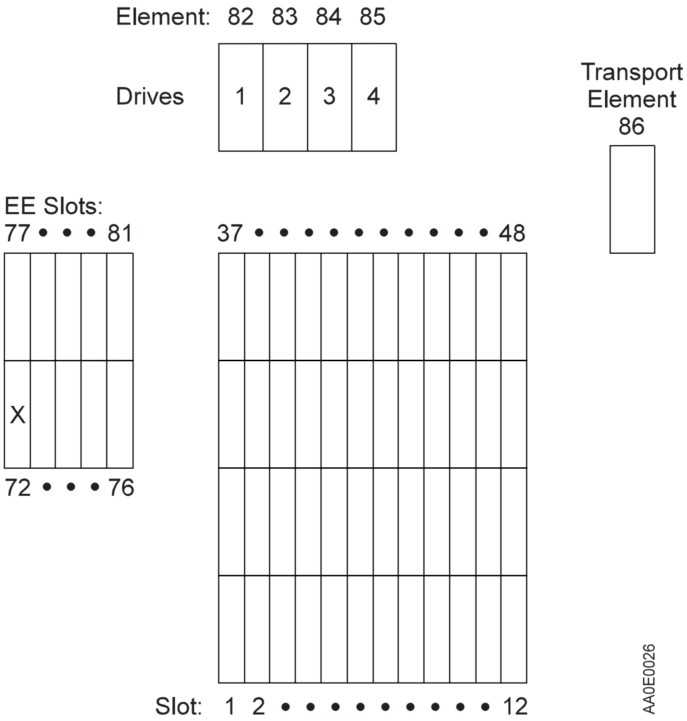


Table 32. ADIC Scalar 218, 224/M, 248, 424/M, 448, or 458

Device	SCSI ID	Device Name
Tape drive 1 (element 82)	__________	/dev/rmt/____mt
Tape drive 2 (element 83)	__________	/dev/rmt/____mt
Tape drive 3 (element 84)	__________	/dev/rmt/____mt
Tape drive 4 (element 85)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb



Note: ADSM currently only supports using one entry/exit port (element address of 72). Only Models 224/M, 424/M, and 458 have mailbox slots.

Table 33. ADIC VLS 4mm

Device	**SCSI ID**	**Device Name**
Tape drive 1 (element 2)	__________	/dev/rmt/____mt
Tape drive 2 (element 3)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

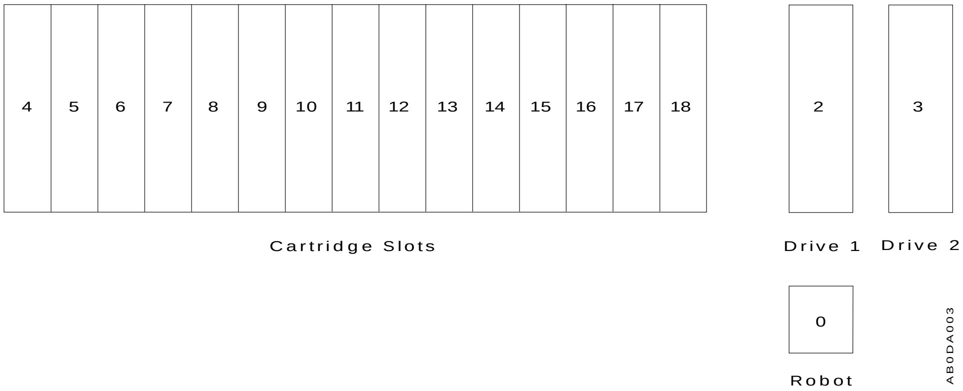


Table 34. ADIC 1200D 4mm

Device	**SCSI ID**	**Device Name**
Tape drive (element 2)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

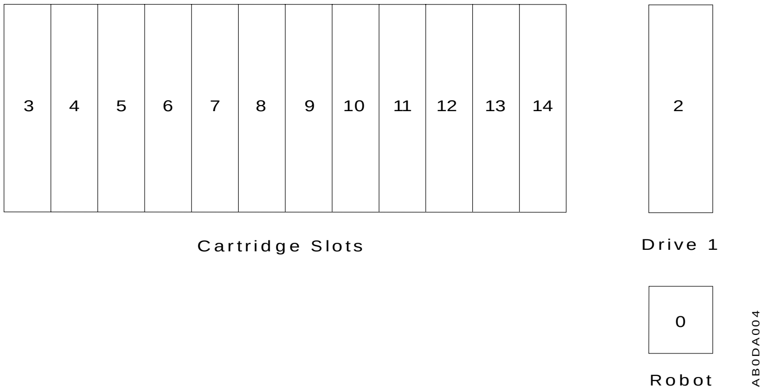


Table 35. Andataco Encore 440, BoxHill CubeBox, or Exabyte EXB-440

Device	SCSI ID	Device Name
Tape drive 1 (element 82)	__________	/dev/rmt/____mt
Tape drive 2 (element 83)	__________	/dev/rmt/____mt
Tape drive 3 (element 84)	__________	/dev/rmt/____mt
Tape drive 4 (element 85)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

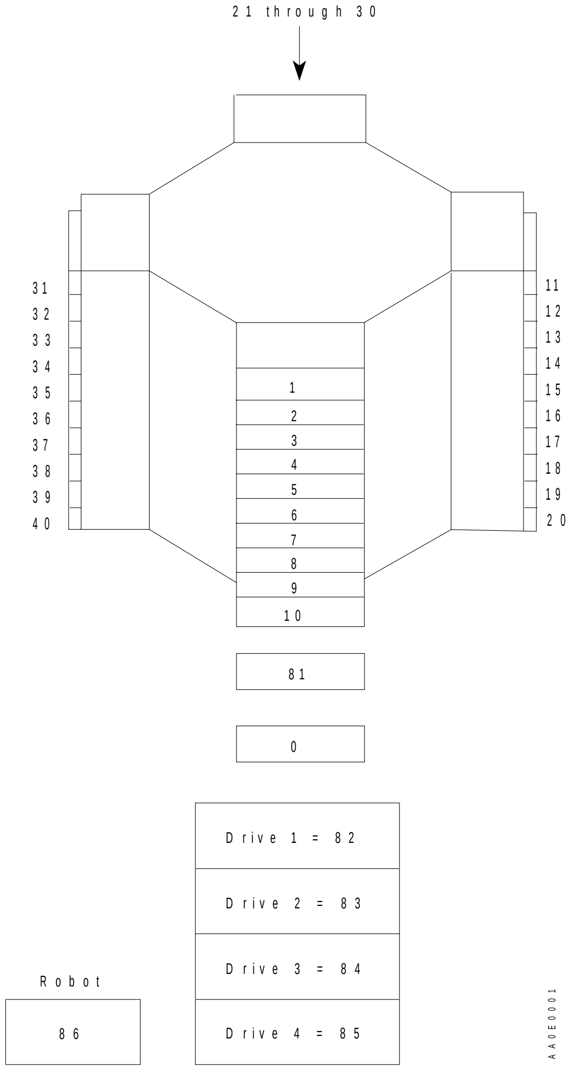


Table 36. Andataco Encore 480, BoxHill BorgBox, or Exabyte EXB-480

Device	SCSI ID	Device Name
Tape drive 1 (element 82)	__________	/dev/rmt/____mt
Tape drive 2 (element 83)	__________	/dev/rmt/____mt
Tape drive 3 (element 84)	__________	/dev/rmt/____mt
Tape drive 4 (element 85)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

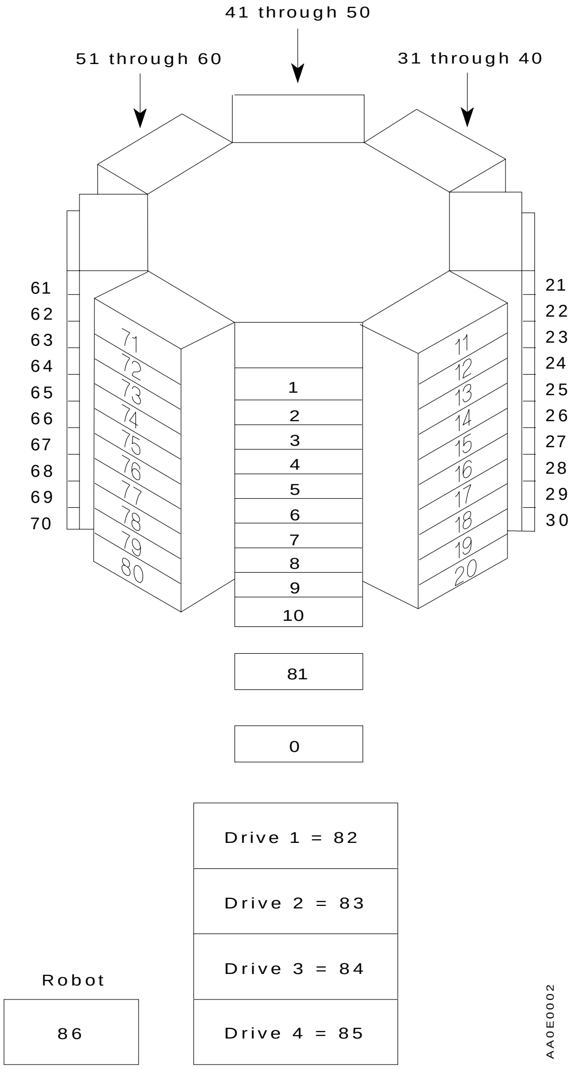
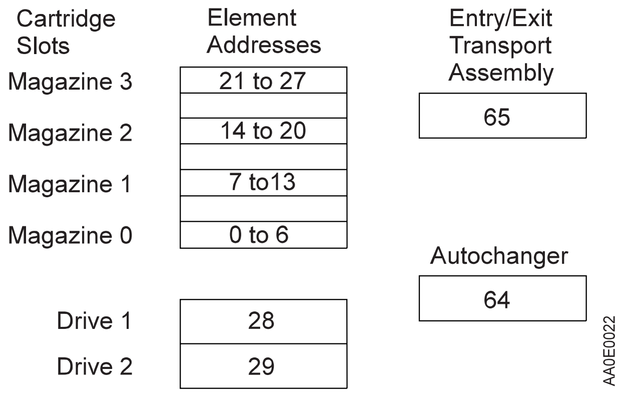


Table 37. Breece Hill Q7, MountainGate D-28, or TTi Q7

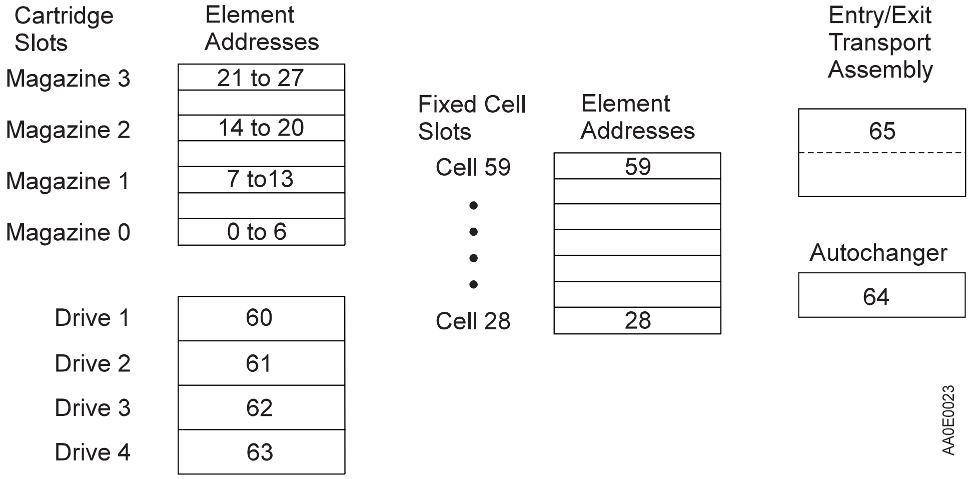
Device	SCSI ID	Device Name
Tape drive 1 (element 60)	__________	/dev/rmt/____mt
Tape drive 2 (element 61)	__________	/dev/rmt/____mt
Tape drive 3 (element 62)	__________	/dev/rmt/____mt
Tape drive 4 (element 63)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb



Note: Magazine slots are numbered from top to bottom.

Table 38. Breece Hill Q47, MountainGate D-60, or TTi Q47

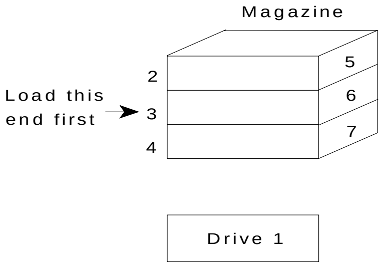
Device	SCSI ID	Device Name
Tape drive 1 (element 60)	__________	/dev/rmt/____mt
Tape drive 2 (element 61)	__________	/dev/rmt/____mt
Tape drive 3 (element 62)	__________	/dev/rmt/____mt
Tape drive 4 (element 63)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb



Note: Magazine slots are numbered from top to bottom. Fixed cell are directly underneath magazines. Underneath magazine 0 are fixed cells element addresses 28 to 35. Underneath magazine 1 are fixed cells 36 to 43 and so on.

Table 39. Hewlett-Packard C1553A or C1561A, or SureStore 12000e

Device	SCSI ID	Device Name
Tape drive (element 1)	__________	/dev/rmt/____mt
Autochanger	__________	/dev/rmt/____lb



The tape drive and the autochanger share a SCSI ID, but have different LUNs. Usually the drive is LUN 0 and the autochanger is LUN 1.

Table 40. Lago LS-380L or STK 9711

Device	SCSI ID	Device Name
Tape drive 1 (element 54) 1	__________	/dev/rmt/____mt
Tape drive 2 (element 55) 2	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

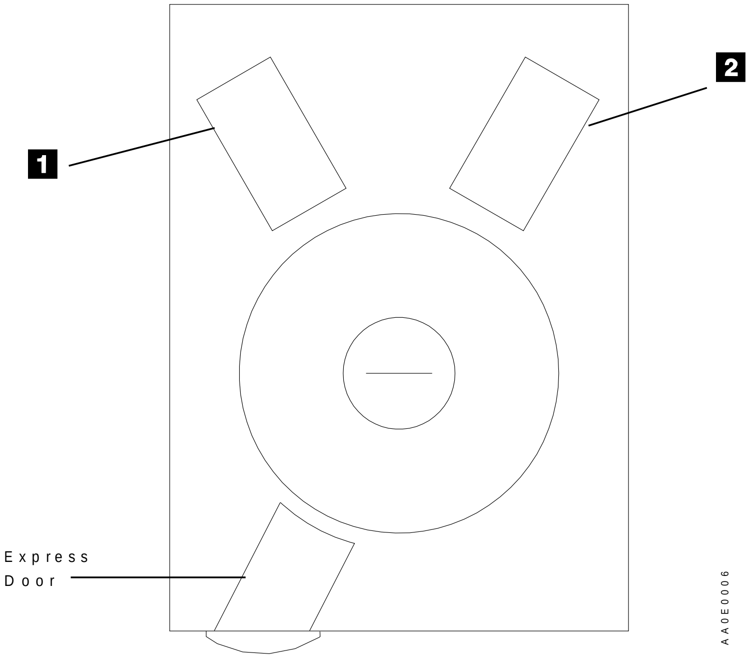
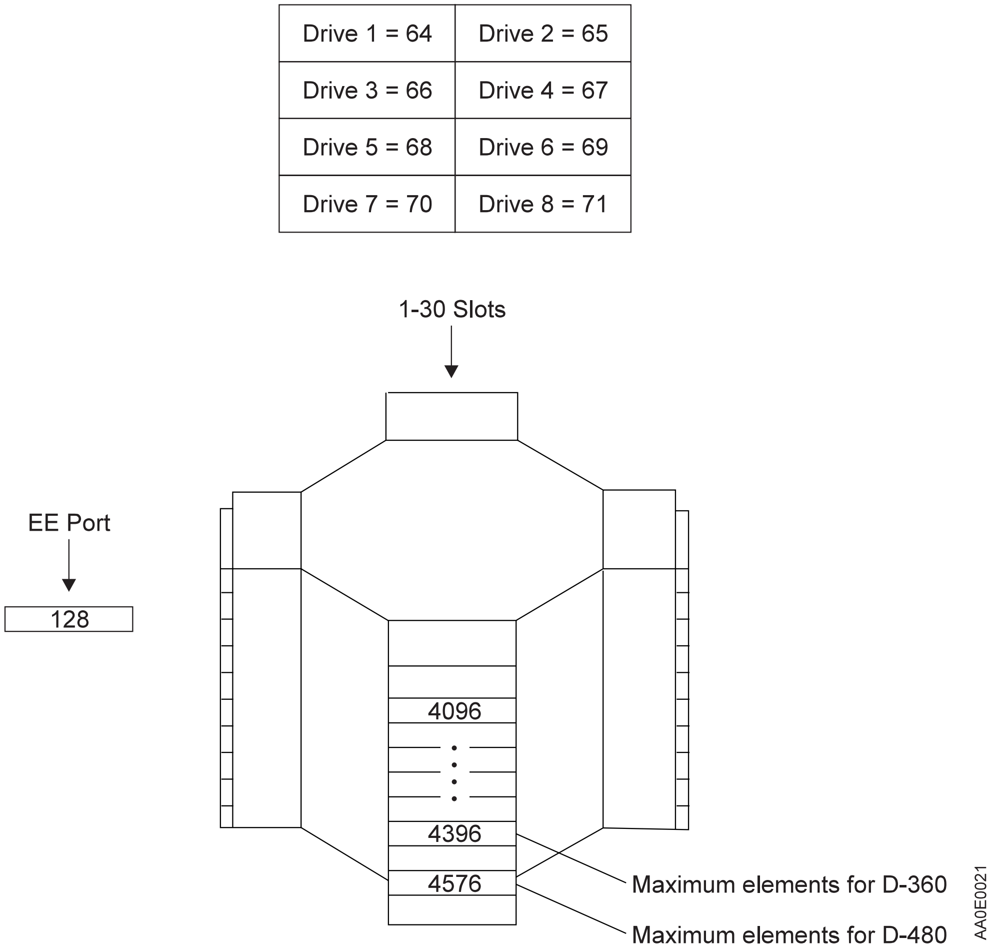


Table 41. MountainGate D-360 or D-480

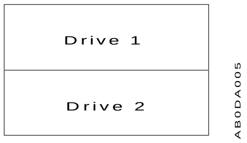
Device	SCSI ID	Device Name
Tape drive 1 (element 64)	__________	/dev/rmt/____mt
Tape drive 2 (element 65)	__________	/dev/rmt/____mt
Tape drive 3 (element 66)	__________	/dev/rmt/____mt
Tape drive 4 (element 67)	__________	/dev/rmt/____mt
Tape drive 5 (element 68)	__________	/dev/rmt/____mt
Tape drive 6 (element 69)	__________	/dev/rmt/____mt
Tape drive 7 (element 70)	__________	/dev/rmt/____mt
Tape drive 8 (element 71)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb



Drive 1 = 64	Drive 2 = 65
Drive 3 = 66	Drive 4 = 67
Drive 5 = 68	Drive 6 = 69
Drive 7 = 70	Drive 8 = 71

Table 42. TTi CTS-8519H

Device	SCSI ID	Device Name
Tape drive 1	__________	/dev/rmt/____mt
Tape drive 2	__________	/dev/rmt/____mt



Two 8mm tape drives are housed in one box. The device can be configured as two independent drives or as one single drive that mirrors the other drive. However, ADSM only supports the configuration with two independent drives.

When the device is configured as two independent drives, the drives have the same SCSI ID but different LUN numbers. LUN 0 is assigned to the left drive (viewed from the front). LUN 1 is assigned to the right drive (viewed from the front).

Table 43. DEC DLT TZ875, TZ875n, TZ885, or TZ885n, or Quantum DLT 2500 or DLT 4500

Device	SCSI ID	Device Name
Tape drive (element 16)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

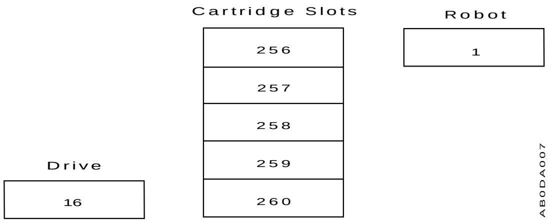


Table 44. DEC DLT TZ877, TZ877n, TZ887, or TZ887n, or Quantum DLT 2700 or DLT 4700

Device	**SCSI ID**	**Device Name**
Tape drive (element 16)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

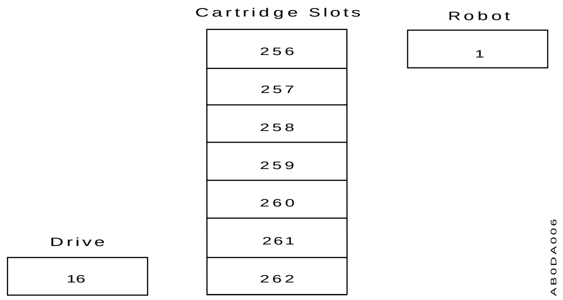


Table 45. Quantum DLT 2500XT or 2700XT Autochanger

Device	**SCSI ID**	**Device Name**
Tape drive (element 16)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

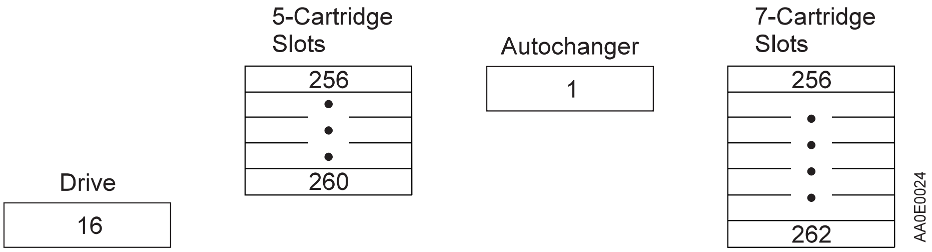
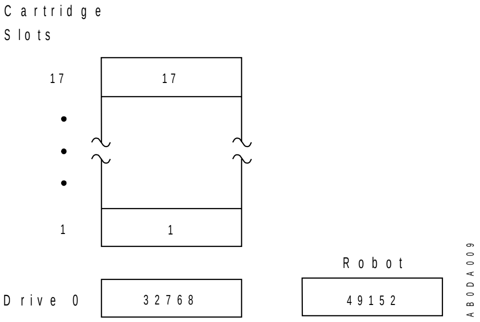


Table 46. AIWA AL-D210

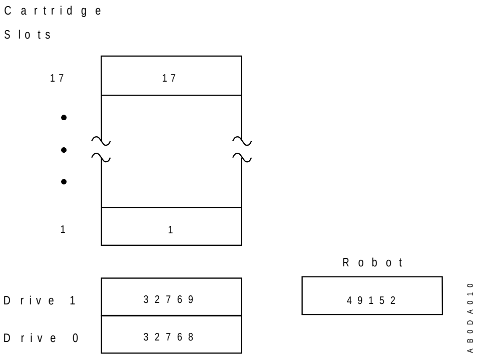
Device	**SCSI ID**	**Device Name**
Tape drive (element 32768)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb



Note: Initialize Status Switch by setting the DIP switch to the ON position.

Table 47. AIWA AL-D220

Device	**SCSI ID**	**Device Name**
Tape drive 0 (element 32768)	__________	/dev/rmt/____mt
Tape drive 1 (element 32769)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb



Note: Initialize Status Switch by setting the DIP switch to the ON position.

Table 48. Exabyte EXB-018/218

Device	SCSI ID	Device Name
Tape drive 1 (element 82)	__________	/dev/rmt/____mt
Tape drive 2 (element 83)	__________	/dev/rmt/____mt
Autochanger	__________	/dev/rmt/____lb

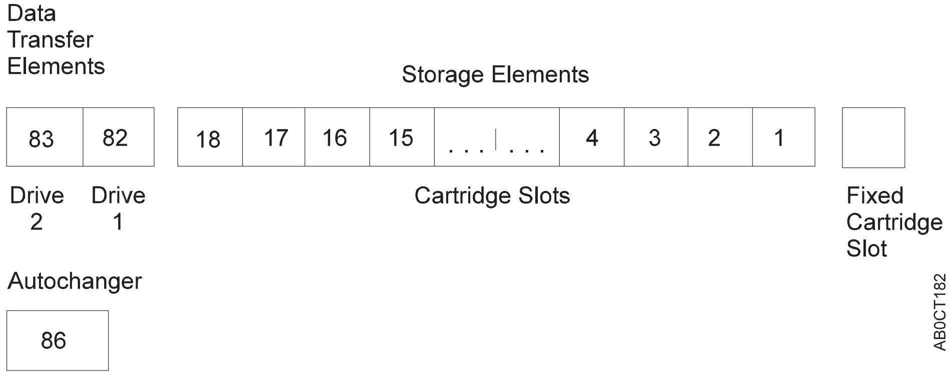


Table 49. Odetics ACL 2/28

Device	SCSI ID	Device Name
Tape drive 1 (element 128)	__________	/dev/rmt/____mt
Tape drive 2 (element 129)	__________	/dev/rmt/____mt
Autochanger	__________	/dev/rmt/____lb

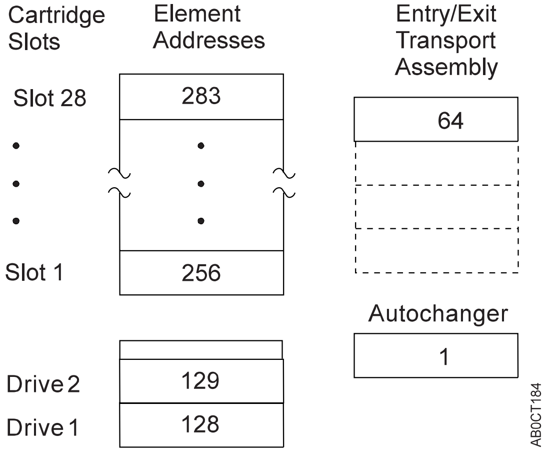
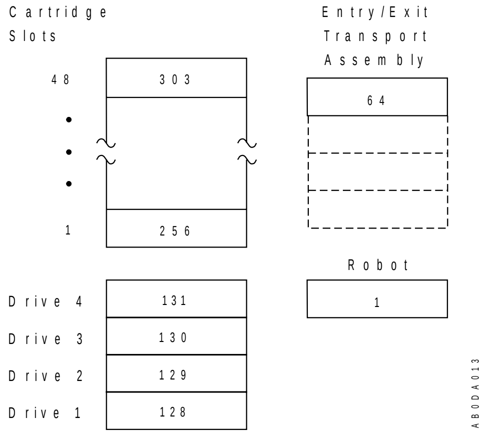


Table 50. Odetics ACL 4/52 or DEC TL10

Device	SCSI ID	Device Name
Tape drive 1 (element 128)	__________	/dev/rmt/____mt
Tape drive 2 (element 129)	__________	/dev/rmt/____mt
Tape drive 3 (element 130)	__________	/dev/rmt/____mt
Tape drive 4 (element 131)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb



Notes:

- The ACL 4/52 or DEC TL10 library has a load port (Entry/Exit Transport Assembly) containing four cartridge slots. Currently, ADSM considers that this library has only one slot (which is the top slot with element address 64) of the four. All ADSM Entry/Exit operations must use the top slot.
- Suggestions for configuring the ACL 4/52 or DEC TL810 for the ADSM server:
  1. Power-up State: the recommended setting is *online*. If set to *offline*, manual intervention is required to bring the library up for use by ADSM.
  2. Automatic Drive Cleaning: set to either *host initiated* or *fully automatic*. When set to *fully automatic*, ADSM will ignore cleaner cartridges with proper bar code labels and will wait while drives are being cleaned.
  3. Retry Option: use the default of *retries enabled*.
  4. Auto Load: must be *disabled*.
  5. Auto Inventory at Power-up: must be *enabled*.
- For any other configurations and settings, refer to the documentation about your library.

Table 51. Odetics ACL 6/176

Device	SCSI ID	Device Name
Tape drive 1 (element 128)	__________	/dev/rmt/____mt
Tape drive 2 (element 129)	__________	/dev/rmt/____mt
Tape drive 3 (element 130)	__________	/dev/rmt/____mt
Tape drive 4 (element 131)	__________	/dev/rmt/____mt
Tape drive 5 (element 132)	__________	/dev/rmt/____mt
Tape drive 6 (element 133)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

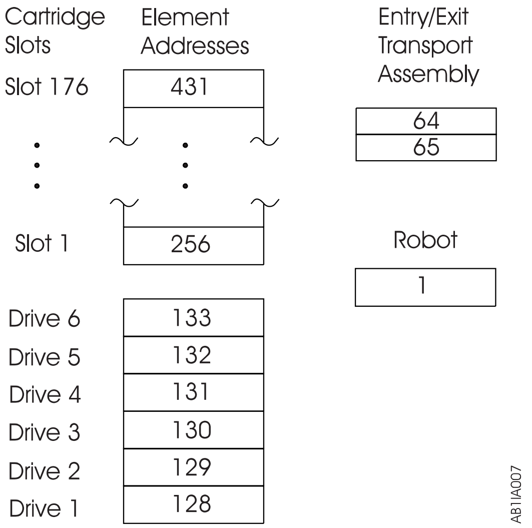


Table 52. Odetics ACL 9/88

Device	SCSI ID	Device Name
Tape drive 1 (element 128)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

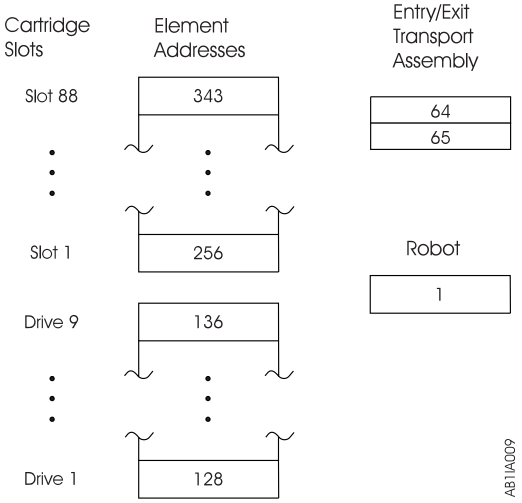
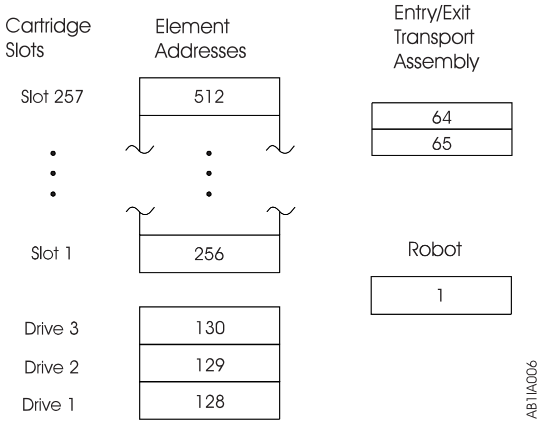


Table 53. Odetics ACL 2640 or DEC TL820

Device	SCSI ID	Device Name
Tape drive 1 (element 128)	__________	/dev/rmt/____mt
Tape drive 2 (element 129)	__________	/dev/rmt/____mt
Tape drive 3 (element 130)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

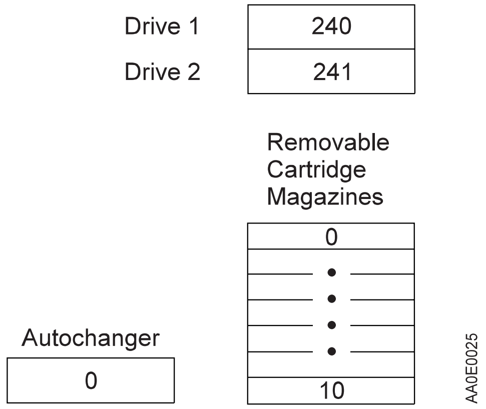


Note: Suggestions for configuring the Odetics ACL 2640 or DEC TL820 for the ADSM server:

1. ADSM does not support the RS-232 interface to the robot, only SCSI.
2. ADSM does not support the PTM (passthrough) mechanism that allows attaching multiple boxes together as one big library.
3. ADSM supports the box in its default mode of operation. (The device has no touchpad such as the Odetics ACL 4/52 has.)

Table 54. OverlandData LXB 2110, 2210, 4110, or 4210

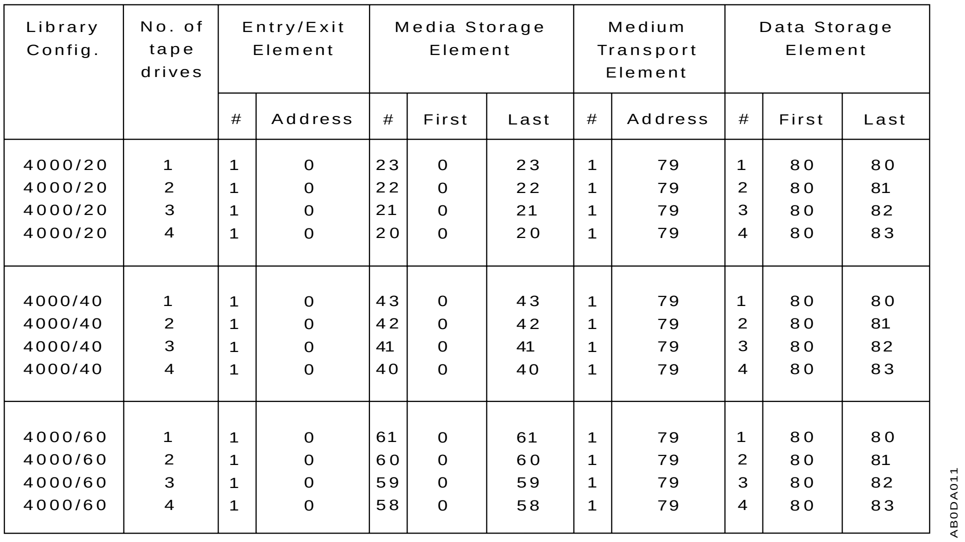
Device	SCSI ID	Device Name
Tape drive 1 (element 240)	__________	/dev/rmt/____mt
Tape drive 2 (element 241)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb



Note: Element 241 applies to dual-drive models.

Table 55. Spectralogic 4000/20, 4000/40, or 4000/60

Device	SCSI ID	Device Name
Tape drive 1 (element 80)	__________	/dev/rmt/____mt
Tape drive 2 (element 81)	__________	/dev/rmt/____mt
Tape drive 3 (element 82)	__________	/dev/rmt/____mt
Tape drive 4 (element 83)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb



Library Config.	No. of tape drives	Entry/Exit Element		Media Storage Element			Medium Transport Element		Data Storage Element		
		#	Address	#	First	Last	#	Address	#	First	Last
4000/20	1	1	0	23	0	23	1	79	1	80	80
4000/20	2	1	0	22	0	22	1	79	2	80	81
4000/20	3	1	0	21	0	21	1	79	3	80	82
4000/20	4	1	0	20	0	20	1	79	4	80	83
4000/40	1	1	0	43	0	43	1	79	1	80	80
4000/40	2	1	0	42	0	42	1	79	2	80	81
4000/40	3	1	0	41	0	41	1	79	3	80	82
4000/40	4	1	0	40	0	40	1	79	4	80	83
4000/60	1	1	0	61	0	61	1	79	1	80	80
4000/60	2	1	0	60	0	60	1	79	2	80	81
4000/60	3	1	0	59	0	59	1	79	3	80	82
4000/60	4	1	0	58	0	58	1	79	4	80	83

AB0DA011

Table 56. StorageTek 9704

Device	SCSI ID	Device Name
Tape drive 0 (element 25)	__________	/dev/rmt/____mt
Tape drive 1 (element 26)	__________	/dev/rmt/____mt
Robot	__________	/dev/rmt/____lb

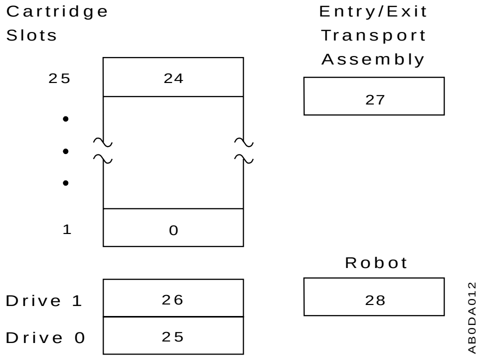


Table 57. Qualstar TLS 4mm Autochanger

Device	SCSI ID	Device Name
Tape drive 1 (element 500)	__________	/dev/rmt/____mt
Tape drive 2 (element 501)	__________	/dev/rmt/____mt
Tape drive 3 (element 502)	__________	/dev/rmt/____mt
Tape drive 4 (element 503)	__________	/dev/rmt/____mt
Autochanger (element 700)	__________	/dev/rmt/____lb

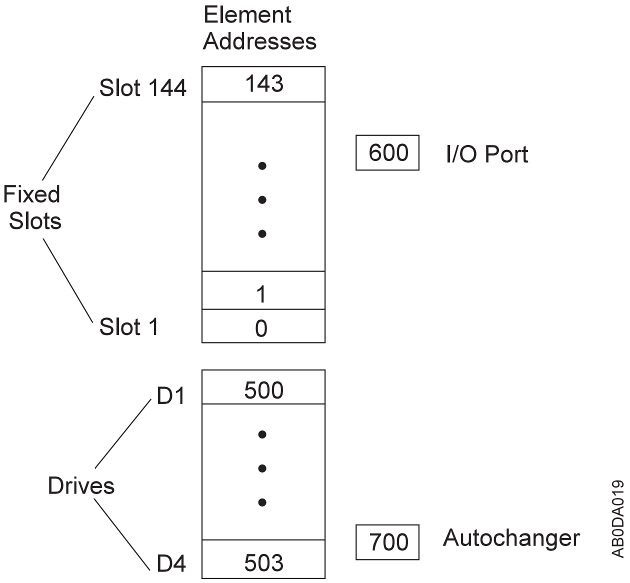


Table 58. Qualstar TLS 8mm Autochanger

Device	SCSI ID	Device Name
Tape drive 1 (element 500)	__________	/dev/rmt/____mt
Tape drive 2 (element 501)	__________	/dev/rmt/____mt
Tape drive 3 (element 502)	__________	/dev/rmt/____mt
Tape drive 4 (element 503)	__________	/dev/rmt/____mt
Tape drive 5 (element 504)	__________	/dev/rmt/____mt
Tape drive 6 (element 505)	__________	/dev/rmt/____mt
Autochanger (element 700)	__________	/dev/rmt/____lb

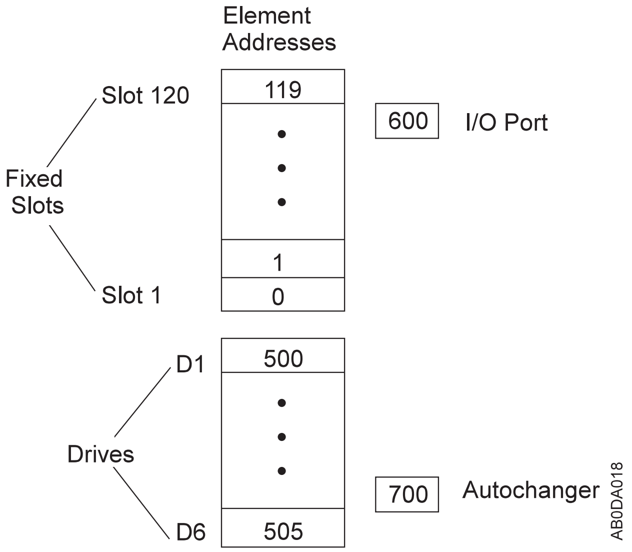


Table 59. IBM 3995 A63, 063, 163, or IBM 3995 C60, C62, C64, C66, or C68

Device	SCSI ID	Device Name
Tape drive 1 (element 1)	__________	/dev/rmt/____mt
Tape drive 2 (element 2)	__________	/dev/rmt/____mt
Tape drive 3 (element 3)	__________	/dev/rmt/____mt
Tape drive 4 (element 4)	__________	/dev/rmt/____mt
Tape drive 5 (element 5)	__________	/dev/rmt/____mt
Tape drive 6 (element 6)	__________	/dev/rmt/____mt
Autochanger (element 8001)	__________	/dev/rmt/____lb

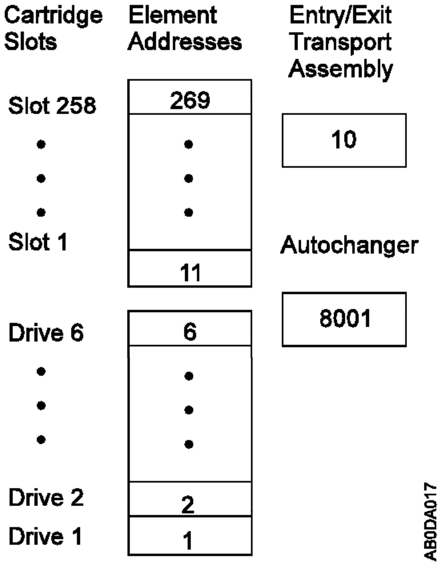


Table 60. HP 20T or 40T

Device	SCSI ID	Device Name
Drive 1 (element 1)	__________	/dev/rmt/____mt
Drive 2 (element 2)	__________	/dev/rmt/____mt
Robot (element 0)	__________	/dev/rmt/____lb

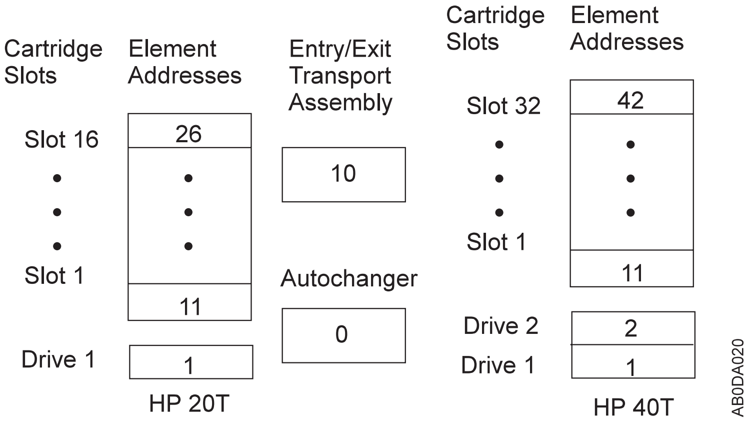
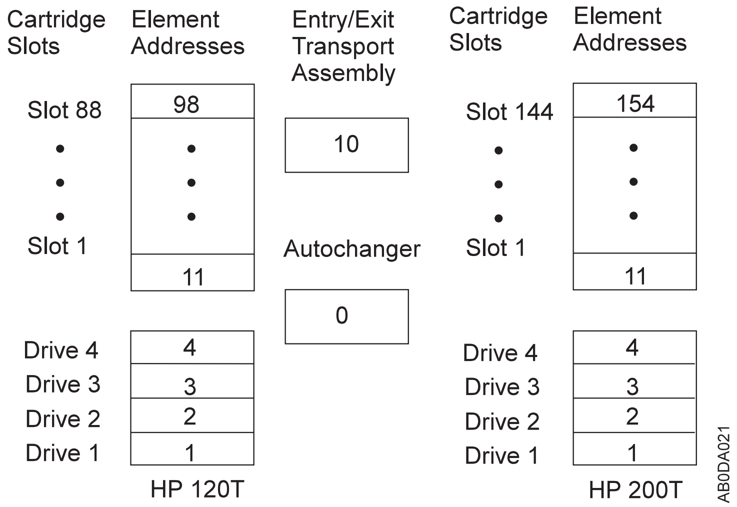


Table 61. HP 120T or 200T

Device	SCSI ID	Device Name
Drive 1 (element 1)	__________	/dev/rmt/____mt
Drive 2 (element 2)	__________	/dev/rmt/____mt
Drive 3 (element 3)	__________	/dev/rmt/____mt
Drive 4 (element 4)	__________	/dev/rmt/____mt
Robot (element 0)	__________	/dev/rmt/____lb



Appendix B. External Media Management Interface Description

This appendix contains General-use Programming Interface and Associated Guidance Information about the interface that ADSM provides to external media management programs. To use the interface, you must first define an EXTERNAL library. For information on this library type, see “External Libraries” on page 19.

The interface consists of request description strings that ADSM sends and response strings that the external program sends.

The details of the request types and the required processing are described in the sections that follow. The request types are:

- Initialization of the external program
- Volume mount
- Volume dismount
- Volume release (return to scratch)

Processing during ADSM Server Initialization

Ensure that the external media management program cooperates with the ADSM server during the server’s initialization. For each external library defined to the ADSM server, the following must occur during server initialization:

1. The ADSM server loads the external program in a newly created process.
2. The server sends an initialization request description string, in text form, into the standard input (**stdin**) stream of the external program. The server waits for the response.
3. When the external process completes the request, the process must write an initialization response string, in text form, into its standard output (**stdout**) stream.
4. The external program must end by calling the stdlib **exit** routine.

Processing for Volume Mount, Dismount, and Release Requests

Ensure that the external media management program responds to ADSM server requests, other than initialization, according to the following process:

1. If the request is for a volume mount or release, the external program must be initialized:
   a. The server loads the external program in a newly created process.
   b. The server sends an initialization request description string, in text form, into the standard input (**stdin**) stream of the external program. The ADSM server waits for the response.
   c. When the external process completes the request, the process must write an initialization response string, in text form, into its standard output (**stdout**) stream.

© Copyright IBM Corp. 1995, 1997

2. The server sends the request description string, in text form, into the standard input (**stdin**) stream of the external program. The ADSM server waits for the response.
3. When the external process completes the request, the process must write a response string, in text form, into its standard output (**stdout**) stream.
4. If the request was for a volume mount, the external program must remain active and ready to accept the volume dismount request from the ADSM server. Otherwise, the program must end by using the stdlib **exit** routine.

Initialization Requests

When the ADSM server is started, the server sends an initialization request to the external media management program for each EXTERNAL library. The external program must process this request to ensure that the external program is present, functional, and ready to process ADSM requests. If the initialization request is successful, ADSM informs its operators that the external program reported its readiness for ADSM operations. Otherwise, ADSM reports a failure to its operators.

ADSM does not attempt any other type of operation with that library until an initialization request has succeeded. For a mount or release operation, the ADSM server sends an initialization request first. If the initialization is successful, the request is sent. If the initialization is not successful, the request fails. The external media management program can detect whether the initialization request is being sent by itself or with another request by detecting end-of-file on the **stdin** stream. When end-of-file is detected, the external program must end by using the stdlib **exit** routine (not the **return** call).

When a valid response is sent by the external program, the external program must end by using the **exit** routine.

Format of the ADSM request:

```
INITIALIZE libraryname
```

where *libraryname* is the name of the EXTERNAL library as defined to ADSM.

Format of the external program response:

```
INITIALIZE libraryname COMPLETE, RESULT=resultcode
```

where:

libraryname
Specifies the name of the EXTERNAL library as defined to ADSM.

resultcode
One of the following:

- SUCCESS
- NOT_READY
- INTERNAL_ERROR

Volume Mount Requests

When the ADSM server requires a volume mount, the server starts the external media management program, issues a request to initialize, then issues a request to mount a volume. The external program is responsible for verifying that this request is coming from ADSM and not from an unauthorized system.

The volume mounted by the external media management program must be a tape with a standard IBM label. When the external program completes the mount request, the program must send a response. If the mount was successful, the external program must remain active. If the mount failed, the external program must end immediately by using the **exit** routine.

Format of the ADSM request:

```
MOUNT libraryname volname accessmode devicetypes timelimit userid volumenumber
```

where:

libraryname
Specifies the name of the EXTERNAL library as defined to ADSM.

volname
Either SCRTCH if the request is for a scratch mount, or the actual volume name if the request is for an existing volume.

accessmode
Specifies the access mode required for the volume. Possible values are `READONLY` and `READWRITE`.

devicetypes
Specifies a list of possible device types and formats that can be used to satisfy the request for the volume. The most preferred device type and format is first in the list. Items are separated by commas, with no intervening spaces. Possible values are:

- 3480
- 3480XF
- 3490E
- 3570
- 3590
- 4MM_DDS1
- 4MM_DDS1C
- 4MM_DDS2
- 4MM_DDS2C
- 8MM_8200
- 8MM_8205
- 8MM_8500
- 8MM_8500C
- 8MM_8900
- DLT_2000

- DLT_4000
- DLT_7000
- OPT_RW_650MB
- OPT_RW_1300MB
- OPT_RW_2600MB
- OPT_WORM_650MB
- OPT_WORM_1300MB

timelimit
Specifies the maximum number of minutes that the server waits for the volume to be mounted. If the mount request is not completed within this time, the external manager responds with the result code TIMED_OUT.

userid
Specifies the user ID of the process that needs access to the drive.

volumenumber
For non-optical media, the *volumenumber* is 1. For optical media, the *volumenumber* is 1 for side A, 2 for side B.

Format of the external program response:

```
MOUNT libraryname volname COMPLETE ON specialfile, RESULT=resultcode
```

where:

libraryname
Specifies the name of the EXTERNAL library as defined to ADSM.

volname
Specifies the name of the volume mounted for the request.

specialfile
The fully qualified path name of the device special file for the drive in which the volume was mounted. If the mount request fails, the value should be set to /dev/null.

The external program must ensure that the special file is closed before the response is returned to the ADSM server.

resultcode
One of the following:

- SUCCESS
- DRIVE_ERROR
- LIBRARY_ERROR
- VOLUME_UNKNOWN
- VOLUME_UNAVAILABLE
- CANCELLED
- TIMED_OUT
- INTERNAL_ERROR

Volume Dismount Requests

When a successful mount operation completes, the external process must wait for a request to dismount the volume. When the dismount operation completes, the external program must send a response to the ADSM server.

After the dismount response is sent, the external process ends immediately by using the **exit** routine.

Format of the ADSM request:

```
DISMOUNT libraryname volname
```

where:

libraryname
Specifies the name of the EXTERNAL library as defined to ADSM.

volname
Specifies the name of the volume to be dismounted.

Format of the external program response:

```
DISMOUNT libraryname volname COMPLETE, RESULT=resultcode
```

where:

libraryname
Specifies the name of the EXTERNAL library as defined to ADSM.

volname
Specifies the name of the volume dismounted.

resultcode
One of the following:

- SUCCESS
- DRIVE_ERROR
- LIBRARY_ERROR
- INTERNAL_ERROR

Volume Release Requests

When the ADSM server returns a volume to scratch status, the server starts the external media management program, issues a request to initialize, then issues a request to release a volume.

The external program must send a response to the release request. No matter what response is received from the external program, ADSM returns the volume to scratch. For this reason, ADSM and the external program can have conflicting information on which volumes are scratch. If an error occurs, the external program logs the failure so that the external library inventory can be synchronized later with ADSM. The synchronization can be a manual operation.

Format of the ADSM request:

```
RELEASE libraryname volname
```

where:

libraryname
Specifies the name of the EXTERNAL library as defined to ADSM.

volname
Specifies the name of the volume to be returned to scratch (released).

Format of the external program response:

```
RELEASE libraryname volname COMPLETE, RESULT=resultcode
```

where:

libraryname
Specifies the name of the EXTERNAL library as defined to ADSM.

volname
Specifies the name of the volume returned to scratch (released).

resultcode
One of the following:

- SUCCESS
- VOLUME_UNKNOWN
- VOLUME_UNAVAILABLE
- INTERNAL_ERROR

Appendix C. Interface Cross-Reference

The following table lists each ADSM command and shows if and where the function performed by that command can be performed on the graphical user interface. You can find detailed help for using the GUI in the online help facilities. For details about the commands, refer to the *ADSM Administrator's Reference* and the command-line interface online help (accessed through the HELP command).

Table 62 (Page 1 of 9). Interface Cross Reference

Command	**GUI**
ACTIVATE POLICYSET	1. Policy Domains 2. Policy Sets
ASSIGN DEFMGMTCLASS	1. Policy Domains 2. Management Classes
AUDIT LIBRARY	—
AUDIT LICENSES	Server
AUDIT VOLUME	1. Storage Pools 2. Storage Pool Volumes
BACKUP DB	Database
BACKUP DEVCONFIG	Server
BACKUP STGPOOL	Storage Pools
BACKUP VOLHISTORY	1. Server 2. Sequential Volume History
CANCEL PROCESS	1. Server 2. Processes
CANCEL REQUEST	—
CANCEL SESSION	1. Server 2. Sessions
CHECKIN LIBVOLUME	—
CHECKOUT LIBVOLUME	—
COMMIT	—
COPY DOMAIN	Policy Domains
COPY MGMTCLASS	1. Policy Domains 2. Management Class
COPY POLICYSET	1. Policy Domains 2. Policy Sets

© Copyright IBM Corp. 1995, 1997

Table 62 (Page 2 of 9). Interface Cross Reference

Command	GUI
COPY SCHEDULE	1. Central Scheduler 2. Either a. Administrative Command Schedules or b. Backup/Archive Schedules
DEFINE ASSOCIATION	1. Central Scheduler 2. Either a. Administrative Command Events or b. Backup/Archive Events
DEFINE COPYGROUP	1. Policy Domains 2. Backup Copy Groups
DEFINE DBBACKUPTRIGGER	Database
DEFINE DBCOPY	1. Database 2. Database Volumes
DEFINE DBVOLUME	1. Database 2. Database Volumes
DEFINE DEVCLASS	—
DEFINE DOMAIN	Policy Domains
DEFINE DRIVE	—
DEFINE LIBRARY	—
DEFINE LOGCOPY	1. Database Recovery Log 2. Recovery Log Volumes
DEFINE LOGVOLUME	1. Database Recovery Log 2. Recovery Log Volumes
DEFINE MACHINE	—
DEFINE MACHNODEASSOCIATION	—
DEFINE MGMTCLASS	1. Policy Domains 2. Management Class
DEFINE POLICYSET	1. Policy Domains 2. Policy Sets
DEFINE RECOVERYMEDIA	—
DEFINE RECMEDMACHASSOCIATION	—

Table 62 (Page 3 of 9). Interface Cross Reference

Command	**GUI**
DEFINE SCHEDULE	1. Central Scheduler 2. Either a. Administrative Command Schedules or b. Backup/Archive Schedules
DEFINE STGPOOL	Storage Pools
DEFINE VOLUME	1. Storage Pools 2. Storage Pool Volumes
DELETE ASSOCIATION	1. Central Scheduler 2. Either a. Administrative Command Events or b. Backup/Archive Events
DELETE COPYGROUP	1. Policy Domains 2. Backup Copy Groups or Archive Copy Groups
DELETE DBBACKUPTRIGGER	Database
DELETE DBVOLUME	1. Database 2. Database Volumes
DELETE DEVCLASS	—
DELETE DOMAIN	Policy Domains
DELETE DRIVE	—
DELETE EVENT	1. Central Scheduler 2. Either a. Administrative Command Events or b. Backup/Archive Events
DELETE FILESPACE	File Spaces
DELETE LIBRARY	—
DELETE LOGVOLUME	1. Database Recovery Log 2. Recovery Log Volumes
DELETE MGMTCLASS	1. Policy Domains 2. Management Class
DELETE POLICYSET	1. Policy Domains 2. Policy Sets

Table 62 (Page 4 of 9). Interface Cross Reference

Command	**GUI**
DELETE SCHEDULE	1. Central Scheduler 2. Administrative Command Schedules or 3. Backup/Archive Schedules
DELETE STGPOOL	Storage Pools
DELETE VOLHISTORY	1. Server 2. Sequential Volume History
DELETE VOLUME	1. Storage Pools 2. Storage Pool Volumes
DISABLE	Server
DISMOUNT VOLUME	—
DSMLABEL	—
DSMSERV DISPLAY DBBACKUPVOLUME	—
DSMSERV DISPLAY DBVOLUMES	—
DSMSERV DISPLAY LOGVOLUMES	—
DSMSERV EXTEND LOG	—
DSMSERV RESTORE DB	—
ENABLE	Server
EXPIRE INVENTORY	—
EXPORT ADMIN	Administrators
EXPORT NODE	Nodes
EXPORT POLICY	Policy Domains
EXPORT SERVER	Server
EXTEND DB	Database
EXTEND LOG	Database Recovery Log
GRANT AUTHORITY	Administrators
GETIPXAD	—
HALT	Server
HELP	Help (menu bar), Online documentation
IMPORT ADMIN	Administrators
IMPORT NODE	Nodes
IMPORT POLICY	Policy Domains

Table 62 (Page 5 of 9). Interface Cross Reference

Command	**GUI**
IMPORT SERVER	Server
INSERT MACHINE	—
LOCK ADMIN	Administrators
LOCK NODE	Nodes
MACRO	—
MOVE DATA	1. Storage Pools 2. Storage Pool Volumes
MOVE DRMEDIA	—
PREPARE	—
QUERY ACTLOG	1. Server 2. Activity Log
QUERY ADMIN	Administrators
QUERY ASSOCIATION	1. Central Scheduler 2. Either a. Administrative Command Events or b. Backup/Archive Events
QUERY AUDITOCCUPANCY	Nodes
QUERY CONTENT	1. Storage Pools 2. Storage Pool Volumes
QUERY COPYGROUP	1. Policy Domains 2. Backup Copy Groups
QUERY DB	Database
QUERY DBBACKUPTRIGGER	Database
QUERY DBVOLUME	1. Database 2. Database Volumes
QUERY DEVCLASS	—
QUERY DOMAIN	Policy Domains
QUERY DRIVE	—
QUERY DRMEDIA	—
QUERY DRMSTATUS	—

Table 62 (Page 6 of 9). Interface Cross Reference

Command	**GUI**
QUERY EVENT	1. Central Scheduler 2. Either a. Administrative Command Events or b. Backup/Archive Events
QUERY FILESPACE	File Spaces
QUERY LIBRARY	—
QUERY LIBVOLUME	—
QUERY LICENSE	Server
QUERY LOG	Database Recovery Log
QUERY LOGVOLUME	1. Database Recovery Log 2. Recovery Log Volumes
QUERY MACHINE	—
QUERY MGMTCLASS	1. Policy Domains 2. Management Class
QUERY MOUNT	—
QUERY NODE	Nodes
QUERY OCCUPANCY	—
QUERY OPTION	—
QUERY POLICYSET	1. Policy Domains 2. Policy Sets
QUERY PROCESS	1. Server 2. Processes
QUERY RECOVERYMEDIA	—
QUERY REQUEST	—
QUERY SCHEDULE	1. Central Scheduler 2. Either a. Administrative Command Schedules or b. Backup/Archive Schedules
QUERY SESSION	1. Server 2. Sessions
QUERY STATUS	Server
QUERY STGPOOL	Storage Pools

Table 62 (Page 7 of 9). Interface Cross Reference

Command	**GUI**
QUERY VOLHISTORY	1. Server 2. Sequential Volume History
QUERY VOLUME	1. Storage Pools 2. Storage Pool Volumes
QUIT	—
REDUCE DB	Database
REDUCE LOG	Database Recovery Log
REGISTER ADMIN	Administrators
REGISTER LICENSE	—
REGISTER NODE	Nodes
REMOVE ADMIN	Administrators
REMOVE NODE	Nodes
RENAME ADMIN	Administrators
RENAME FILESPACE	File Spaces
RENAME NODE	Nodes
REPLY	—
RESET BUFPOOL	—
RESET DBMAXUTILIZATION	Database
RESET LOGCONSUMPTION	—
RESET LOGMAXUTILIZATION	Database Recovery Log
RESTORE STGPOOL	Storage Pools
RESTORE VOLUME	1. Storage Pools 2. Storage Pool Volumes
REVOKE AUTHORITY	Administrators
ROLLBACK	—
SET ACCOUNTING	Server
SET ACTLOGRETENTION	Server
SET AUTHENTICATION	Server
SET EVENTRETENTION	1. Central Scheduler 2. Either a. Administrative Command Events or b. Backup/Archive Events

Table 62 (Page 8 of 9). Interface Cross Reference

Command	GUI
SET DRMCHECKLABEL	—
SET DRMCMDFILENAME	—
SET DRMCOPYSTGPOOL	—
SET DRMCOURIERNAME	—
SET DRMDBBACKUPEXPIREDAYS	—
SET DRMFILEPROCESS	—
SET DRMINSTRPREFIX	—
SET DRMNOTMOUNTABLENAME	—
SET DRMPLANPREFIX	—
SET DRMPLANVPOSTFIX	—
SET DRMPRIMSTGPOOL	—
SET DRMVAULTNAME	—
SET LICENSEAUDITPERIOD	Server
SET LOGMODE	Database Recovery Log
SET MAXCMDRETRIES	Central Scheduler
SET MAXSCHEDSESSIONS	Central Scheduler
SET PASSEXP	Server
SET QUERYSCHEDPERIOD	Central Scheduler
SET RANDOMIZE	Central Scheduler
SET REGISTRATION	Server
SET RETRYPERIOD	Central Scheduler
SET SCHEDMODES	Central Scheduler
SET SERVERNAME	Server
UNLOCK ADMIN	Administrators
UNLOCK NODE	Nodes
UPDATE ADMIN	Administrators
UPDATE COPYGROUP	1. Policy Domains 2. Backup Copy Groups
UPDATE DBBACKUPTRIGGER	Database
UPDATE DEVCLASS	—
UPDATE DOMAIN	Policy Domains

Table 62 (Page 9 of 9). Interface Cross Reference

Command	**GUI**
UPDATE DRIVE	—
UPDATE LIBRARY	—
UPDATE LIBVOLUME	—
UPDATE MGMTCLASS	1. Policy Domains 2. Management Class
UPDATE NODE	Nodes
UPDATE POLICYSET	1. Policy Domains 2. Policy Sets
UPDATE RECOVERYMEDIA	—
UPDATE SCHEDULE	1. Central Scheduler 2. Either a. Administrative Command Schedules or b. Backup/Archive Schedules
UPDATE STGPOOL	Storage Pools
UPDATE VOLUME	1. Storage Pools 2. Storage Pool Volumes
VALIDATE POLICYSET	1. Policy Domains 2. Policy Sets
VARY	*Database volumes:* 1. Database 2. Database Volumes *Recovery log volumes:* 1. Database Recovery Log 2. Recovery Log Volumes *Storage pool volumes:* 1. Storage Pools 2. Storage Pool Volumes

Glossary

The terms in this glossary are defined as they pertain to the ADSM library. If you do not find the term you are looking for, refer to the *IBM 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 (ANSI). Copies may be purchased from the American National Standards Institute, 11 West 42nd Street, New York 10036.
- 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 JTC2/SC1).

A

absolute. A backup copy group mode value indicating that a file is considered for incremental backup even if the file has not changed since the last backup. See also *mode*. Contrast with *modified*.

access mode. A storage pool and storage volume attribute that specifies whether data can be written to or read from storage pools or storage volumes. The access mode can be read/write, read-only, or unavailable. Volumes in primary storage pools can also have an access mode of destroyed. Volumes in copy storage pools can also have an access mode of offsite.

accounting facility. A facility that records statistics about client session activity.

accounting records. Files that record session resource usage at the end of each client session.

activate. The process of validating the contents of a policy set and copying the policy set to the ACTIVE policy set.

active policy set. The policy set within a policy domain that contains the most recently activated policy currently in use by all client nodes assigned to that policy domain. See *policy set*.

active version. The most recent backup copy of a file stored by ADSM. Such a file is exempt from deletion until a backup detects that the user has either replaced the file with a newer version, or has explicitly deleted the file from the workstation. Contrast with *inactive version*.

activity log. A log that records normal activity messages generated by the server. These messages include information about server and client operations, such as the start time of sessions or device I/O errors. Each message includes a message ID, date and time stamp, and a text description. The number of days to retain messages in the activity log can be specified.

administrative client. A program that runs on a file server, workstation, or mainframe that allows administrators to control and monitor the server through administrator commands. Contrast with *backup-archive client.*

administrative command schedule. A database record that describes the planned processing of an administrative command during a specific time period. See also *client schedule*.

administrative privilege class. A permission granted to an administrator that controls the commands that the administrator can issue. See *system privilege class, analyst privilege class, operator privilege class, policy privilege class or storage privilege class*.

administrative session. A period of time in which an administrator user ID can communicate with a server to perform administrative tasks. Contrast with *client node session*.

administrator. A user who has been registered to the server. Administrators can be authorized to one or more of the following administrative privilege classes: system, policy, storage, operator, or analyst. Administrators can use the administrative client to enter server commands and queries in accordance with their privileges.

administrator definition. Server control information that includes the administrator's name, password, contact information, administrative privilege classes, policy domains and storage pools assigned to an administrator, and whether the administrative ID is locked from the server. An administrator definition can be exported from a source server and imported to a target server at a later date.

ADSM. ADSTAR Distributed Storage Manager.

© Copyright IBM Corp. 1995, 1997

ADSM application program interface (API). A set of functions that applications running on a client platform can call to store, query, and retrieve objects from ADSM storage.

ADSTAR Distributed Storage Manager (ADSM). A client/server program that provides storage management to customers in a multivendor computer environment.

Advanced Interactive Executive (AIX). An operating system used in the RISC System/6000 computers. The AIX operating system is IBM's implementation of the UNIX operating system.

Advanced Peer-to-Peer Networking (APPN). An extension to the LU6.2 peer orientation for end-user services. See *SNA LU6.2* and *Systems Network Architecture*.

Advanced Program-to-Program Communication (APPC). An implementation of the SNA/SDLC LU6.2 protocol that allows interconnected systems to communicate and share the processing of programs. See *SNA LU6.2*, *Systems Network Architecture*, and *Common Programming Interface Communications*.

AFS. Andrew file system.

AIX. Advanced Interactive Executive.

analyst privilege class. An administrative privilege class that allows an administrator to reset statistics.

Andrew file system (AFS). A distributed file system developed for UNIX operating systems.

API. Application program interface.

APPC. Advanced Program-to-Program Communication.

APPN. Advanced Peer-to-Peer Networking.

archive. A function that allows users to copy one or more files to a storage pool for long-term storage. Archive copies may be accompanied by descriptive information and may be retrieved by archive date, by file name, or by description. Contrast with *retrieve*.

archive copy. A user file that has been archived to an ADSM storage pool.

archive copy group. A policy object containing attributes that control the generation, destination, and expiration of archive files. An archive copy group belongs to a management class.

ARCHIVEPOOL. A disk storage pool defined by ADSM at installation. It can be the destination for client files that are archived to the server. See *storage pool*.

archive retention grace period. The number of days ADSM retains an archive copy when the server is unable to rebind the file to an appropriate management class.

AS/400. Application System/400.

assigned capacity. The portion of available space that can be used to store database or recovery log information. See also *available space*.

association. The relationship between a client node and a client schedule. An association identifies the name of a schedule, the name of the policy domain to which the schedule belongs, and the name of a client node that performs scheduled operations.

audit. The process of checking for logical inconsistencies between information that the server has and the actual condition of the system. ADSM has processes for auditing volumes, the database, libraries, and licenses. For example, in auditing a volume ADSM checks for inconsistencies between information about backed up or archived files stored in the database and actual data associated with each backup version or archive copy in server storage.

authentication. The process of checking a user's password before allowing that user access to the server. Authentication can be turned on or off by an administrator with system privilege.

autochanger. A small multislot tape device that has a mechanism that automatically puts tape cartridges into the tape drive or drives. Also called *medium* or *media changer*, or a *library*.

availability management. Managing recovery from relatively common computer system outages such as a disk drive head crash. Recovery is often accomplished by using disk mirroring and other forms of RAID technology, or by maintaining onsite backup copies of data.

available space. The amount of space, in megabytes, that is available to the database and recovery log. This space can be used to extend the capacity of the database or recovery log, or to provide sufficient free space before a volume is deleted from the database or recovery log.

awk. In AIX, a pattern-matching program for processing text files. With the DRM feature, you can use an awk script to break up the disaster recovery plan file into usable parts.

B

background process. A server process that runs in the background, allowing the administrative client to be used for other work.

backup. The process of copying information for safekeeping. ADSM has processes for backing up user files, the database, and storage pools. For example, users can back up one or more files to a storage pool to ensure against loss of data. Contrast with *restore*. See also *database backup series* and *incremental backup*.

backup-archive client. A program that runs on a workstation or file server and provides a means for users to back up, archive, restore, and retrieve files. Contrast with *administrative client*.

backup copy. A user file that has been backed up to an ADSM storage pool.

backup copy group. A policy object containing attributes that control the generation, destination, and expiration of backup files. A backup copy group belongs to a management class.

BACKUPPOOL. A disk storage pool defined by ADSM at installation. It can be the destination for client files that are backed up to the server. See *storage pool*.

backup retention grace period. The number of days ADSM retains a backup version after the server is unable to rebind the file to an appropriate management class.

backup series. See *database backup series*.

backup version. A file, directory, or file space that a user has backed up, which resides in ADSM server storage. There may be more than one backup version of a file in the storage pool, but at most only one is an active backup version. See *active version* and *inactive version*.

binding. The process of associating a file with a management class name. See *rebinding*.

boot media. Media that contains operating system and other files essential to running a workstation or server.

buffer. Storage used to compensate for differences in the data rate flow, when transferring data from one device to another.

buffer pool. Temporary space used by the server to hold database or recovery log pages. See *database buffer pool* and *recovery log buffer pool*.

buffer pool size. The size of an area in memory used to store database or recovery log pages.

C

cache. The process of leaving a duplicate copy on random access media when the server migrates a file to another storage pool in the hierarchy.

CARTRIDGE. On ADSM servers that support it, a device class that is used to categorize tape devices that support tape cartridges, such as the 3495 Tape Library Dataserver.

cartridge system tape (CST). The base tape cartridge media used with 3480 or 3490 Magnetic Tape Subsystems. When specified as a media type in ADSM, CST identifies standard length tape. Contrast with *enhanced capacity cartridge system tape*.

central scheduler. A function that allows an administrator to schedule client operations and administrative commands. The operations can be scheduled to occur periodically or on an explicit date. See *client schedule* and *administrative command schedule*.

CID. Configuration Installation and Distribution.

client. A program running on a PC, workstation, file server, LAN server, or mainframe that requests services of another program, called the server. There are three types of ADSM clients: administrative, backup-archive, and space management. See *administrative client*, *backup-archive client*, and *space management client*.

Client Access/400. A software product that supports advanced program-to-program communications (APPC) in the DOS, OS/2, and Microsoft Windows environments and provides a set of end user services.

client domain. The set of drives, file systems, or volumes selected by a backup-archive client user during a backup or archive operation.

client migration. The process of copying a file from a client node to ADSM storage and replacing the file with a stub file on the client node. The process is controlled by the user and by space management attributes in the management class. See also *space management*.

client node. A file server or workstation on which the backup-archive client program has been installed, which has been registered to the server.

client node definition. Server control information that includes the client's user ID, password, contact information, policy domain, file compression status, deletion authority, and whether the user ID is locked from the server. A client node definition can be exported from a source server so that it can be imported to a target server at a later date.

client node session. A period of time in which a user can communicate with a server to perform backup, archive, restore, or retrieval requests. Contrast with *administrative session*.

client options file. A file that a client can edit, containing a default set of processing options that identify the server, communication method, backup and archive options, space management options, and scheduling options. Also called the *dsm.opt* file.

client polling scheduling mode. A client/server communication technique where the client queries the server for work.

client schedule. A database record that describes the planned processing of a client operation during a specific time period. The client operation can be a backup, archive, restore, or retrieve operation, a client operating system command, or a macro. See also *administrative command schedule*.

client/server. A system architecture in which one or more programs (clients) request computing or data services from another program (server).

client system options file. A file, used on UNIX clients, containing a default set of processing options that identify the ADSM servers to be contacted for services. This file also specifies communication methods and options for backup, archive, space management, and scheduling. Also called the *dsm.sys* file. See also *client user options file*.

client user options file. A user-created file, used on UNIX clients, containing a default set of processing options that identify the server, communication method, backup and archive options, space management options, and scheduling options. Also called the *dsm.opt* file. See also *client system options file*.

closed registration. A registration process in which an administrator must register workstations as client nodes with the server. Contrast with *open registration*.

collocation. A process that attempts to keep all data belonging to a single client node or a single client file space on a minimal number of sequential access media volumes within a storage pool. The purpose of collocation is to minimize the number of volumes that must be accessed when a large amount of data must be restored.

commit. To make changes permanent in the database files. Changes made to the database files are not permanent until they are committed.

Common Programming Interface Communications (CPI-C). A programming interface that allows program-to-program communication using SNA LU6.2. See also *Systems Network Architecture*.

Common User Access (CUA). Guidelines for the dialog between a human and a workstation or terminal. One of the three Systems Application Architecture areas.

communication manager. A component of OS/2 that allows a workstation to connect to a host computer and use the host resources as well as the resources of other personal computers to which the workstation is attached, either directly or through a host.

communication method. The method used by a client and server for exchanging information.

communication protocol. A set of defined interfaces that allow computers to communicate with each other.

compression. The process of saving storage space by eliminating empty fields or unnecessary data to shorten the length of the file. In ADSM, compression can occur at a workstation before files are backed up or archived to server storage. On some types of tape drives, hardware compression can be used.

Configuration Installation and Distribution (CID). IBM's term for capabilities to automate installation. CID-enabled products are capable of unattended, remote installation.

conversion. On VM servers, the process of changing from WDSF/VM to ADSM.

copy group. A policy object that contains attributes that control the generation, destination, and expiration of backup and archive files. There are two kinds of copy groups: backup and archive. Copy groups belong to management classes. See also *frequency*, *destination*, *mode*, *serialization*, *retention*, and *version*.

copy status. The status of volume copies defined to the database or recovery log. The copy status can be synchronized, stale, off-line, or undefined.

copy storage pool. A named set of volumes that contains copies of files that reside in primary storage pools. Copy storage pools are used to back up the data stored in primary storage pools. A copy storage pool cannot be a destination for a backup copy group, an archive copy group, or a management class (for space-managed files). See *primary storage pool* and *destination*.

CPI-C. Common Programming Interface Communications.

CST. Cartridge system tape.

CUA. Common User Access.

D

daemon. In the AIX operating system, a program that runs unattended to perform a standard service. Some daemons are triggered automatically to perform their tasks; others operate periodically.

daemon process. In the AIX operating system, a process begun by the root user or by the root shell that can be stopped only by the root user. Daemon processes generally provide services that must be available at all times, such as sending data to a printer.

damaged file. A file for which ADSM has detected data-integrity errors.

DASD. Direct access storage device.

database. A collection of information about all objects managed by the server, including policy management objects, users and administrators, and client nodes.

database audit. A utility that checks for and optionally corrects inconsistent database references.

database backup series. One full backup of the database, plus up to 32 incremental backups made since that full backup. Each full backup that is run starts a new database backup series. A backup series is identified with a number.

database backup trigger. A set of criteria that defines when and how database backups are run automatically. The criteria determine how often the backup is run, whether the backup is a full or incremental backup, and where the backup is stored.

database buffer pool. Storage that is used as a cache to allow database pages to remain in memory for long periods of time, so that the server can make continuous updates to pages without requiring input or output (I/O) operations from external storage.

database dump. The action performed by the DSMSERV DUMPDB utility (DMPADSM command on AS/400), which copies ADSM database entries to media for later reload in case a catastrophic error occurs.

database load. The action performed by the DSMSERV LOADDB utility (LODADSM command on AS/400), which copies ADSM database entries from media to a newly installed database.

database volume. A volume that has been assigned to the database.

dataserver. See *Tape Library Dataserver*.

data set. See *linear data set*.

DDM. Distributed Data Management.

default management class. A management class assigned to a policy set, which is used to govern backed up or archived files when a user does not explicitly bind a file to a specific management class.

deletion exit. An installation-wide exit that informs a tape management system or operator that the server has deleted a sequential access media volume from its database.

delimiter. (1) A character used to indicate the beginning and end of a character string. (2) A character that groups or separates words or values in a line of input.

density. On MVS and VM servers, a device class attribute that identifies the bits per inch that can be stored on tape reels. ADSM supports 1600 and 6250 bits per inch (bpi).

desktop client. The group of clients supported by ADSM that include clients on OS/2, DOS, Windows, Apple, and Novell NetWare operating systems.

destination. A copy group or management class attribute that specifies the primary storage pool to which a client file will be backed up, archived, or migrated. At installation, ADSM provides storage destinations named BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL.

device class. A named group of storage devices. Each device class has a unique name and represents a device type of disk, file, optical disk, or tape.

device configuration file. A file that contains information about defined device classes, and, on some ADSM servers, defined libraries and drives. The file can be created by using an ADSM command or by using an option in the server options file. The information is a copy of the device configuration information in the ADSM database.

device driver. A collection of subroutines that control the interface between I/O device adapters and the processor.

device type. A category of storage device. Each device class must be categorized with one of the supported device types, for example, DISK or CARTRIDGE.

direct access storage device (DASD). A device in which access time is effectively independent of the location of the data.

disaster recovery. Recovery from catastrophic interruptions of computer systems, such as loss of the system location because of natural events. Backup data is kept offsite to protect against such catastrophes.

Disaster Recovery Manager (DRM). An ADSM feature that assists in preparing and later using a disaster recovery plan for the ADSM server.

disaster recovery plan. A document that contains information about how to recover computer systems if a disaster occurs. With DRM, the plan is a file that contains information about the software and hardware used by the ADSM server, and the location of recovery media.

DISK. A device class that is defined by ADSM at installation. It is used to categorize disk drives, such as 3390 DASD or 3380 DASD.

disk operating system (DOS). An operating system used in IBM PC, PS/2, and compatible computers.

Distributed Data Management (DDM). A feature of the System Support Program Product that allows an application program (client) to use server program functions to work on files that reside in a remote system.

DLL. Dynamic link library.

DLT. Digital linear tape.

domain. See *policy domain* or *client domain*.

DOS. Disk operating system.

drive. A device used to read and write data on a medium such as a disk, diskette, or tape.

DRM. Disaster Recovery Manager.

dsm.opt file. See *client options file* and *client user options file*.

dsmserv.opt. See *server options file*.

dsm.sys file. See *client system options file*.

dynamic. A copy group serialization value that specifies that ADSM accepts the first attempt to back up or archive a file regardless of whether the file is modified during the backup or archive process. See also *serialization*. Contrast with *shared dynamic*, *shared static*, and *static*.

dynamic link library. A file containing executable code and data bound to a program at load time or run time, rather than during linking. The code and data in a dynamic link library can be shared by several applications simultaneously.

E

ECCST. Enhanced capacity cartridge system tape.

enhanced capacity cartridge system tape (ECCST). Cartridge system tape with increased capacity that can only be used with 3490E tape subsystems. Contrast with *cartridge system tape*.

error log. A character file written on random access media that contains information about errors detected by the server or client.

estimated capacity. The available space, in megabytes, of a storage pool.

Ethernet. A data link protocol and LAN that interconnects personal computers and workstations via coaxial cable.

event. Administrative commands or client operations that are scheduled to be executed at a particular time.

event record. A database record that describes actual status and results for events.

exclude. The process of identifying files or directories in an include-exclude list to prevent these objects from being backed up whenever a user or schedule issues an incremental or selective backup operation, or to prevent these objects from being migrated off the client node via ADSM space management.

exclude-include list. See *include-exclude list*.

exit. To execute an instruction within a portion of a computer program in order to terminate the execution of that portion.

exit machine. On a VM server, a virtual machine that runs the mount and deletion installation-wide exits on VM systems.

expiration. The process by which files are identified for deletion because their expiration date or retention period has passed. Backed up or archived files are marked expired by ADSM based on the criteria defined in the backup or archive copy group.

expiration date. On MVS, VM, and VSE servers, a device class attribute used to notify tape management systems of the date when ADSM no longer needs a tape volume. The date is placed in the tape label so that the tape management system does not overwrite the information on the tape volume before the expiration date.

export. The process of copying administrator definitions, client node definitions, policy definitions, server control information or file data to external media.

export/import facility. See *import/export facility*.

extend. The process of increasing the portion of available space that can be used to store database or recovery log information. Contrast with *reduce*.

F

file data. File space definitions, authorization rules, backed up files, archive copies, and space-managed files. File data can be exported from a source server to external media so that it can be imported to a target server at a later date.

file record extent. The extent of the file enumerated in number of records.

file space. A logical space in a client's storage that can contain a group of files. For clients on systems such as OS/2, a file space is a logical partition and is identified by a volume label. For clients on systems such as AIX and UNIX, a file space can consist of any subset of directories and subdirectories stemming from a virtual mount point. Clients can restore, retrieve, or delete their file spaces from ADSM server storage. ADSM does not necessarily store all the files from a single file space together, but can identify all the files in server storage that came from a single file space.

File Transfer Protocol (FTP). In TCP/IP, the protocol that makes it possible to transfer data among hosts and to use foreign hosts indirectly.

format. A device class attribute that specifies the recording format used to read or write to sequential access media, for example to cartridge tape.

frequency. A copy group attribute that specifies the minimum interval, in days, between incremental backups.

FTP. File Transfer Protocol.

full backup. An ADSM function that copies the entire database. A full backup begins a new database backup series. Contrast with *incremental backup*. See *database backup series*.

fuzzy copy. A backup version or archive copy of a file that might not accurately reflect what is currently in the file because ADSM backed up or archived the file while the file was being modified.

G

GUI. Graphical user interface.

H

HDA. Head-disk assembly.

head-disk assembly (HDA). A field replaceable unit in a direct access storage device containing the disks and actuators.

hierarchical storage management (HSM) client. A program that runs on workstations to allow users to maintain free space on their workstations by migrating and recalling files to and from ADSM storage. The HSM client allows use of ADSM space management functions. Synonymous with *space management client*.

high migration threshold. A percentage of the storage pool capacity that identifies when ADSM can start migrating files to the next available storage pool in the hierarchy. Contrast with *low migration threshold*. See *server migration*.

HP-UX. Hewlett-Packard UNIX operating system. HP-UX is one of the operating systems that ADSM supports as a client environment and a server environment.

HSM client. Hierarchical storage management client.

I

import. The process of copying administrator definitions, client node definitions, policy definitions, server control information or file data from external media to a target server.

import/export facility. The facility that allows system administrators to copy definitions and file data from a source server to external media to move or copy information between servers. Any subset of information can be imported to a target server from the external media.

inactive version. A backup version of a file for which a more recently backed up version exists. Inactive backup versions are eligible for expiration processing according to the management class assigned to the file. Contrast with *active version*.

include-exclude file. On UNIX clients, a file containing statements that ADSM uses to determine whether to back up or migrate certain files, and to determine the associated management classes to use for backup, archive, and space management. See *include-exclude list*.

include-exclude list. A group of include and exclude option statements in a file. ADSM uses the statements to determine whether to back up or migrate certain files, and to determine the associated management classes to use for backup, archive, and space management. The exclude options identify files that should not be backed up or migrated off the client node. The include options identify files that are exempt from the exclusion rules, or assign a management class to a file or group of files for backup, archive, or space management services. The include-exclude list is defined either in the include-exclude file (for UNIX clients) or in the client options file (for other clients).

incremental backup. (1) A function that allows users to back up files or directories that are new or have changed since the last incremental backup. With this function, users can back up files or directories from a client domain that are not excluded in the include-exclude list and that meet the requirements for frequency, mode, and serialization as defined in the backup copy group of the management class assigned to the files. Contrast with *selective backup*. (2) An ADSM function that copies only the pages in the database that are new or changed since the last full or incremental backup. Contrast with *full backup*. See *database backup series*.

internal mounting facility. On a VM server, a VM facility that allows the server to request tape mounts by sending a message to a mount operator. The message is repeated until the tape is mounted or until the mount wait time is exceeded.

inter-user communication vehicle (IUCV) facility. On a VM server, a VM communication method used to pass data between virtual machines and VM components.

IPX/SPX. Internetwork Packet Exchange/Sequenced Packet Exchange. IPX/SPX is Novell NetWare's communication protocol.

IUCV. Inter-user communication vehicle.

K

KB. Kilobyte.

kernel. The part of an operating system that performs basic functions such as allocating hardware resources.

kernel extension. A program that modifies parts of the kernel that can be customized to provide additional services and calls. See *kernel*.

kilobyte (KB). 1024 bytes.

L

LAN. Local area network.

length. A device class attribute that specifies the length of cartridge tape by specifying one of the following media types: CST for standard length tape or ECCST for double length tape.

library. (1) A repository for demountable recorded media, such as magnetic tapes. (2) In ADSM, a collection of one or more drives, and possibly robotic devices (depending on the library type), which can be used to access storage volumes. (3) In the AS/400 system, a system object that serves as a directory to other objects. A library groups related objects, and allows the user to find objects by name.

linear data set. A type of MVS data set that ADSM uses for the database, the recovery log, and storage pools. The data set must be preallocated using VSAM IDCAMS and formatted by ADSM for its use. See *minidisk*.

load. See *mount*.

local area network (LAN). A network in which a set of devices are connected to one another for communication and that can be connected to a larger network.

log pool size. The size of an area in memory used to store recovery log pages.

logical volume. The combined space from all volumes defined to either the database or the recovery log. In ADSM, the database is one logical volume and the recovery log is one logical volume.

low migration threshold. A percentage of the storage pool capacity that specifies when ADSM can stop the migration of files to the next storage pool. Contrast with *high migration threshold*. See *server migration*.

M

machine information. Details about the machine on which a client node resides.

macro file. An optional file that contains one or more administrative commands and is invoked from an administrative client.

management class. A policy object that users can bind to each file to specify how the server manages the file. The management class can contain a backup copy group, an archive copy group, and space management attributes. The copy groups determine how the ADSM server manages backup versions or archive copies of files. The space management attributes determine whether files are eligible for migration from client nodes to ADSM storage, and under what conditions. See also *copy group*, *binding* and *rebinding*.

mask. A pattern of characters that controls the keeping, deleting, or testing of positions of another pattern of characters or bits.

maximum extension. Specifies the maximum amount of storage space, in megabytes, that you can extend the database or recovery log.

maximum reduction. Specifies the maximum amount of storage space, in megabytes, that you can reduce the database or recovery log.

maximum utilization. The highest percentage of assigned capacity used by the database or recovery log.

MB. Megabyte.

megabyte (MB). (1) For processor storage and real and virtual memory, 2^{20} or 1 048 576 bytes. (2) For disk storage capacity and transmission rates, 1 000 000 bytes.

migrate. (1) To move data from one storage pool to the storage pool specified as the next pool in the hierarchy. The process is controlled by the high and low migration thresholds for the first storage pool. See *high migration threshold* and *low migration threshold*. (2) To copy a file from a client node to ADSM storage. ADSM replaces the file with a stub file on the client node. The process is controlled by the include-exclude list and by space management attributes in management classes.

migration. The process of moving data from one storage location to another. See *client migration* and *server migration*.

minidisk. A logical subdivision of a VM physical disk that provides storage on contiguous cylinders of DASD. On a VM server, a minidisk can be defined as a disk volume that can be used by the database, recovery log, or a storage pool. See also *linear data set*.

mirroring. A feature that protects against data loss within the database or recovery log by writing the same data to multiple disks at the same time. Mirroring supports up to three exact copies of each database or recovery log volume.

mm. Millimeter.

mode. A copy group attribute that specifies whether to back up a file that has not been modified since the last time the file was backed up. See *modified* and *absolute*.

modified. A backup copy group mode value indicating that a file is considered for incremental backup only if it has changed since the last backup. A file is considered changed if the date, size, owner, or permissions have changed. See *mode*. Contrast with *absolute*.

Motif. A graphical user interface that performs window management and contains a high level toolkit for application program development. It provides an icon view of the UNIX file system. Also known as X-Windows/Motif or Motif X—Toolkit.

mount. To place a data medium (such as a tape cartridge) on a drive in a position to operate.

mount exit. On a VM server, an installation-wide exit (DSMMOUNT EXEC) that requests tape mounts on behalf of the server on VM systems.

mount limit. A device class attribute specifying the maximum number of volumes that can be simultaneously accessed from the same device class, that is, the maximum number of mount points. See *mount point*.

mount operator. On a VM server, a VM user ID that can receive tape mount messages from the server.

mount point. A logical drive through which ADSM accesses volumes in a sequential access device class. For a device class with a removable media device type (for example, CARTRIDGE), a mount point is a logical drive associated with a physical drive. For a device class with the device type of FILE, a mount point is a logical drive associated with an I/O stream. The number of mount points for a device class is determined by the mount limit for that class. See *mount limit*.

mount request. A server request to mount a sequential access media volume so that data can be read from or written to the sequential access media.

mount retention period. A device class attribute that specifies the maximum amount of time, in minutes, that the server retains a mounted sequential access media volume that is not being used before it dismounts the sequential access media volume.

mount wait period. A device class attribute that specifies the maximum amount of time, in minutes, that the server waits for a sequential access volume mount request to be satisfied before canceling the request.

Multiple Virtual Storage (MVS). One of the family of IBM operating systems for the System/370 or System/390 processor, such as MVS/ESA. MVS is one of the supported server environments.

MVS. Multiple Virtual Storage.

N

Named Pipes. A communication protocol that is built into the OS/2 operating system. It can be used to establish communications between an ADSM/2 server and OS/2 clients. The client and ADSM/2 server must reside on the same system.

NetBIOS. Network Basic Input/Output System.

network adapter. A physical device, and its associated software, that enables a processor or controller to be connected to a network.

Network Basic Input/Output System (NetBIOS). An operating system interface for application programs used on IBM personal computers that are attached to the IBM Token-Ring Network.

Network File System (NFS). A protocol defined by Sun Microsystems that extends TCP/IP network file services. NFS permits remote node files to appear as though they are stored on a local workstation.

Networking Services/DOS (NS/DOS). A software product that supports advanced program-to-program communications (APPC) in the DOS and Microsoft Windows 3.1 environments. With NS/DOS, communications applications on your workstation "talk to" partner applications on other systems that support APPC.

NFS. Network File System.

node. A unique name used to identify a workstation to the server. See also *client node.*

notify operator. A VM user ID that specifies an operator who receives messages about severe errors and abnormal conditions.

O

object. A collection of data managed as a single entity.

offsite recovery media. Media that is kept at a different location to ensure its safety if a disaster occurs at the primary location of the computer system. The media contains data necessary to recover the ADSM server and clients. The offsite recovery media manager, which is part of DRM, identifies recovery media to be moved offsite and back onsite, and tracks media status.

offsite volume. A removable media volume that is at a location where it cannot be mounted for use.

OpenEdition MVS. MVS/ESA services that support an environment within which operating systems, servers, distributed systems, and workstations share common interfaces. OpenEdition MVS supports standard application development across multivendor systems and is required to create and use applications that conform to the POSIX standard.

open registration. A registration process in which users can register their own workstations as client nodes with the server. Contrast with *closed registration*.

Operating System/2 (OS/2). An operating system used in IBM PC AT, PS/2, and compatible computers. OS/2 is one of the supported client environments and one of the supported server environments.

operator privilege class. An administrative privilege class that allows an administrator to issue commands that control the operation of the server. This privilege class allows disabling or halting the server to perform maintenance, enabling the server, canceling server processes, and managing tape.

optical library. A disk storage device that houses optical disk drives and optical disks, and contains a mechanism for moving optical disks between a storage area and optical disk drives.

OS/2. Operating System/2.

OS/400. Operating System/400.

owner. The owner of backup-archive files sent from a multiuser client node, such as AIX.

P

page. (1) A block of instructions, data, or both. (2) In ADSM, a unit of space allocation within database volumes. (3) In a virtual storage system, a fixed block that has a virtual address and is transferred as a unit between real storage and auxiliary storage.

paging. (1) The action of transferring instructions, data, or both, between real storage and external page storage. (2) Moving data between memory and a mass storage device as the data is needed.

pattern-matching expression. A string expression that uses wildcard characters to specify one or more ADSM objects. See also *wildcard character*.

PC Support/400. A software product that supports advanced program-to-program communications (APPC) in the DOS, OS/2, and Microsoft Windows environments and provides a set of end user services.

platform. The operating system environment in which a program runs.

policy definition. Server control information that includes information about policy domains, policy sets (including the ACTIVE policy set), management classes (including the default management class), copy groups, schedules, and associations between client nodes and schedules. A policy definition can be exported from a source server so that it can be imported to a target server at a later date.

policy domain. A policy object that contains policy sets, management classes, and copy groups that is used by a group of client nodes. See *policy set*, *management class*, and *copy group*.

policy privilege class. An administrative privilege class that allows an administrator to manage policy objects, register client nodes, and schedule client operations (such as backup services) for client nodes. Administrators can be authorized with unrestricted or restricted policy privilege. See *unrestricted policy privilege* or *restricted policy privilege*.

policy set. A policy object that contains a group of management class definitions that exist for a policy domain. At any one time there can be many policy sets within a policy domain but only one policy set can be active. See *management class* and *active policy set*.

premigration. For an HSM client, the process of copying files that are eligible for migration to ADSM storage, but leaving the original file intact on the local system.

primary storage pool. A named set of volumes that ADSM uses to store backup versions of files, archive copies of files, and files migrated from client nodes via ADSM space management. A primary storage pool may be backed up to a copy storage pool either automatically or by command. See *destination* and *copy storage pool*.

privilege class. A level of authority granted to an ADSM administrator. ADSM has five privilege classes: system, policy, storage, operator, and analyst. The privilege class determines which ADSM administrative tasks the administrator can perform. For example, an administrator with system privilege class can perform any administrative task.

protection status. A device class attribute that specifies whether to update the RACF profile to identify which users have access to cartridge tapes associated with this device class on MVS servers.

Q

QIC. Quarter-inch cartridge (a type of magnetic tape media).

R

random access media. Any volume accessed in a nonsequential manner. In ADSM, volumes are accessed in a nonsequential manner if they reside in the DISK device class.

randomization. The percentage of the startup window that the server can use to randomize start times for different client nodes associated with a schedule.

rebinding. The process of associating a file with a new management class name. For example, rebinding occurs when the management class associated with a file is deleted. See *binding*.

recall. A function that allows users to access files that have been migrated from their workstations to ADSM storage via ADSM space management. Contrast with *migrate*.

reclamation. A process of consolidating the remaining data from many sequential access media onto a single new sequential access media.

reclamation threshold. A value that specifies a percentage of space on sequential access media volumes that can be occupied by reclaimable space. The remainder of the space is for active data. (Space becomes reclaimable when files are expired.)

recovery log. A log of updates that are about to be written to the database. The log can be used to recover from system and media failures.

recovery log buffer pool. Used to hold new transactions records until they can be written to the recovery log.

recovery media. Media that contains data necessary to recover the ADSM server and clients.

reduce. The process of freeing up enough space to allow you to delete a volume from the database or recovery log. Contrast with *extend*.

REEL. On ADSM servers that support it, a device class that is used to categorize tape devices that support tape reels, such as the 3420 9-track tape device.

register. (1) Define a client node or administrator who can access the server. See *registration.* (2) Specify licenses that have been purchased for the server.

registration. The process of identifying a client node or administrator to the server.

reply operator. On a VM server, a VM user ID that specifies an operator who will reply to tape mount requests by the server.

restore. The process of returning a backup copy to an active storage location for use. ADSM has processes for restoring its database, storage pools, storage pool volumes, and users' backed-up files. For example, users can copy a backup version of a file from the storage pool to the workstation. The backup version in the storage pool is not affected. Contrast with *backup*.

restricted policy privilege. An administrative privilege class that enables an administrator to manage policy objects only for the policy domains for which the administrator has been authorized.

restricted storage privilege. An administrative privilege class that enables an administrator to control the allocation and use of storage resources only for the storage pools for which the administrator has been authorized.

retention. The amount of time, in days, that inactive backed up or archived files will be retained in the storage pool before they are deleted. The following copy group attributes define retention: retain extra versions, retain only version, retain version.

retention period. On an MVS server, a device class attribute that specifies how long files are retained on sequential access media. When used, ADSM passes this information to the MVS operating system to ensure that other tape management systems do not overwrite tape volumes that contain retained data.

retrieve. A function that allows users to copy an archive copy from the storage pool to the workstation. The archive copy in the storage pool is not affected. Contrast with *archive*.

rollback. To remove changes that were made to database files since the last commit point.

root. In the AIX and UNIX environments, the user name for the system user with the most authority.

root user. In the AIX and UNIX environments, an expert user who can log in and execute restricted commands, shut down the system, and edit or delete protected files. Also called the *superuser*.

S

schedule. A database record that describes scheduled client operations or administrative commands. See *administrative command schedule* and *client schedule*.

scheduling mode. The type of scheduling operation set for the server and client. ADSM supports two scheduling modes for client operations: client-polling and server-prompted.

scratch volume. A volume that is available for ADSM use. The volume is labeled, is either blank or contains no valid data, and is not defined to ADSM.

SCSI. Small computer system interface.

selective backup. A function that allows users to back up specific files or directories from a client domain. With this function, users can back up files or directories that are not excluded in the include-exclude list and that meet the requirement for serialization as defined in the backup copy group of the management class assigned to the files. Contrast with *incremental backup*.

sequential access media. Any volume that is accessed in a sequential manner, as opposed to a random manner. In ADSM, volumes are accessed sequentially if they reside in a device class other than DISK.

serialization. A copy group attribute that specifies what ADSM does if files are modified during back up or archive processing. The value of this attribute determines whether processing continues, is retried, or is stopped. See *static*, *dynamic*, *shared static*, and *shared dynamic*.

server. The program that provides backup, archive, space management, and administrative services to clients. The server program must be at the necessary level to provide all of these services.

server migration. The process of moving data from one storage pool to the next storage pool as controlled by the high and low migration thresholds. See *high migration threshold* and *low migration threshold*.

server options file. A file that specifies processing options for communication methods, tape handling, pool sizes, language, and date, time, and number formats.

server-prompted scheduling mode. A client/server communication technique where the server contacts the client when work needs to be done.

server storage. The primary and copy storage pools used by the server to store users' files: backup versions, archive copies, and files migrated from client nodes (space-managed files). See *primary storage pool*, *copy storage pool*, *storage pool volume*, and *volume*.

session resource usage. The amount of wait time, CPU time, and space used or retrieved during a client session.

shared dynamic. A copy group serialization value that specifies that a file must not be modified during a backup or archive operation. ADSM attempts to retry the backup or archive operation a number of times; if the file is in use during each attempt, ADSM will back up or archive the file on its last try even though the file is in use. See also *serialization*. Contrast with *dynamic*, *shared static*, and *static*.

shared static. A copy group serialization value that specifies that the file must not be modified during backup or archive. ADSM will retry the backup or archive operation a number of times; if the file is in use during each attempt, ADSM will not back up or archive the file. See also *serialization*. Contrast with *dynamic*, *shared dynamic*, and *static*.

shell. In the AIX and UNIX environments, a software interface between a user and the operating system of a computer. Shell programs interpret commands and user interactions on devices such as keyboards, pointing devices, and touch-sensitive screens and communicate them to the operating system.

SMIT. System Management Interface Tool.

SNA LU6.2. Systems Network Architecture Logical Unit 6.2.

socket. (1) An endpoint for communication between processes or applications. (2) A pair consisting of TCP port and IP address, or UDP port and IP address.

space-managed file. A file that is migrated from and recalled to a client node via ADSM space management.

space management. The process of keeping sufficient free storage space available on a client node by migrating files to ADSM storage. The files are migrated based on criteria defined in management classes to which files are bound, and the include-exclude list. Synonymous with *hierarchical storage management*. See also *migration*.

space management client. A program that runs on workstations to allow users to maintain free space on their workstations by migrating and recalling files to and from ADSM storage. Synonymous with *hierarchical storage management client*.

SPACEMGPOOL. A disk storage pool defined by ADSM at installation. It can be the destination for files that are migrated from client nodes via ADSM space management. See *storage pool*.

stale copy status. Specifies that a volume copy is not available to the database or recovery log.

STANDARD copy group. A backup or archive copy group that is defined by ADSM at installation. See *copy group*.

STANDARD management class. A management class that is defined by ADSM at installation. See *management class*.

STANDARD policy domain. A policy domain that is defined by ADSM at installation. See *policy domain*.

STANDARD policy set. A policy set that is defined by ADSM at installation. See *policy set*.

stanza. A group of lines in a file that together have a common function or define a part of the system. Stanzas are usually separated by blank lines or colons, and each stanza has a name.

startup window. A time period during which a schedule must be initiated.

static. A copy group serialization value that specifies that the file must not be modified during backup or archive. If the file is modified during the attempt, ADSM will not back up or archive the file. See also *serialization*. Contrast with *dynamic*, *shared dynamic*, and *shared static*.

storage hierarchy. A logical ordering of primary storage pools, as defined by an administrator with system privilege. Generally, the ordering is based on the speed and capacity of the devices that the storage pools use. In ADSM, the storage hierarchy is defined by identifying the *next* storage pool in a storage pool definition. See *storage pool*.

storage management services. A component that allows a central system to act as a file backup and archive server for local area network file servers and workstations.

storage pool. A named set of storage volumes that ADSM uses to store client data. A storage pool is either a primary storage pool or a copy storage pool. See *primary storage pool* and *copy storage pool*.

storage pool volume. A volume that has been assigned to an ADSM storage pool. See *volume*, *copy storage pool*, and *primary storage pool*.

storage privilege class. An administrative privilege class that allows an administrator to control the allocation and use of storage resources for the server, such as monitoring the database, recovery log, and server storage. Administrators can be authorized with unrestricted or restricted storage privilege. See *restricted storage privilege* or *unrestricted storage privilege*.

stub file. A file that replaces the original file on a client node when the file is migrated from the client node to ADSM storage.

superuser. See *root user*.

synchronized copy status. Specifies that the volume is the only volume copy or is synchronized with other volume copies in the database or recovery log. When synchronized, mirroring has started.

system privilege class. An administrative privilege class that allows an administrator to issue all server commands.

Systems Application Architecture (SAA). Software interfaces, conventions, and protocols that provide a framework for designing and developing applications that are consistent across systems.

Systems Network Architecture (SNA). A set of rules for data to be transmitted in a network. Application programs communicate with each other using a layer of SNA called advanced program-to-program communications (APPC).

T

tape library. (1) A term used to refer to a collection of tape cartridges. (2) An automated device that performs tape cartridge mounts and demounts without operator intervention.

Tape Library Dataserver. An automated tape library consisting of mechanical components, cartridge storage frames, IBM tape subsystems, and controlling hardware and software. The tape library dataserver performs tape cartridge mounts and demounts without operator intervention.

tape volume prefix. A device class attribute that is the high-level-qualifier of the file name or the data set name in the standard tape label.

TCP/IP. Transmission Control Protocol/Internet Protocol.

Telnet. In TCP/IP, the protocol that opens the connection to the system.

Transmission Control Protocol/Internet Protocol (TCP/IP). A set of communication protocols that support peer-to-peer connectivity functions for both local and wide area networks.

trusted communication agent. A program that performs communication tasks on behalf of the client or server, and ensures the security of the communications.

U

unit name. On an MVS server, a device class attribute that specifies a group of tape devices used with the MVS server. A unit name can be a generic device type, an esoteric unit name, or a physical device.

unrestricted policy privilege. An administrative privilege class that enables an administrator to manage policy objects for any policy domain.

unrestricted storage privilege. An administrative privilege class that enables an administrator to control the database, recovery log, and all storage pools.

utilization. The percent of assigned capacity used by the database or recovery log at a specific point of time.

V

validate. The process of ensuring that the active policy set contains a default management class and reporting on copy group definition errors.

version. The maximum number of backup copies retained for files and directories. The following copy group attributes define version criteria: versions data exists and versions data deleted.

Virtual Machine (VM). One of the family of IBM operating systems for the System/370 or System/390 processor, including VM/ESA, VM/XA, VM/SP, and VM/HPO. VM is one of the supported server environments.

Virtual Storage Extended (VSE). One of the family of IBM operating systems for the System/370 or System/390 processor, including VSE/ESA. VSE is one of the supported server environments.

VM. Virtual Machine.

volume. The basic unit of storage for the database, recovery log, or a storage pool. A volume can be an

LVM logical volume, a standard file system file, a tape cartridge, or an optical cartridge. Each volume is identified by a unique volume identifier. See *database volume*, *scratch volume*, and *storage pool volume*.

volume history file. A file that contains information about: volumes used for database backups and database dumps; volumes used for export of administrator, node, policy, or server data; and sequential access storage pool volumes that have been added, reused, or deleted. The information is a copy of the same types of volume information in the ADSM database.

volume set. An entire image of the database or recovery log, as displayed on the administrative graphical user interface.

VSE. Virtual Storage Extended.

W

WDSF/VM. Workstation Data Save Facility/Virtual Machine.

wildcard character. A character or set of characters used to specify an unknown number or set of characters in a search string. Also called *pattern-matching character*.

Workstation Data Save Facility/Virtual Machine (WDSF/VM). The predecessor product to ADSTAR Distributed Storage Manager.

WORM. A type of optical media that can only be written to and cannot be erased.

X

X Windows. A network transparent windowing system developed by MIT. It is the basis for other products, such as Enhanced X Windows which runs on the AIX operating system.

Index

© Copyright IBM Corp. 1995, 1997

Communicating Your Comments to IBM

ADSTAR Distributed Storage Manager
for Sun** Solaris**
Administrator's Guide
Version 2

Publication No. GC35-0263-00

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

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

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:
  - United States and Canada: 520 799-2906
  - Other countries: (1) 520 799-2906

  The contact department is 61C/031.
- If you prefer to send comments by electronic mail, use one of the following addresses:
  - Internet: starpubs@vnet.ibm.com (or starpubs at vnet.ibm.com)
  - IBMLink from U.S.A.: STARPUBS at SJEVM5
  - IBMLink from Canada: STARPUBS at TORIBM
  - IBM Mail Exchange: USIB3VVD at IBMMAIL

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

Readers' Comments — We'd Like to Hear from You

ADSTAR Distributed Storage Manager
for Sun Solaris****
Administrator's Guide
Version 2

Publication No. GC35-0263-00

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	□	□	□	□	□

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	□	□	□	□	□
Complete	□	□	□	□	□
Easy to find	□	□	□	□	□
Easy to understand	□	□	□	□	□
Well organized	□	□	□	□	□
Applicable to your tasks	□	□	□	□	□

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? □ Yes □ No

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

Name

Address

Company or Organization

Phone No.

Readers' Comments — We'd Like to Hear from You
GC35-0263-00

Cut or Fold
Along Line

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

International Business Machines Corporation
Information Development
Department 61C
9000 South Rita Road
TUCSON AZ 85775-4401

Fold and Tape

Please do not staple

Fold and Tape

Cut or Fold
Along Line

Program Number: 5639-B22

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GC35-0263-00

Spine information:

IBM ADSTAR Distributed Storage Manager for Sun** Solaris** Administrator's Guide *Version 2*