

ADSTAR Distributed Storage Manager
for AS/400



Administrator's Guide

Version 2

Note!

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

This book is also available in a softcopy form that can be viewed with the IBM BookManager READ licensed program.

First Edition (June 1996)

This edition applies to Version 2 of the ADSTAR Distributed Storage Manager for AS/400 (5763-SV2) 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 1994, 1996. All rights reserved.**

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

Contents

Notices	xv
Programming Interface	xv
Trademarks	xvii
Preface	xix
Who Should Read This Publication	xix
What You Should Know before Reading This Publication	xix
Conventions Used in This Book	xix
ADSTAR Distributed Storage Manager Publications	xx
Related AS/400 System Publications	xxi
Related IBM Hardware Products Publications	xxii
IBM International Technical Support Center Publications (Redbooks)	xxii
Software Developer's Program	xxiii
Do You Have Comments or Suggestions?	xxiii
Translations	xxiii
Summary of Changes for ADSTAR Distributed Storage Manager	xxv
Changes for Version 2—June 1996	xxv

Part 1. ADSM Basics 1

Chapter 1. Introducing ADSTAR Distributed Storage Manager	3
How ADSM Stores Client Data	6
How ADSM Represents Devices	9
Disk Devices	9
Tape Devices	10
Files as Logical Devices	10
How ADSM Represents Storage Media	11
What are the ADSM Storage Objects?	11
Device Class	11
Library	11
Drive	12
Storage Pools	12
Storage Pool Volumes	12
What Does a Device Class Contain?	13
Device Classes for Random Access Devices	13
Device Classes for Sequential Access Devices	13
Putting It All Together	14
Planning to Configure the ADSM Storage Environment	16
Evaluating Your Storage Environment	16
Mapping Devices to Device Classes	16
Mapping Storage Pools to Device Classes and Devices	17
Configuring Devices	18
MANUAL Libraries	19

AS400MLB Libraries	19
USRDFN Libraries	20
Automating Client Operations	20
Chapter 2. Administrator Tasks	23
Using Magnetic Disk Devices with ADSM	23
Using Tape Devices with ADSM	24
Using a Tape Management System with ADSM	24
Managing Tape Operations	24
Defining Drives and Libraries	24
Defining Device Classes	25
Managing Storage Pools	25
Managing Storage Pool Volumes	25
Managing Policies	26
Automating Operations	26
Managing Server Operations	26
Managing the Database and Recovery Log	26
Managing Licensing, Privilege Classes, and Registration	27
Exporting and Importing Data	27
Protecting and Recovering Your Data	27
Task, GUI, and Command Cross-Reference	28
Using Magnetic Disk Devices	29
Using Tape Devices with ADSM	29
Using a Tape Management System	31
Managing Tape Operations	32
Defining Drives and Libraries	33
Defining Device Classes	34
Managing Storage Pools	35
Managing Storage Pool Volumes	36
Managing Policies	37
Automating ADSM Operations	39
Managing Server Operations	41
Managing the Database and Recovery Log	43
Managing Licensing, Privilege Classes, and Registration	45
Exporting and Importing Data	47
Protecting and Recovering Your Data	49

Part 2. Configuring and Managing Server Storage 53

Chapter 3. Using Magnetic Disk Devices with ADSM	55
Setting Up Storage Pools on Disk Devices	55
Using Random Access Volumes on Disk Devices	55
Using Disk for FILE Logical Devices	56
Notes on Operations	57
Using Cache	57
Freeing Space on Disk	57
Scratch FILE Volumes	57
FILE Volumes Used for Database Backups and Export Operations	57

Chapter 4. Using Tape Devices with ADSM	59
Configuring the Devices	59
Summary: Configuring Devices	60
Configuring Devices for ADSM Using the Device Configuration Utility	62
Example of a Manual Library: Setting Up Two 3490E Drives	69
Example of an Automated Library: Setting up a 9427 Library	73
Configuring IBM 3590 Drives	79
Configuring non-IBM Devices	79
Notes on Operations	79
Mount Operations for Manual Libraries	80
Handling Messages for Automated Libraries	80
Drive Selection	80
Collocation	81
Using Automatic Cartridge Loaders	81
Maintaining the Volume Inventory	81
Managing Storage Volumes in Automated Libraries	83
Using Categories for Volumes in Automated Libraries	84
Private and Scratch Volumes in Automated Libraries	85
Informing the Server about New Volumes in a Library	86
Changing the Status of a Volume in a Library	87
Removing Volumes from a Library	87
Returning Volumes to a Library	88
Auditing a Library's Volume Inventory	88
Chapter 5. Using a Tape Management System with ADSM	89
The Exits	89
Creating an Exit Program	90
Setting Up to Use a Tape Management System	90
A Detailed Example	91
Preparing the Exit Programs	92
Defining the Exit Programs	92
Defining the Libraries	93
Defining the Drives	94
Defining the Device Classes	94
Defining the Storage Pools	95
Defining and Managing Exit Programs	96
Defining an Exit Program to ADSM	96
Updating an Exit Program	96
Querying an Exit Program	96
Deleting an Exit Program	97
Notes on Operations	97
Handling Messages	97
Drive Selection	98
Collocation	98
Reusing Tapes in Storage Pools	98
Maintaining the Exits	99

Chapter 6. Managing Tape Operations	101
Requesting Information About Pending Operator Requests	101
Replying to Operator Requests	103
Cancelling an Operator Request	104
Determining Which Volumes are Mounted	104
Dismounting an Idle Volume	105
Chapter 7. Defining Drives and Libraries	107
How ADSM Uses Sequential Access Devices	108
Defining and Managing Libraries	110
Defining Libraries	110
Requesting Information about Libraries	111
Updating Libraries	111
Deleting Libraries	112
Defining and Managing Drives	112
Defining Drives	112
Requesting Information about Drives	113
Updating Drives	114
Deleting Drives	114
Chapter 8. Defining Device Classes	115
Defining and Updating Device Classes for Sequential Media	116
Defining and Updating Device Classes for Tape	116
Defining and Updating FILE Device Classes	122
Requesting Information about a Device Class	123
Deleting a Device Class	124
Filling Volumes to Capacity	124
Tape Volume Capacity and Data Compression	125
Recovering from Error ANR8263W	127
Chapter 9. Managing Storage Pools	131
Storage Pools	132
An Example of Server Storage	133
Assigning Volumes to Storage Pools	135
Assigning Random Access Storage Pool Volumes	135
Assigning Sequential Access Storage Pool Volumes	135
Storage Pool Hierarchy	135
How ADSM Stores Files in a Storage Pool Hierarchy	136
How the Storage Hierarchy Affects Planning for Copy Storage Pools	138
Using the Hierarchy to Stage Client Data from Disk to Tape	138
Server Migration of Files	139
Migration Thresholds for Disk Storage Pools	139
Migration Thresholds for Sequential Access Storage Pools	142
Migration and Copy Storage Pools	143
The Use of Cache on Disk Storage Pools	143
Why Use Cache?	143
When Not to Use Cache	144
Collocation on Sequential Access Storage Pools	144

How the Server Selects Volumes with Collocation Enabled	147
How the Server Selects Volumes with Collocation Disabled	148
Turning Collocation On or Off	148
Collocation on Copy Storage Pools	149
Space Reclamation for Sequential Access Storage Pools	149
Choosing a Reclamation Threshold	150
Reclamation for Copy Storage Pools	151
How Collocation Affects Reclamation	153
Reclamation in a Single-Drive Library	153
Expiration Processing	154
Delaying Reuse of Sequential Access Volumes	154
How Restore Processing Works	155
Estimating Space Needs for Storage Pools	156
Estimating Space Needs in Random Access Storage Pools	157
Estimating Space Needs in Sequential Access Storage Pools	158
Defining or Updating Storage Pools	159
Defining a Primary Storage Pool	159
Defining a Copy Storage Pool	162
Backing Up Storage Pools	163
Using Copy Storage Pools to Improve Data Availability	165
Example: Simple Hierarchy with One Copy Storage Pool	165
Monitoring the Use of Storage Pools	166
Monitoring the Use of Storage Pool Space	166
Monitoring Migration Processes	167
Monitoring the Use of Cache Space on Disk Storage	171
Requesting Information on Storage Occupancy	172
Deleting a Storage Pool	175
Restoring Storage Pools	176
What Happens When a Storage Pool Is Restored	177
When a Storage Pool Restoration is Incomplete	178
Chapter 10. Managing Storage Pool Volumes	179
Storage Pool Volumes	179
Access Modes for Storage Pool Volumes	180
Preparing Volumes for Random Access Storage Pools	181
Preparing Volumes for Sequential Access Storage Pools	181
Labeling Sequential Storage Pool Volumes	181
Overwriting Existing Volume Labels	182
Defining Storage Pool Volumes	183
Updating Storage Pool Volumes	183
Monitoring the Use of Storage Pool Volumes	185
Requesting General Information about Storage Pool Volumes	185
Requesting Detailed Information about Storage Pool Volumes	185
Requesting Information about Storage Pool Volume Contents	187
Auditing a Storage Pool Volume	189
What Happens When You Audit Storage Pool Volumes	190
Auditing a Volume in a Disk Storage Pool	191
Auditing Multiple Volumes in a Sequential Access Storage Pool	192

Auditing a Single Volume in a Sequential Access Storage Pool	193
Moving Files from One Volume to Another Volume	193
Moving Data to Other Volumes in the Same Storage Pool	194
Moving Data to Another Storage Pool	194
Moving Data from an Offsite Volume in a Copy Storage Pool	195
Procedure for Moving Data	195
Deleting Storage Pool Volumes	197
Deleting an Empty Storage Pool Volume	197
Deleting a Storage Pool Volume with Data	198
Restoring Storage Pool Volumes	199
What Happens When a Volume Is Restored	200
When a Volume Restoration is Incomplete	200

Part 3. Policies 201

Chapter 11. Managing Policies	203
Operations Controlled by Policy	204
Backup and Restore	204
Archive and Retrieve	204
Migration and Recall	204
Policy Objects	205
Management Classes	207
Management Class Configuration	207
Default Management Classes	208
The Include-Exclude List	208
How Files Are Associated with a Management Class	209
File Eligibility for Policy Operations	212
Incremental Backup	212
Selective Backup	214
Archive	214
Automatic Migration from a Client Node	215
How Client Migration Works with Backup and Archive	215
Using the Standard Storage Management Policies	216
Creating Your Own Storage Management Policies	217
Defining and Updating a Policy Domain	220
Defining and Updating a Policy Set	221
Defining and Updating a Management Class	222
Defining and Updating a Backup Copy Group	223
Defining and Updating an Archive Copy Group	227
Assigning a Default Management Class	229
Validating and Activating Policy Sets	229
Activating Policy Sets	230
Running Expiration Processing to Delete Expired Files	231
Querying Policy Objects	231
Querying Copy Groups	232
Querying Management Classes	232
Querying Policy Sets	233
Querying Policy Domains	233

Deleting Policy Objects	234
Deleting Copy Groups	234
Deleting Management Classes	235
Deleting Policy Sets	235
Deleting Policy Domains	235

Part 4. Automating Operations 237

Chapter 12. Automating Operations	239
Automating Server Operations	240
Defining the Schedule	240
Verifying the Schedule	241
Automating Client Operations	241
Defining the Client Schedule	242
Associating Client Nodes with Schedules	242
Starting the Scheduler on the Clients	243
Verifying the Schedule	243
Coordinating Client Schedules	244
Setting the Scheduling Mode	245
Specifying the Schedule Period for Incremental Backup Operations	246
Controlling the Server's Scheduled Workload	246
Controlling Contact with the Server	248
Tailoring Schedules	250
Common Schedule Parameters	250
Specifying Administrative Command Schedule Parameters	251
Specifying Client Schedule Parameters	252
Copying Schedules	254
Deleting Schedules	255
Managing Scheduled Event Records	255
Querying Event Records	255
Removing Event Records from the Database	257
Managing Client Associations with Schedules	258
Querying Associations	258
Deleting Associations	259

Part 5. Maintaining the Server 261

Chapter 13. Managing Server Operations	263
Starting, Halting, and Restarting the Server	263
Starting the Server	263
Halting the Server	264
Restarting the Server	265
Managing Client Sessions	265
Requesting Information about Client Sessions	265
Canceling a Client Session	267
Disabling or Enabling Server Access	267
Managing Server Processes	268

Requesting Information about Server Processes	269
Canceling Server Processes	269
Varying Disk Volumes Online or Offline	270
Requesting Information about Server Status	270
Setting the Server Name	271
Querying Server Options	271
Managing the Activity Log	272
Changing the Size of the Activity Log	272
Setting the Activity Log Retention Period	273
Requesting Information from the Activity Log	273
Monitoring Accounting Records	274
Getting Help on Commands and Error Messages	275
Chapter 14. Managing the Database and Recovery Log	277
Database and Recovery Log	277
How ADSM Processes Transactions	278
How Space is Managed by the Server	278
Estimating and Monitoring Database and Recovery Log Space Requirements	280
Monitoring the Database and Recovery Log	280
Adding Space to the Database or Recovery Log	282
Step 1: Allocating Space for the Database and Recovery Log	282
Step 2: Defining Database or Recovery Log Volumes to ADSM	283
Step 3: Extending the Capacity of the Database or Recovery Log	285
Deleting Space from the Database or Recovery Log	286
Step 1: Determining If Volumes Can Be Deleted	286
Step 2: Reducing the Capacity of the Database or Recovery Log	288
Step 3: Deleting a Volume from the Database or Recovery Log	288
Optimizing the Performance of the Database or Recovery Log	290
Adjusting the Database Buffer Pool	290
Adjusting the Recovery Log Buffer Pool	292
Chapter 15. Managing Licensing, Privilege Classes, and Registration	295
Managing ADSM Licenses	295
Licensed Features	296
Licensing Example	298
License Compliance	298
Monitoring Licenses	298
Ensuring Client/Server Authentication	299
Setting Password Authentication	299
Setting User Password Expiration	299
Registering Administrators or Updating Information	300
Granting Administrative Authority	300
System Privilege	301
Unrestricted Policy Privilege	301
Restricted Policy Privilege	302
Unrestricted Storage Privilege	302
Restricted Storage Privilege	303
Operator Privilege	303

Analyst Privilege	304
Changing Administrative Authority	304
Extending Administrative Privilege	304
Revoking One or More Administrative Privilege Classes	305
Revoking All Administrative Privilege Classes	305
Reducing Privilege Classes	305
Managing Administrator Access	305
Renaming an Administrator	306
Removing Administrators	306
Locking and Unlocking Administrators from the Server	307
Managing Client Nodes	308
Setting Client Node Registration	308
Managing Client Node Access	310
Requesting Information about Client Nodes	311
Requesting Information about File Spaces	313
Deleting File Spaces and Client Nodes	314
Registering an Application Programming Interface to the Server	315
Understanding How the Compression Option is Set	315
Understanding How the File Deletion Option is Set	316
Chapter 16. Exporting and Importing Data	317
Data That Can Be Exported and Imported	317
Preparing to Export or Import Data	318
Using Preview before Exporting or Importing Data	318
Planning for Sequential Media Used to Export Data	319
Monitoring Export and Import Processes	320
Requesting Information about an Export or Import Process	320
Viewing Information from an Administrative Client	321
Querying the Activity Log for Export or Import Information	323
Exporting Data to Sequential Media Volumes	324
Deciding When to Export Data	325
Exporting Server Data	326
Exporting Administrator Information	326
Exporting Client Node Information	327
Exporting Policy Information	328
Importing Data from Sequential Media Volumes	328
Step 1: Previewing Information before You Import Data	329
Step 2: Importing Definitions	331
Step 3: Tailoring Server Storage Definitions on the Target Server	333
Step 4: Importing File Data Information	334
Considerations When Importing Data	336
Recovering from Errors during the Import Process	337

Part 6. Protecting the Server 339

Chapter 17. Protecting and Recovering Your Data	341
Levels of Protection	341
Storage Pool Protection	342

Database and Recovery Log Protection	342
An Overview of the Process	344
Backing Up Storage Pools	345
Mirroring the Database and Recovery Log	347
Allocating Volume Copies to Auxiliary Storage Pools	347
Defining Database or Recovery Log Mirrored Volumes	347
Requesting Information about Mirrored Volumes	348
Backing Up the Database	349
Defining Device Classes for Backups	349
Setting the Recovery Log Mode	350
Scheduling Database Backups	350
Estimating the Size of the Recovery Log	351
Setting a Database Backup Trigger	352
Saving the Volume History File	354
Saving the Device Configuration Backup File	355
Doing Full and Incremental Backups	357
Recovering by Using Mirrored Volumes	358
Recovering by Using Database and Storage Pool Backups	358
Restoring a Database to a Point in Time	359
Restoring a Database to its Most Current State	362
Correcting Damaged Files	363
Maintaining the Integrity of Files	363
Restore Damaged Files	364
Backup and Recovery Scenarios	365
Protecting Your Database and Storage Pool	365
Recovering to a Point in Time from a Disaster	367
Recovering a Lost or Damaged Storage Pool Volume	369

Part 7. Appendix, Glossary, and Index 371

Appendix. Interface for Media Management Systems	373
Mount Exit Program	373
Required Parameter Group	373
Format of Mount Information	373
Mount Information Field Descriptions	374
Format of Completion Information	376
Completion Information Field Descriptions	377
Error Messages	377
Dismount Exit Program	378
Required Parameter Group	378
Format of Dismount Information	378
Dismount Information Field Descriptions	379
Format of Completion Information	381
Completion Information Field Descriptions	381
Error Messages	381
Deletion Exit Program	382
Required Parameter Group	382
Format of Deletion Information	382

Deletion Information Field Descriptions	383
Format of Completion Information	384
Completion Information Field Descriptions	384
Error Messages	384
Expiration Exit Program	385
Required Parameter Group	385
Format of Expiration Information	385
Expiration Information Field Descriptions	386
Format of Completion Information	387
Completion Information Field Descriptions	388
Error Messages	388
Glossary	389
Index	405

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, Product-sensitive 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 _____

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

The following terms are trademarks of other companies:

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

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 AS/400: Licensed Program Specifications*. For information on installing ADSM, see *ADSTAR Distributed Storage Manager for AS/400: 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 an AS/400 command line:

```
===> wrkcfgsts *dev *tap
```

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

```
query devclass
```

- Command to be entered on an OS/2 command line:

```
> dsmadm -mountmode
```

ADSTAR Distributed Storage Manager Publications

The ADSM library is available in softcopy on the following CD-ROMs:

Title	Order Number
ADSM Online Library	SK2T-1878
AS/400 Softcopy Library	SK2T-2171

The following table lists ADSM publications.

Short Title	Publication Title	Order Number
ADSM General Information	<i>ADSTAR Distributed Storage Manager: General Information</i>	GH35-0131
ADSM Messages	<i>ADSTAR Distributed Storage Manager: Messages</i>	SH35-0133
ADSM Licensed Program Specifications	<i>ADSTAR Distributed Storage Manager for AS/400: Licensed Program Specifications</i>	GH35-0139
ADSM Quick Start	<i>ADSTAR Distributed Storage Manager for AS/400: Quick Start</i>	GA32-0357
ADSM Administrator's Reference	<i>ADSTAR Distributed Storage Manager for AS/400: Administrator's Reference</i>	SC35-0197
ADSM Using the UNIX HSM Clients	<i>ADSTAR Distributed Storage Manager: Using the UNIX Hierarchical Storage Management Clients</i>	SH26-4030
ADSM V2 Using the Apple Macintosh Backup-Archive Client	<i>ADSTAR Distributed Storage Manager Version 2: Using the Apple Macintosh Backup-Archive Client</i>	SH26-4051
ADSM Using the UNIX Backup-Archive Clients	<i>ADSTAR Distributed Storage Manager Version 2: Using the UNIX Backup-Archive Clients</i>	SH26-4052
ADSM V2 Using the OS/2 Backup-Archive Client	<i>ADSTAR Distributed Storage Manager Version 2: Using the OS/2 Backup-Archive Client</i>	SH26-4053
ADSM V2 Using the DOS Backup-Archive Client	<i>ADSTAR Distributed Storage Manager Version 2: Using the DOS Backup-Archive Client</i>	SH26-4054

Short Title	Publication Title	Order Number
ADSM V2 Using the Novell NetWare Backup-Archive Client	<i>ADSTAR Distributed Storage Manager Version 2: Using the Novell NetWare Backup-Archive Client</i>	SH26-4055
ADSM V2 Using the Microsoft Windows Backup-Archive Clients	<i>ADSTAR Distributed Storage Manager Version 2: Using the Microsoft Windows Backup-Archive Clients</i>	SH26-4056
ADSM Using the Lotus Notes Backup Agent	<i>ADSTAR Distributed Storage Manager: Using the Lotus Notes Backup Agent</i>	SH26-4047
ADSM Installing the Clients	<i>ADSTAR Distributed Storage Manager: Installing the Clients</i>	SH26-4049
ADSM Client Reference Cards	<i>ADSTAR Distributed Storage Manager: Client Reference Cards</i>	SX26-6013
ADSM Using the Application Programming Interface	<i>ADSTAR Distributed Storage Manager: Using the Application Programming Interface</i>	SH26-4002

Related AS/400 System Publications

The following table lists related AS/400 publications.

Publication Title	Order Number
<i>AS/400 Automated Tape Library Planning and Management</i>	SC41-3309
<i>AS/400 Local Device Configuration</i>	SC41-3121
<i>AS/400 Publications Reference</i>	SC41-3003
<i>Backup Recovery and Media Services for OS/400</i>	SC41-3345
<i>OS/400 CL Reference</i>	SC41-3722
<i>OS/400 Backup and Recovery – Basic</i>	SC41-3304
<i>OS/400 Backup and Recovery – Advanced</i>	SC41-3305
<i>OS/400 Communications Configuration</i>	SC41-3401
<i>OS/400 Remote Work Station Support</i>	SC41-3402
<i>OS/400 Work Management</i>	SC41-3306

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	<i>IBM 3490 Magnetic Tape Subsystem Enhanced Capability Models E01 and E11 User's Guide</i>	GA32-0298
IBM 3494 Media Library Device Driver for AS/400 User's Guide	<i>IBM 3494 Tape Library Dataserver User's Guide: Media Library Device Driver for AS/400</i>	GC35-0153
IBM 3494 Operator's Guide	<i>IBM 3494 Tape Library Dataserver Operator's Guide</i>	GA32-0280
IBM 3590 Tape Subsystem User's Guide	<i>IBM 3590 High Performance Tape Subsystem User's Guide</i>	GA32-0330
IBM 3495 Operator's Guide	<i>IBM 3495 Tape Library Dataserver Models L20, L30, L40, and L50 Operator's Guide</i>	GA32-0235

IBM International Technical Support Center Publications (Redbooks)

The following table lists International Technical Support Center (ITSC) publications:

Title	Order Number
<i>ADSM Version 2 Presentation Guide</i>	SG24-4532
<i>ADSM Implementation Examples</i>	GG24-4034
<i>ADSM Advanced Implementation Examples</i>	GG24-4221
<i>Getting Started with ADSM/2</i>	GG24-4321
<i>Getting Started with ADSM/6000</i>	GG24-4421
<i>Getting Started with the NetWare Client</i>	GG24-4242
<i>Getting Started with the AIX/6000 Client</i>	GG24-4243
<i>ADSM API Examples for OS/2 and Windows</i>	SG24-2588
<i>Using ADSM to Back Up Databases</i>	GG24-4335
<i>AIX Storage Management</i>	GG24-4484
<i>Easy Access to Host Data with Distributed File Manager</i>	GG24-4427
<i>ADSM/VSE Implementation</i>	GG24-4266
<i>AIX Storage Management</i>	GG24-4484
<i>ADSM/6000 on 9076 SP2</i>	GG24-4499

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 SJSVM28
- 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: 520 799-6487
- Fax from other countries: (1) 520 799-6487

Translations

Selected ADSM publications have been translated into languages other than American English. For a complete list of the available translations and their order numbers, see *ADSM General Information*. 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—June 1996

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 17, "Protecting and Recovering Your Data" on page 341 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 9, "Managing Storage Pools" on page 131 for details.

Administrative command scheduling

You can define schedules for automatically issuing administrative commands once or periodically. See Chapter 12, "Automating Operations" on page 239 for details.

Configuration and administration enhancements

You can use the ADSM utilities interface to make ADSM configuration and administration tasks easier. Select the Utilities option on the ADSM menu. See "Configuring Devices for ADSM Using the Device Configuration Utility" on page 62 for an example.

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 11, "Managing Policies" on page 203 for details.

Device support enhancements

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.

See "Managing Storage Volumes in Automated Libraries" on page 83 for details.

Part 1. ADSM Basics

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 client

Provides space management services for workstations. 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.

Figure 1 on page 4 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 or tape volumes in server storage, which can be grouped into storage pools.

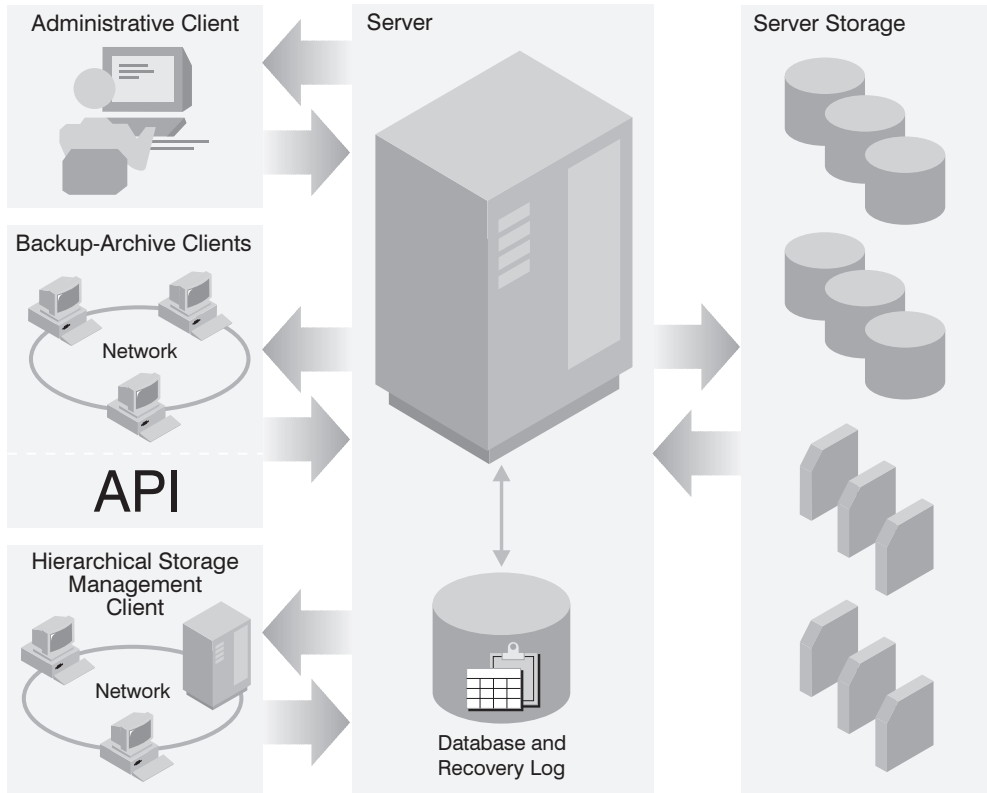


Figure 1. Sample Client/Server Environment

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.

Section	Page
Concepts:	
How ADSM Stores Client Data	6
How ADSM Represents Devices	9
How ADSM Represents Storage Media	11
What are the ADSM Storage Objects?	11
Putting It All Together	14
Planning to Configure the ADSM Storage Environment	16
Configuring Devices	18
Automating Client Operations	20

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. 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 216.

Figure 2 on page 7 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 or tape 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.

See Chapter 11, “Managing Policies” on page 203 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 14, “Managing the Database and Recovery Log” on page 277 for information on managing the database.

3 Stores the file in ADSM server storage

ADSM stores backup-archive client files and HSM client files on disk or tape volumes. These media are associated with ADSM storage pools.

For information about storage pools and storage pool volumes, see Chapter 9, “Managing Storage Pools” on page 131 and “Storage Pool Volumes” on page 179.

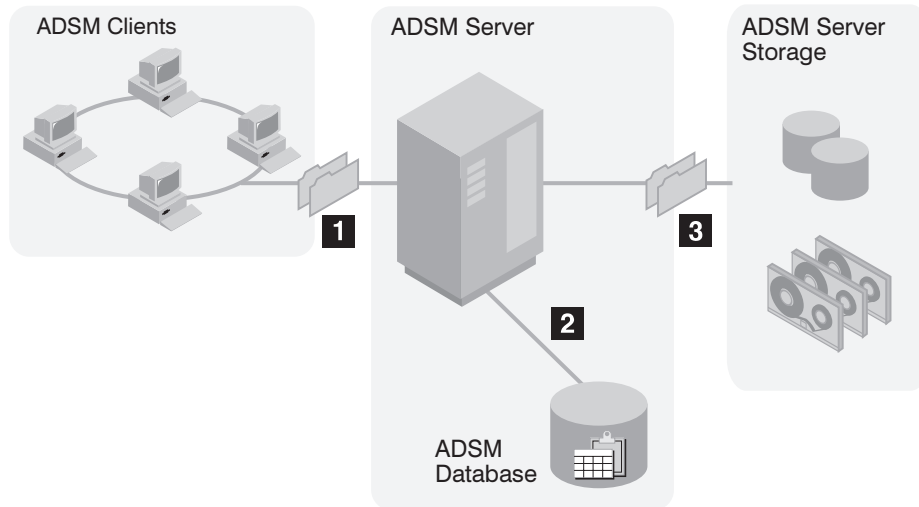


Figure 2. Overview of How ADSM Stores Client Data

Figure 3 shows in more detail the interaction between ADSM policy objects and ADSM backup, archive, and migration services.

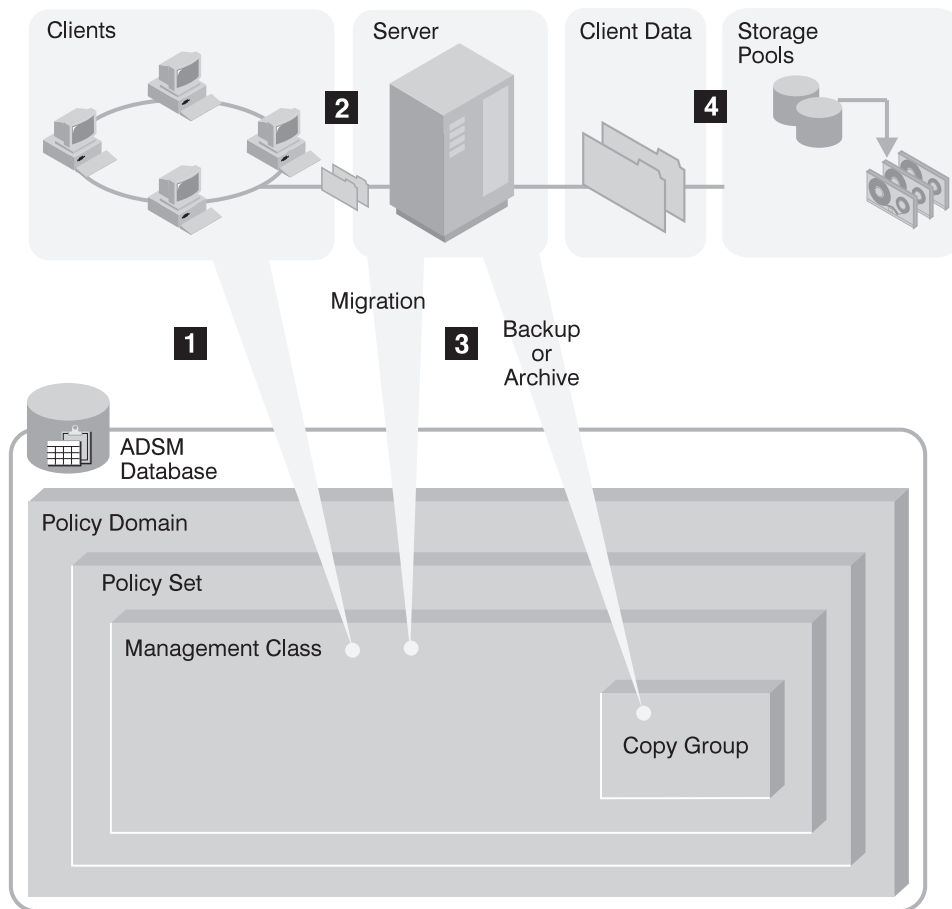


Figure 3. How ADSM Controls Backup, Archive, and Migration

- 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 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. Whether the device accesses the data on its media randomly or sequentially is the key factor in determining whether a library and drive object are also defined. Sequential devices (such as tape) require a library and at least one drive specification.

Disk Devices

Magnetic disk devices are the only devices in the random access category so they all share the same ADSM device type: DISK. ADSM predefines the DISK device class so administrators need only define storage volumes.

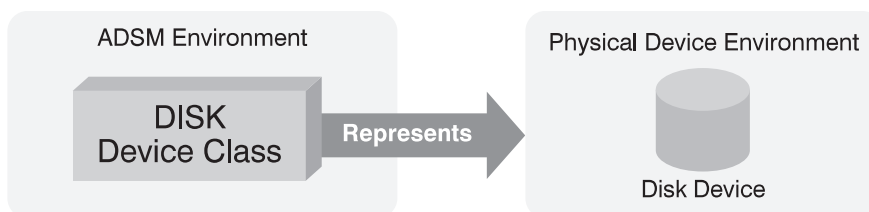


Figure 4. Magnetic Disk Devices Are Represented by Only a Device Class

Tape Devices

Figure 5 shows that a tape 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 an automated library type for devices loaded by a robot.

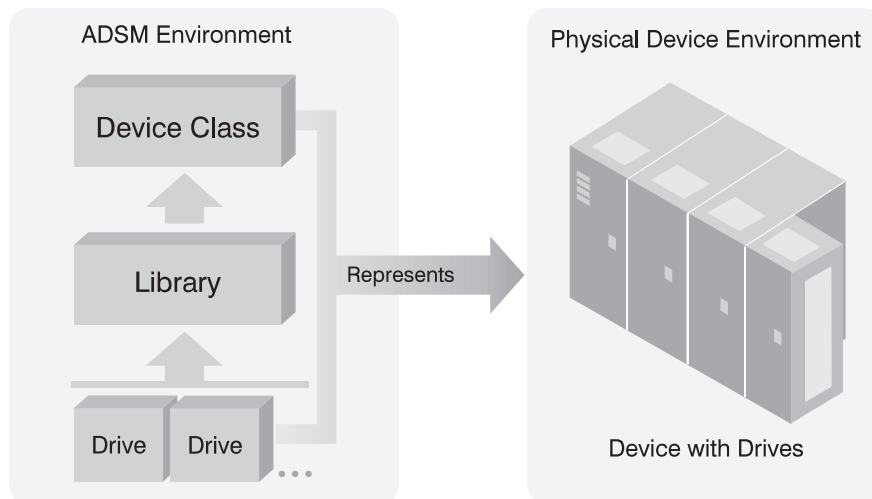


Figure 5. Tape Devices are Represented by a Library, Drive, and Device Class

Files as Logical Devices

ADSM allows administrators to create logical volumes on server disk space with the characteristics of sequential tape volumes. The support for these virtual devices is obtained 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 (logical) devices are often useful when transferring data as in electronic vaulting. For example, an administrator can create FILE devices that append data at the end of existing data and can be restored to actual tape devices at the receiving site.

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 an 8mm device represents collections of only 8mm tapes.

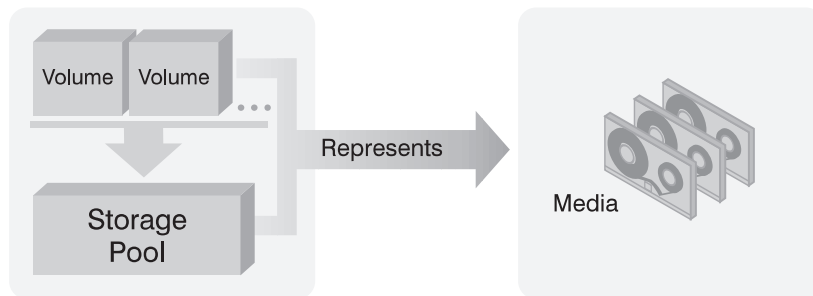


Figure 6. Relationships of Storage Pool Volumes, Storage Pools, and Media

What are the ADSM Storage Objects?

The following ADSM storage objects are collections of information that indicate to ADSM how to communicate with devices and how 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 8, “Defining Device Classes” on page 115 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. If the sequential device is a tape drive, the device class is associated with a library and a drive. Additional storage objects are required for sequential devices because of the many variations in media type (for example, 8mm, QIC, 3490 cartridge) 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 device must be associated with an ADSM library because the ADSM server must know which drives are available before it can mount a volume.

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). See Chapter 7, “Defining Drives and Libraries” on page 107 for more information about ADSM libraries.

Drive

Each drive mechanism within a tape device is represented by an ADSM drive. For devices with multiple drives, including automated libraries, each drive is separately defined to ADSM. See Chapter 7, “Defining Drives and Libraries” on page 107 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, an 8mm tape storage pool contains only 8mm tape volumes. Many of the parameters associated with a storage pool depend on whether the data on storage pool media is accessed randomly or sequentially. These parameters are described in more detail in Chapter 9, “Managing Storage Pools” on page 131.

ADSM supplies default disk storage pools named BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL. To use these default pools, all you must do is define volumes to them. For more information, see “Using Random Access Volumes on Disk Devices” on page 55.

Storage Pool Volumes

ADSM represents space on media with an object called a storage pool volume. A storage pool volume is associated with a storage pool and represents a unit of space available for ADSM client data. For example, 8mm tapes and QIC tapes become storage pool volumes when they are assigned to an ADSM storage pool.

See Chapter 10, “Managing Storage Pool Volumes” on page 179 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 data becomes deleted by the server, the space occupied by that data can be reused.

Device Classes for Sequential Access Devices

Devices such as tape drives access their data sequentially. Device classes for sequential devices contain media management information in addition to a device type. Usually, device classes for sequential access devices also contain a library specification. Figure 7 shows the contents of a device class for a typical sequential access device.

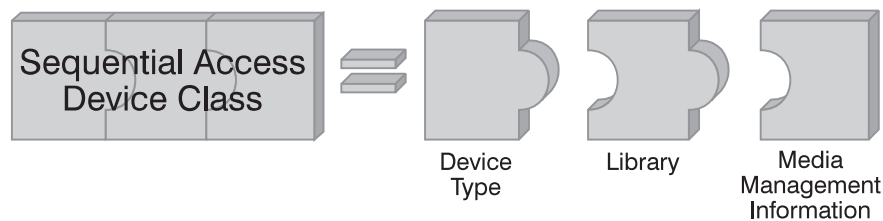


Figure 7. Contents of Device Class for Sequential Access Devices

Sequential access device types 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 149 for details).

Tape devices and FILE type devices are members of the sequential access category of devices. FILE is a special kind of ADSM sequential device type that allows the administrator to create logical tape devices by creating files on the ADSM server that have the characteristics of a tape device.

Device Type

Every sequential access device class requires an ADSM-provided device type as part of its definition. A device type identifies a device as a member of a broad category of devices sharing similar media characteristics. ADSM provides device types for many devices including 8MM, QIC, 3590, CARTRIDGE, REEL, and FILE device types. For

example, 8mm tape devices require 8mm tapes; all 8mm tape devices share a device type of 8MM.

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 you have defined to ADSM, as discussed in “Library” on page 11.

Media Management Information

Every sequential access device class contains media management information, for example, recording format, estimated capacity, and labeling prefixes. For more information about how ADSM helps to manage media, see “Using Disk for FILE Logical Devices” on page 56, Chapter 4, “Using Tape Devices with ADSM” on page 59, and Chapter 7, “Defining Drives and Libraries” on page 107.

Putting It All Together

Figure 8 on page 15 summarizes the relationships among the physical device environment, ADSM storage 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 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 storage pools if the pools are set up in a storage hierarchy.

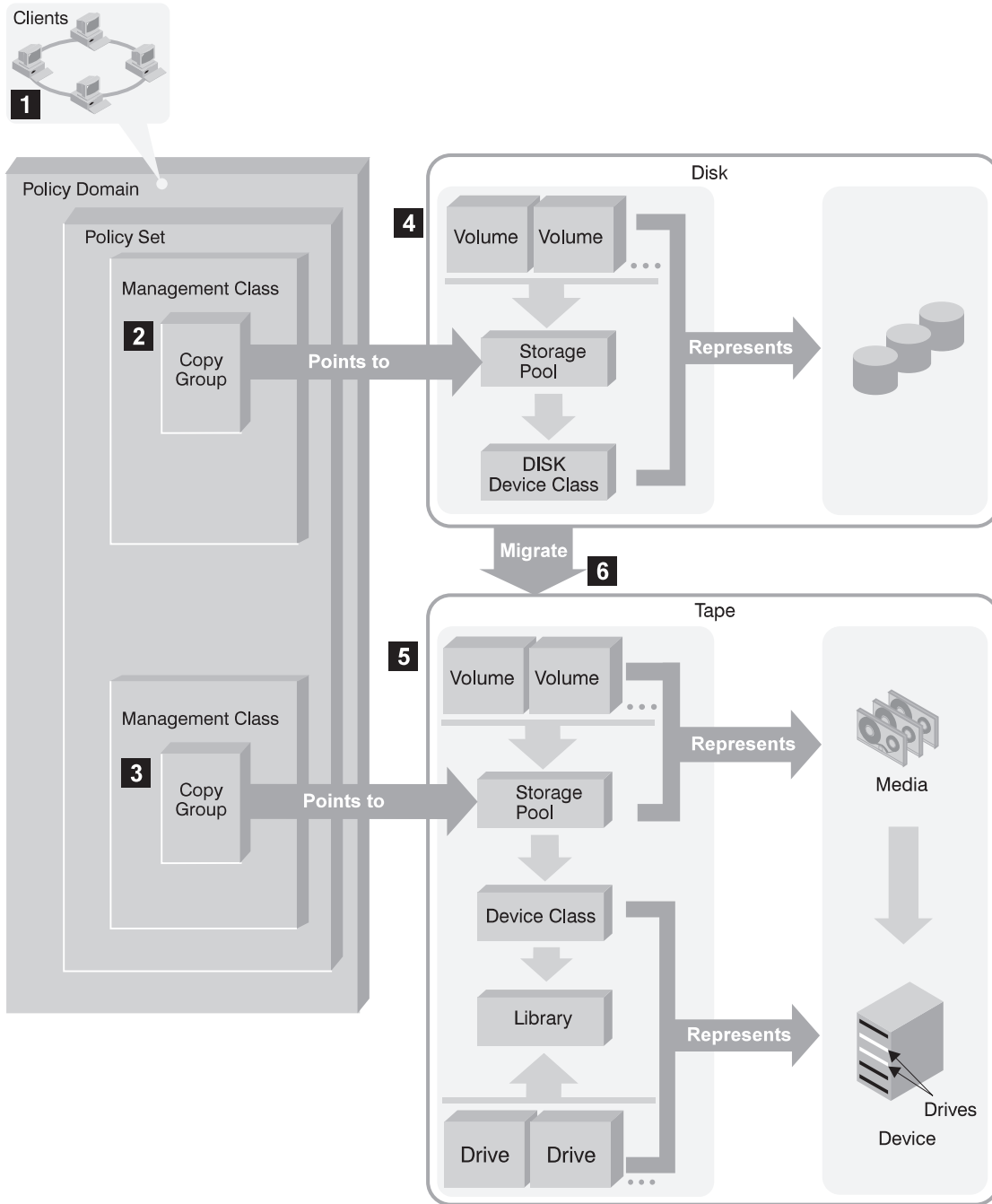


Figure 8. Putting It All Together

Planning to Configure the ADSM Storage Environment

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

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, it is helpful to evaluate the hardware available to ADSM:

1. Determine the storage devices that are available to ADSM. Determine how many tape drives you have that you will allow ADSM to use.
2. Determine the ADSM device type for each of the available devices. Group together similar devices and identify their device classes. For example, create separate categories for QIC, 8mm, and 3490E cartridge devices.

Note: For sequential access devices, categorize the type of tape cartridge based upon capacity. For example, standard cartridge tapes and enhanced capacity cartridge tapes require different device classes.

3. Determine which devices require that an operator mount volumes and which devices are in an automated library (automatic volume mounts). The IBM 3494 and IBM 9427 are examples of automated tape libraries.
4. Categorize storage pools by 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 will need to create a new or use an existing labeling convention for ADSM storage pool volumes.

Mapping Devices to Device Classes

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

- Internal disk drives
- 3490 Enhanced Capability models (3490E) in a 3494 tape library
- 3490 Magnetic Tape Subsystems
- 3480 Magnetic Tape Subsystems

You can map storage devices to device classes as shown in Table 1 on page 17.

Table 1. Mapping Storage Devices to Device Class

Device Class	Description
DISK	Storage volumes that reside on the internal disk drive ADSM provides one DISK device class that is already defined, and you cannot define another device class for disk storage.
CARTRIDGE_E	Enhanced Capacity Cartridge System Tape (ECCST) volumes used with 3490E tape devices
CARTRIDGE_B	Standard Cartridge System Tape (CST) volumes used with 3480 or 3490 Base tape devices

You must define any device classes that you need for your tape devices. See Chapter 8, "Defining Device Classes" on page 115 for information on defining tape 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 on disk or tape. You can then assign each storage pool as a storage destination for backed up, archived, or space-managed files.

For example, an administrator determines that users in the business department have three requirements:

- Immediate access to all space-managed files and to some backed up files, such as accounts receivable and payroll accounts
- Periodic access to some archived files, such as monthly sales and inventory reports
- Occasional access to backed up or archived files that are rarely modified, such as yearly revenue reports

To match user requirements to storage devices, the administrator defines storage pools, device classes, and, for device types that require them, libraries and drives. See Table 2 on page 18.

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

Storage Pool	Device Class	Library (Hardware)	Drives (Hardware)	Volume Type	Storage Destination
BACKUPOOL	DISK	—	—	Storage volumes on the internal disk drive	For a backup copy group for files requiring immediate access
SPACEMGPOOL	DISK	—	—	Storage volumes on the internal disk drive	For a management class for space-managed files that require immediate access
ARCHTAPEF	CARTRIDGE_E	AUTOLIB_3494 (3494)	TAP07, TAP08 (3490Es)	ECCST volumes	For an archive copy group for files requiring quick, reliable access
BACKTAPE	CARTRIDGE_E	AUTOLIB_3494 (3494)	TAP07, TAP08 (3490Es)	ECCST volumes	For backup copy groups for files not requiring immediate access
ARCHTAPES	CARTRIDGE_B	MANUAL_LIB (Manually mounted)	TAP03, TAP04 (3480, 3490)	CST volumes	For archived data not requiring immediate access

Note: ADSM supplies default disk storage pools named BACKUPOOL, ARCHIVEPOOL, and SPACEMGPOOL. To use these default pools, all you must do is define volumes to them. For more information, see “Using Random Access Volumes on Disk Devices” on page 55.

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 55
- Chapter 4, “Using Tape Devices with ADSM” on page 59
- Chapter 5, “Using a Tape Management System with ADSM” on page 89

ADSM categorizes tape devices by their *library type*. A library type denotes 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)
AS400MLB	Volumes are mounted automatically (by robotics, for example)
USRDFN	Volumes are mounted under the control of a tape management system, either by an operator or by an automated library device

Figure 9 on page 19 shows the ADSM library types and the sections that follow describe each library type.

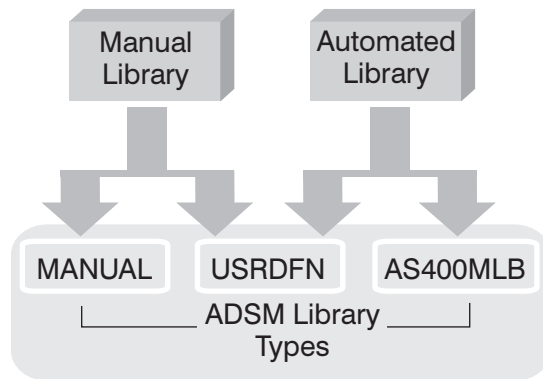


Figure 9. Types of ADSM Libraries

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 REEL 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 an OS/400 message queue 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 Tape Devices with ADSM” on page 59. For how to monitor mount messages for a *MANUAL* library, see “Mount Operations for Manual Libraries” on page 80.

AS400MLB Libraries

An *AS400MLB* library is a collection of drives for which volume mounts and demounts are handled automatically by a robot or other mechanism. The IBM 9427 and the IBM 3494 are examples. When you define an *AS400MLB* library to the ADSM server, you must specify the media library device (MLD) name. OS/400 provides a set of commands that the ADSM server uses to interact with the automated media library. When a volume is mounted in a drive that resides in the *AS400MLB* library, ADSM uses the MLD name to manipulate the volume.

IBM 3590 Drive with ACF: The IBM 3590 cartridge tape device with Automatic Cartridge Facility (ACF) must be defined as an *AS400MLB* library, with one drive.

For guidance on configuring an *AS400MLB* library, see Chapter 4, “Using Tape Devices with ADSM” on page 59. For an example of how to add volumes to an *AS400MLB* library, see “Prepare Volumes for Use by the Library” on page 77.

USRDFN Libraries

A *USRDFN* (user-defined) library is a collection of drives for which volume loads, or mounts, are accomplished via an interaction with a set of user-written exit programs. These exit programs can be customized to provide an interface between ADSM and your media management system. You define a USRDFN library if you want your media management system to support ADSM. Drives with different device types, such as REEL and 8MM, can be combined in a single USRDFN library. For more information on setting up a tape management system for use with ADSM, see Chapter 5, “Using a Tape Management System with ADSM” on page 89. For the details of the interface that ADSM provides, see Appendix, “Interface for Media Management Systems” on page 373.

Automating Client Operations

You can automate operations such as backup for the ADSM clients. Figure 10 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 10 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 10 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 10 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 10 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 10 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 11, “Managing Policies” on page 203. For more details on how to automate client operations, see “Automating Client Operations” on page 241.

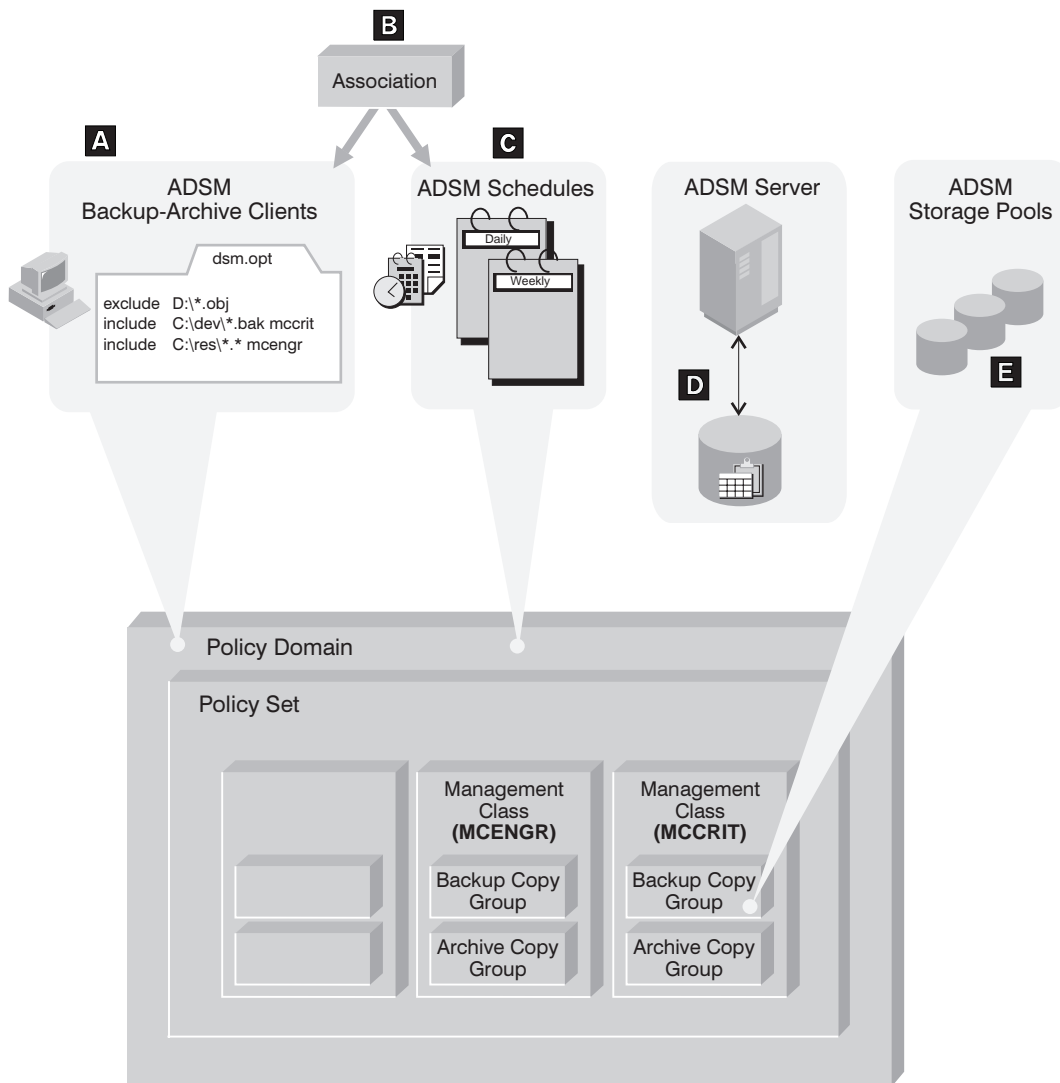


Figure 10. Automating Client Operations

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 tape devices with ADSM
 - Using a tape management system with ADSM
 - Managing tape 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

There are three interfaces to ADSM:

- The graphical user interface (GUI). For information about using the GUI interfaces, see *ADSM Quick Start*.
- The command-line interface. For information about using the command-line interface, see *ADSM Administrator's Reference*.
- The application programming interface. For more information, see *ADSM Using the Application Programming Interface*.

See “Task, GUI, and Command Cross-Reference” on page 28 for tables that relate GUI locations and administrative commands with specific tasks.

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 using random access volumes or logical volumes with 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 55.

Using Tape Devices with ADSM

Tape devices can be used with ADSM for the following purposes:

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

You can configure your tape devices from the Work with devices menu in the ADSM Utilities or manually.

For guidance and scenarios on configuring your tape devices, see Chapter 4, “Using Tape Devices with ADSM” on page 59.

Using a Tape Management System with ADSM

To support ADSM storage media with a tape management system, you can define a USRDFN (user-defined) library to load or mount volumes using a set of user-written exit programs. The exit programs can be customized to provide volume mount, volume dismount, and volume inventory services to ADSM.

For guidance and a scenario for setting up a tape management system for use with ADSM, see Chapter 5, “Using a Tape Management System with ADSM” on page 89.

Managing Tape Operations

When the server requires a volume mount in a library, it generates a request. 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. It is usually not necessary for an operator to respond to the server after completing a requested activity. When the server requires a reply, the message that is displayed by the server requests that the operator issue a reply when the activity is completed.

For information about managing tape operations, see Chapter 6, “Managing Tape Operations” on page 101.

Defining Drives and Libraries

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

For more information, see Chapter 4, “Using Tape Devices with ADSM” on page 59. For additional detailed information about these tasks see Chapter 7, “Defining Drives and Libraries” on page 107.

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 8, “Defining Device Classes” on page 115.

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 keeps 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 9, “Managing Storage Pools” on page 131.

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 10, “Managing Storage Pool Volumes” on page 179.

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. Recall of migrated files is transparent and automatic on recall.

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 11, “Managing Policies” on page 203.

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 12, “Automating Operations” on page 239.

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 13, “Managing Server Operations” on page 263.

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 and the buffer pool 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 14, “Managing the Database and Recovery Log” on page 277.

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 the preceding concepts and tasks, see Chapter 15, “Managing Licensing, Privilege Classes, and Registration” on page 295.

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 16, “Exporting and Importing Data” on page 317.

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 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 17, “Protecting and Recovering Your Data” on page 341.

Task, GUI, and Command Cross-Reference

The tables that follow list the tasks performed by the ADSM administrator and shows where they can be performed on the administrative graphical user interface and command line interface. The sequence of windows that you will follow is shown under column heading "Location in the GUI." You can find detailed help for using the graphical user interface in its online help facility. You can find details about the commands in *ADSM Administrator's Reference* or in the online help for the command-line interface (by using the HELP command).

The following tasks are listed in this section:

1. "Using Magnetic Disk Devices" on page 29
2. "Using Tape Devices with ADSM" on page 29
3. "Using a Tape Management System" on page 31
4. "Managing Tape Operations" on page 32
5. "Defining Drives and Libraries" on page 33
6. "Defining Device Classes" on page 34
7. "Managing Storage Pools" on page 35
8. "Managing Storage Pool Volumes" on page 36
9. "Managing Policies" on page 37
10. "Automating ADSM Operations" on page 39
11. "Managing Server Operations" on page 41
12. "Managing the Database and Recovery Log" on page 43
13. "Managing Licensing, Privilege Classes, and Registration" on page 45
14. "Exporting and Importing Data" on page 47
15. "Protecting and Recovering Your Data" on page 49

Using Magnetic Disk Devices

Table 3 shows a listing of tasks and commands referenced in Chapter 3, “Using Magnetic Disk Devices with ADSM” on page 55.

<i>Table 3. Using Magnetic Disk Devices</i>		
Task	Location in the GUI	Command Used
Format a volume (“Using Random Access Volumes on Disk Devices” on page 55 or “Using Disk for FILE Logical Devices” on page 56)	Not available	CRTVOLADSM from an AS/400 command line
Define a storage pool (“Storage Pool Volumes” on page 179)	Storage Pools	DEFINE STGPOOL
Define the DISK or FILE volume (“Storage Pool Volumes” on page 179)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	DEFINE VOLUME

Using Tape Devices with ADSM

Table 4 shows a listing of tasks and commands referenced in Chapter 4, “Using Tape Devices with ADSM” on page 59.

<i>Table 4 (Page 1 of 2). Using Tape Devices with ADSM</i>			
Task	Location in the GUI	Command Used	ADSM Utilities
Define devices with Device Configuration Utility (“Configuring Devices for ADSM Using the Device Configuration Utility” on page 62)	Not available	Not available	Work with devices
Define devices manually. (“Example of a Manual Library: Setting Up Two 3490E Drives” on page 69)	Not available	<ol style="list-style-type: none"> 1. DEFINE LIBRARY 2. DEFINE DRIVE 3. DEFINE DEVCLASS 4. DEFINE STGPOOL 	

Table 4 (Page 2 of 2). Using Tape Devices with ADSM

Task	Location in the GUI	Command Used	ADSM Utilities
Ensure mount messages for drives are sent to appropriate location (“Example of a Manual Library: Setting Up Two 3490E Drives” on page 69)	Not available		Change server options
Check a storage volume into a library (“Informing the Server about New Volumes in a Library” on page 86)	Not available	CHECKIN LIBVOLUME	
Check a storage volume out of a library (“Removing Volumes from a Library” on page 87)	Not available	CHECKOUT LIBVOLUME	
Change the status of a storage volume (“Changing the Status of a Volume in a Library” on page 87)	Not available	UPDATE LIBVOLUME	
Verify an automated library’s inventory (“Auditing a Library’s Volume Inventory” on page 88)	Not available	AUDIT LIBRARY	

Using a Tape Management System

Table 5 shows a listing of tasks and commands referenced in Chapter 5, "Using a Tape Management System with ADSM" on page 89.

<i>Table 5. Using a Tape Management System</i>		
Task	Location in the GUI	Command Used
Create an exit program ("Creating an Exit Program" on page 90)	Not available	Note: Sample is available in the ADSM product library in QADSM/QAANRSMP
Define the exit program ("Defining the Exit Programs" on page 92)	Not available	DEFINE EXIT
Define a drive ("Defining the Drives" on page 94)	Not available	DEFINE DRIVE
Define a device class ("Defining the Device Classes" on page 94)	Not available	DEFINE DEVCLASS
Define a storage pool ("Defining the Storage Pools" on page 95)	Storage Pools	DEFINE STGPOOL
Update a defined exit program ("Updating an Exit Program" on page 96)	Not available	UPDATE EXIT
Request information on a previously defined exit program ("Querying an Exit Program" on page 96)	Not available	QUERY EXIT
Delete a previously defined exit program ("Deleting an Exit Program" on page 97)	Not available	DELETE EXIT

Managing Tape Operations

Table 6 shows a listing of tasks and commands referenced in Chapter 6, “Managing Tape Operations” on page 101.

Task	Location in the GUI	Command Used
Request information about pending operator requests (“Requesting Information About Pending Operator Requests” on page 101)	Not available	QUERY REQUEST
Reply to operator requests (“Replying to Operator Requests” on page 103)	Not available	REPLY or OS/400 mount message queue
Cancel an operator request (“Cancelling an Operator Request” on page 104)	Not available	CANCEL REQUEST or OS/400 message queue
Determine which volumes are mounted (“Determining Which Volumes are Mounted” on page 104)	Not available	QUERY MOUNT
Dismount an idle volume (“Dismounting an Idle Volume” on page 105)	Not available	DISMOUNT VOLUME

Defining Drives and Libraries

Table 7 shows a listing of tasks and commands referenced in Chapter 7, “Defining Drives and Libraries” on page 107.

<i>Table 7. Defining Drives and Libraries</i>		
Task	Location in the GUI	Command Used
Define a library (“Defining Libraries” on page 110)	Not available	DEFINE LIBRARY
Query a library (“Requesting Information about Libraries” on page 111)	Not available	QUERY LIBRARY
Update a library (“Updating Libraries” on page 111)	Not available	UPDATE LIBRARY
Delete a library (“Deleting Libraries” on page 112)	Not available	DELETE LIBRARY
Define a drive to a library (“Defining and Managing Drives” on page 112)	Not available	DEFINE DRIVE
Display information about a drive (“Requesting Information about Drives” on page 113)	Not available	QUERY DRIVE
Update a drive (“Updating Drives” on page 114)	Not available	UPDATE DRIVE
Delete a drive from a library (“Deleting Drives” on page 114)	Not available	DELETE DRIVE

Defining Device Classes

Table 8 shows a listing of tasks and commands referenced in Chapter 8, “Defining Device Classes” on page 115.

Task	Location in the GUI	Command Used
Define a device class for: <ul style="list-style-type: none">• Sequential media (“Defining and Updating Device Classes for Sequential Media” on page 116)• Files (“Defining and Updating FILE Device Classes” on page 122)• Tape (“Defining and Updating Device Classes for Tape” on page 116)	Not available	DEFINE DEVCLASS
Display information on one or more device classes (“Requesting Information about a Device Class” on page 123)	Not available	QUERY DEVCLASS
Delete a device class (“Deleting a Device Class” on page 124)	Not available	DELETE DEVCLASS

Managing Storage Pools

Table 9 shows a listing of tasks and commands referenced in Chapter 9, “Managing Storage Pools” on page 131.

<i>Table 9. Managing Storage Pools</i>		
Task	Location in the GUI	Command Used
Define a storage pool (Chapter 9, “Managing Storage Pools” on page 131)	Storage Pools	DEFINE STGPOOL
Change the attributes of a storage pool (“Defining or Updating Storage Pools” on page 159)	Storage Pools	UPDATE STGPOOL
Backup a storage pool (“Backing Up Storage Pools” on page 163)	Storage Pools	BACKUP STGPOOL
Query one or more storage pools (“Monitoring the Use of Storage Pool Space” on page 166)	Storage Pools	QUERY STGPOOL
Cancel a migration process (“Canceling the Migration Process” on page 170)	1. Server 2. Processes	CANCEL PROCESS
Display server storage occupancy (“Amount of Space Used by Client Node” on page 173)	Not available	QUERY OCCUPANCY
Delete a storage pool (“Deleting a Storage Pool” on page 175)	Storage Pools	DELETE STGPOOL
Restore a storage pool (“Restoring Storage Pools” on page 176)	Storage Pools	RESTORE STGPOOL

Managing Storage Pool Volumes

Table 10 shows a listing of tasks and commands referenced in Chapter 10, “Managing Storage Pool Volumes” on page 179.

<i>Table 10. Managing Storage Pool Volumes</i>		
Task	Location in the GUI	Command Used
Define a volume in a storage pool (“Defining Storage Pool Volumes” on page 183)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	DEFINE VOLUME
Change user access to a storage pool volume (“Updating Storage Pool Volumes” on page 183)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pools Volumes 	UPDATE VOLUME
Query one or more storage pool volumes (“Requesting General Information about Storage Pool Volumes” on page 185)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	QUERY VOLUME
Query the contents of a storage pool volume (“Viewing a Standard Report on the Contents of a Volume” on page 188)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	QUERY CONTENT
Verify database information for a storage pool volume (“Auditing a Volume in a Disk Storage Pool” on page 191)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	AUDIT VOLUME
Move files on a storage pool volume (“Moving Files from One Volume to Another Volume” on page 193)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	MOVE DATA
Query one or more server processes (“Requesting Information about the Data Movement Process” on page 196)	<ol style="list-style-type: none"> 1. Server 2. Processes 	QUERY PROCESS
Delete a storage pool volume. (“Deleting an Empty Storage Pool Volume” on page 197)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	DELETE VOLUME
Restore files in a storage pool. (“Restoring Storage Pool Volumes” on page 199)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	RESTORE VOLUME

Managing Policies

Table 11 on page 37 shows a listing of tasks and commands referenced in Chapter 11, “Managing Policies” on page 203.

<i>Table 11 (Page 1 of 3). Managing Storage Management Policies</i>		
Task	Location in the GUI	Command Used
Define a new policy domain (“Defining and Updating a Policy Domain” on page 220)	Policy Domains	DEFINE DOMAIN
Copy a policy domain (“Defining and Updating a Policy Domain” on page 220)	Policy Domains	COPY DOMAIN
Update a policy domain (“Defining and Updating a Policy Domain” on page 220)	Policy Domains	UPDATE DOMAIN
Define a policy set (“Defining and Updating a Policy Set” on page 221)	1. Policy Domains 2. Policy Sets	DEFINE POLICYSET
Copy a policy set (“Defining and Updating a Policy Set” on page 221)	1. Policy Domains 2. Policy Sets	COPY POLICYSET
Update a policy set (“Defining and Updating a Policy Set” on page 221)	1. Policy Domains 2. Policy Sets	UPDATE POLICYSET
Define a management class (“Defining and Updating a Management Class” on page 222)	1. Policy Domains 2. Management Class	DEFINE MGMTCLASS
Copy a management class (“Defining and Updating a Management Class” on page 222)	1. Policy Domains 2. Management Class	COPY MGMTCLASS
Update a management class (“Defining and Updating a Management Class” on page 222)	1. Policy Domains 2. Management Class	UPDATE MGMTCLASS
Define a copy group (“Defining and Updating a Backup Copy Group” on page 223 and “Defining and Updating an Archive Copy Group” on page 227)	1. Policy Domains 2. Backup Copy Groups	DEFINE COPYGROUP

Table 11 (Page 2 of 3). Managing Storage Management Policies

Task	Location in the GUI	Command Used
Update a copy group (“Defining and Updating a Backup Copy Group” on page 223 and “Defining and Updating an Archive Copy Group” on page 227)	<ol style="list-style-type: none"> Policy Domains Backup Copy Groups 	UPDATE COPYGROUP
Assign a default management class (“Assigning a Default Management Class” on page 229)	<ol style="list-style-type: none"> Policy Domains Management Class 	ASSIGN DEFMGMTCLASS
Validate a policy set (“Validating and Activating Policy Sets” on page 229)	<ol style="list-style-type: none"> Policy Domains Policy Sets 	VALIDATE POLICYSET
Activate a policy set (“Validating and Activating Policy Sets” on page 229)	<ol style="list-style-type: none"> Policy Domains Policy Sets 	ACTIVATE POLICYSET
Manually start inventory expiration processing (“Expiration Processing” on page 211)	Not available	EXPIRE INVENTORY
Display information about a copy group (“Querying Copy Groups” on page 232)	<ol style="list-style-type: none"> Policy Domains Backup Copy Groups or Archive Copy Groups 	QUERY COPYGROUP
Query a management class (“Querying Management Classes” on page 232)	<ol style="list-style-type: none"> Policy Domains Management Class 	QUERY MGMTCLASS
Query a policy set (“Querying Policy Sets” on page 233)	<ol style="list-style-type: none"> Policy Domains Policy Sets 	QUERY POLICYSET
Query one or more policy domains (“Querying Policy Domains” on page 233)	Policy Domains	QUERY DOMAIN
Delete a copy group (“Deleting Copy Groups” on page 234)	<ol style="list-style-type: none"> Policy Domains Backup Copy Groups or Archive Copy Groups 	DELETE COPYGROUP
Delete a management class (“Deleting Management Classes” on page 235)	<ol style="list-style-type: none"> Policy Domains Management Class 	DELETE MGMTCLASS
Delete a policy set (“Deleting Policy Sets” on page 235)	<ol style="list-style-type: none"> Policy Domains Policy Sets 	DELETE POLICYSET

<i>Table 11 (Page 3 of 3). Managing Storage Management Policies</i>		
Task	Location in the GUI	Command Used
Delete a policy domain ("Deleting Policy Domains" on page 235)	Policy Domains	DELETE DOMAIN

Automating ADSM Operations

Table 12 shows a listing of tasks and commands referenced in Chapter 12, "Automating Operations" on page 239.

<i>Table 12 (Page 1 of 3). Automating ADSM Operations</i>		
Task	Location in the GUI	Command Used
Define a client schedule ("Automating Client Operations" on page 241)	1. Central Scheduler 2. Backup/Archive Schedules	DEFINE SCHEDULE
Associate client nodes with a schedule ("Associating Client Nodes with Schedules" on page 242)	1. Central Scheduler 2. Administrative Events or Backup/Archive Events	DEFINE ASSOCIATION
Query client node associations with a schedule ("Querying Associations" on page 258)	1. Central Scheduler 2. Administrative Events or Backup/Archive Events	QUERY ASSOCIATION
Query one or more client schedules ("Verifying the Schedule" on page 243)	1. Central Scheduler 2. Backup/Archive Schedules	QUERY SCHEDULE
Define a server schedule ("Automating Server Operations" on page 240)	1. Central Scheduler 2. Administrative Command Schedules	DEFINE SCHEDULE
Query one or more server schedules ("Verifying the Schedule" on page 241)	1. Central Scheduler 2. Administrative Command Schedules	QUERY SCHEDULE
Select a central scheduling mode ("Setting the Scheduling Mode" on page 245)	Central Scheduler	SET SCHEDMODES
Set maximum scheduled sessions ("Setting the Maximum Percentage of Sessions for Scheduled Operations" on page 247)	Central Scheduler	SET MAXSCHEDSESSIONS

Table 12 (Page 2 of 3). Automating ADSM Operations

Task	Location in the GUI	Command Used
Randomly distribute scheduled start times (“Randomizing Schedule Start Times” on page 247)	Central Scheduler	SET RANDOMIZE
Control how often client nodes contact the server to perform scheduled operations. (“Setting How Often Clients Query the Server” on page 249)	Central Scheduler	SET QUERYSCHEDPERIOD
Set number of times scheduler retries commands (“Setting the Number of Command Retry Attempts” on page 249)	Central Scheduler	SET MAXCMDRETRIES
Set time between retry attempts (“Setting the Amount of Time between Retry Attempts” on page 249)	Central Scheduler	SET RETRYPERIOD
Tailoring schedules (“Tailoring Schedules” on page 250)	<ol style="list-style-type: none"> 1. Central Scheduler 2. Administrative Command Schedules or Backup/Archive Schedules 	UPDATE SCHEDULE
Copy a schedule (“Copying Schedules” on page 254)	<ol style="list-style-type: none"> 1. Central Scheduler 2. Administrative Command Schedules or Backup/Archive Schedules 	COPY SCHEDULE
Delete one or more schedules (“Deleting Schedules” on page 255)	<ol style="list-style-type: none"> 1. Central Scheduler 2. Administrative Command Schedules or Backup/Archive Schedules 	DELETE SCHEDULE
Query scheduled and completed events (“Querying Event Records” on page 255)	<ol style="list-style-type: none"> 1. Central Scheduler 2. Administrative Events or Backup/Archive Events 	QUERY EVENT
Set the retention period for event records (“Setting the Event Record Retention Period” on page 257)	<ol style="list-style-type: none"> 1. Central Scheduler 2. Administrative Events or Backup/Archive Events 	SET EVENTRETENTION

Table 12 (Page 3 of 3). Automating ADSM Operations

Task	Location in the GUI	Command Used
Delete event records (“Deleting Event Records” on page 258)	1. Central Scheduler 2. Administrative Events or Backup/Archive Events	DELETE EVENT
Delete node associations with a schedule (“Associating Client Nodes with Schedules” on page 242)	1. Central Scheduler 2. Administrative Events or Backup/Archive Events	DELETE ASSOCIATION

Managing Server Operations

Table 13 shows a listing of tasks and commands referenced in Chapter 13, “Managing Server Operations” on page 263.

Table 13 (Page 1 of 2). Managing Server Operations

Task	Location in the GUI	Command Used
Starting the server (“Starting the Server” on page 263)	Not available	1. GO ADSM at any command line 2. STRSVADSM
Shut down the server (“Halting the Server” on page 264)	Server	HALT
Query one or more client sessions (“Requesting Information about Client Sessions” on page 265)	1. Server 2. Sessions	QUERY SESSION
Cancel one or more client sessions (“Canceling a Client Session” on page 267)	1. Server 2. Sessions	CANCEL SESSION
Temporarily prevent client node access to the server (“Disabling or Enabling Server Access” on page 267)	Server	DISABLE
Resume user activity on the server (“Disabling or Enabling Server Access” on page 267)	Server	ENABLE
Request information about server background processes (“Requesting Information about Server Processes” on page 269)	1. Server 2. Processes	QUERY PROCESS

Table 13 (Page 2 of 2). Managing Server Operations

Task	Location in the GUI	Command Used
Cancel a server background processes (“Canceling Server Processes” on page 269)	<ol style="list-style-type: none"> Server Processes 	CANCEL PROCESS
Bring a random access volume online or offline (“Varying Disk Volumes Online or Offline” on page 270)	<p><i>Database volumes:</i></p> <ol style="list-style-type: none"> Database Database Volumes <p><i>Recovery log volumes:</i></p> <ol style="list-style-type: none"> Database Recovery Log Recovery Log Volumes <p><i>Storage pool volumes:</i></p> <ol style="list-style-type: none"> Storage Pools Storage Pool Volumes 	VARY
Query system parameters (“Requesting Information about Server Status” on page 270)	Server	QUERY STATUS
Specify the server name (“Setting the Server Name” on page 271)	Server	SET SERVERNAME
Query one or more server options (“Querying Server Options” on page 271)	Not available	QUERY OPTION
Set the retention period for the activity log (“Setting the Activity Log Retention Period” on page 273)	Server	SET ACTLOGRETENTION
Search activity log for messages (“Requesting Information from the Activity Log” on page 273)	<ol style="list-style-type: none"> Server Activity Log 	QUERY ACTLOG
Set accounting records on or off (“Monitoring Accounting Records” on page 274)	Server	SET ACCOUNTING
Get help on administrative commands and messages (“Getting Help on Commands and Error Messages” on page 275)	Not available	HELP

Managing the Database and Recovery Log

Table 14 on page 43 shows a listing of tasks and commands referenced in Chapter 14, “Managing the Database and Recovery Log” on page 277.

<i>Table 14 (Page 1 of 2). Managing the Database and Recovery Log</i>		
Task	Location in the GUI	Command Used
Reset the utilization statistics for the database (“Monitoring the Database and Recovery Log” on page 280)	Database	RESET DBMAXUTILIZATION
Reset the utilization statistics for the log (“Monitoring the Database and Recovery Log” on page 280)	Database Recovery Log	RESET LOGMAXUTILIZATION
Create a database volume (“Step 1: Allocating Space for the Database and Recovery Log” on page 282)	1. Database 2. Database Volumes	CRTVOLADSM from an AS/400 command line
Define a database volume (“Step 2: Defining Database or Recovery Log Volumes to ADSM” on page 283)	1. Database 2. Database Volumes	DEFINE DBVOLUME
Define a recovery log volume (“Step 2: Defining Database or Recovery Log Volumes to ADSM” on page 283)	1. Database Recovery Log 2. Recovery Log Volumes	DEFINE LOGVOLUME
Increase the assigned capacity of the database (“Step 3: Extending the Capacity of the Database or Recovery Log” on page 285)	Database	EXTEND DB
Increase the assigned capacity of the recovery log (“Step 3: Extending the Capacity of the Database or Recovery Log” on page 285)	Database Recovery Log	EXTEND LOG
Display information about volumes defined to the database (“Step 1: Determining If Volumes Can Be Deleted” on page 286)	1. Database 2. Database Volumes	QUERY DBVOLUME

Table 14 (Page 2 of 2). Managing the Database and Recovery Log

Task	Location in the GUI	Command Used
Display information about volumes defined to the recovery log ("Step 1: Determining If Volumes Can Be Deleted" on page 286)	<ol style="list-style-type: none"> Database Recovery Log Recovery Log Volumes 	QUERY LOGVOLUME
Decrease the assigned capacity of the database ("Step 2: Reducing the Capacity of the Database or Recovery Log" on page 288)	Database	REDUCE DB
Decrease the assigned capacity of the recovery log ("Step 2: Reducing the Capacity of the Database or Recovery Log" on page 288)	Database Recovery Log	REDUCE LOG
Delete a database volume ("Step 3: Deleting a Volume from the Database or Recovery Log" on page 288)	<ol style="list-style-type: none"> Database Database Volumes 	DELETE DBVOLUME
Delete a recovery log volume ("Step 3: Deleting a Volume from the Database or Recovery Log" on page 288)	<ol style="list-style-type: none"> Database Recovery Log Recovery Log Volumes 	DELETE LOGVOLUME
Reset the buffer pool statistics for the database ("Step 1: Resetting Database Buffer Pool Utilization Statistics" on page 290)	Not available	RESET BUFPOOL
Display information on the database ("Step 2: Requesting Information about the Database Buffer Pool" on page 290)	Database	QUERY DB
Display information on the recovery log ("Step 1: Requesting Information about the Recovery Log Buffer Pool" on page 292)	Database Recovery Log	QUERY LOG

Managing Licensing, Privilege Classes, and Registration

Table 15 on page 45 shows a listing of tasks and commands referenced in Chapter 15, “Managing Licensing, Privilege Classes, and Registration” on page 295.

<i>Table 15 (Page 1 of 3). Managing Licensing, Privilege Classes, and Registration</i>		
Task	Location in the GUI	Command Used
Register a new license (“Managing ADSM Licenses” on page 295)	Server	REGISTER LICENSE
Audit the current server configuration and licenses (“Monitoring Licenses” on page 298)	Server	AUDIT LICENSES
Display license information (“Monitoring Licenses” on page 298)	Server	QUERY LICENSE
Set license audit period (“Monitoring Licenses” on page 298)	Server	SET LICENSEAUDITPERIOD
Set password authentication (“Setting Password Authentication” on page 299)	Server	SET AUTHENTICATION
Set password expiration date (“Setting User Password Expiration” on page 299)	Server	SET PASSEXP
Register an administrator (“Registering Administrators or Updating Information” on page 300)	Administrators	REGISTER ADMIN
Update an administrator (“Registering Administrators or Updating Information” on page 300)	Administrators	UPDATE ADMIN
Add administrator authority (“Granting Administrative Authority” on page 300)	Administrators	GRANT AUTHORITY
Revoke or reduce administrator authority (“Changing Administrative Authority” on page 304)	Administrators	REVOKE AUTHORITY

Table 15 (Page 2 of 3). Managing Licensing, Privilege Classes, and Registration

Task	Location in the GUI	Command Used
Rename an administrator ("Renaming an Administrator" on page 306)	Administrators	RENAME ADMIN
Delete an administrator ("Removing Administrators" on page 306)	Administrators	REMOVE ADMIN
Lock out an administrator ("Locking and Unlocking Administrators from the Server" on page 307)	Administrators	LOCK ADMIN
Unlock an administrator ("Locking and Unlocking Administrators from the Server" on page 307)	Administrators	UNLOCK ADMIN
Display information on one or more administrators ("Requesting Information about Administrators" on page 307)	Administrators	QUERY ADMIN
Set open or closed registration ("Setting Client Node Registration" on page 308)	Central Scheduler	SET REGISTRATION
Invoke a macro file of one or more ADSM commands ("Administrator Registration of Client Nodes" on page 309)	Not available	MACRO
Register a client node ("Administrator Registration of Client Nodes" on page 309)	Nodes	REGISTER NODE
Register an application programming interface ("Registering an Application Programming Interface to the Server" on page 315)	Nodes	REGISTER NODE
Update a client node ("Updating Client Node Information" on page 310)	Nodes	UPDATE NODE
Rename a client node ("Renaming Client Nodes" on page 311)	Nodes	RENAME NODE

<i>Table 15 (Page 3 of 3). Managing Licensing, Privilege Classes, and Registration</i>		
Task	Location in the GUI	Command Used
Lock out a client node (“Locking and Unlocking Client Nodes” on page 311)	Nodes	LOCK NODE
Unlock a client node (“Locking and Unlocking Client Nodes” on page 311)	Nodes	UNLOCK NODE
Query one or more client nodes (“Requesting Information about Client Nodes” on page 311)	Nodes	QUERY NODE
Query one or more file spaces (“Requesting Information about File Spaces” on page 313)	File Spaces	QUERY FILESPACE
Delete client node data from the server (“Deleting a File Space” on page 314)	File Spaces	DELETE FILESPACE
Delete a client node (“Removing Client Nodes” on page 315)	Nodes	REMOVE NODE

Exporting and Importing Data

Table 16 shows a listing of tasks and commands referenced in Chapter 16, “Exporting and Importing Data” on page 317.

<i>Table 16 (Page 1 of 3). Exporting and Importing Data</i>		
Task	Location in the GUI	Command Used
Copy server information to sequential media (“Preparing to Export or Import Data” on page 318)	Server	EXPORT SERVER
Request information about an export or import process (“Requesting Information about an Export or Import Process” on page 320)	<ol style="list-style-type: none"> 1. Server 2. Processes 	QUERY PROCESS

Table 16 (Page 2 of 3). Exporting and Importing Data

Task	Location in the GUI	Command Used
Query activity log for export or import information (“Querying the Activity Log for Export or Import Information” on page 323)	Administrators	QUERY ACTLOG
Copy administrator information to sequential media (“Exporting Administrator Information” on page 326)	Administrators	EXPORT ADMIN
Copy client node information to sequential media (“Exporting Client Node Information” on page 327)	Nodes	EXPORT NODE
Copy policy information to sequential media (“Exporting Policy Information” on page 328)	Policy Domains	EXPORT POLICY
Preview information before importing data (“Step 1: Previewing Information before You Import Data” on page 329)	Server	IMPORT SERVER
Import administrator information (“Step 2: Importing Definitions” on page 331)	Administrators	IMPORT ADMIN
Import client node information (“Step 2: Importing Definitions” on page 331)	Nodes	IMPORT NODE
Import policy information (“Step 2: Importing Definitions” on page 331)	Policy Domains	IMPORT POLICY
Query one or more copy groups (“Step 3: Tailoring Server Storage Definitions on the Target Server” on page 333)	<ol style="list-style-type: none"> 1. Policy Domains 2. Backup Copy Groups or Archive Copy Groups 	QUERY COPYGROUP

<i>Table 16 (Page 3 of 3). Exporting and Importing Data</i>		
Task	Location in the GUI	Command Used
Query one or more management classes (“Step 3: Tailoring Server Storage Definitions on the Target Server” on page 333)	<ol style="list-style-type: none"> 1. Policy Domains 2. Management Class 	QUERY MGMTCLASS
Import file data information (“Step 4: Importing File Data Information” on page 334)	Server	IMPORT SERVER
Import file data information (“Step 4: Importing File Data Information” on page 334)	Nodes	IMPORT NODE
Rename a client file space on the server (“Renaming a File Space” on page 337)	File Spaces	RENAME FILESPACE

Protecting and Recovering Your Data

Table 17 shows a listing of tasks and commands referenced in Chapter 17, “Protecting and Recovering Your Data” on page 341.

<i>Table 17 (Page 1 of 4). Protecting and Recovering Your Data</i>		
Task	Location in the GUI	Command Used
Backup a storage pool (“Backing Up Storage Pools” on page 345)	Storage Pools	BACKUP STGPOOL
Allocate disk volumes to mirror the database and recovery log (“Mirroring the Database and Recovery Log” on page 347)	Not available	CRTVOLADSM from an AS/400 command line
Define a volume copy of a database volume (“Defining Database or Recovery Log Mirrored Volumes” on page 347)	<ol style="list-style-type: none"> 1. Database 2. Database Volumes 	DEFINE DBCOPY
Define a volume copy of a recovery log volume (“Defining Database or Recovery Log Mirrored Volumes” on page 347)	<ol style="list-style-type: none"> 1. Database Recovery Log 2. Recovery Log Volumes 	DEFINE LOGCOPY

Table 17 (Page 2 of 4). Protecting and Recovering Your Data

Task	Location in the GUI	Command Used
Display information about database mirrored copies (“Requesting Information about Mirrored Volumes” on page 348)	<ol style="list-style-type: none"> Database Database Volumes 	QUERY DBVOLUME
Display information on one or more log volumes (“Requesting Information about Mirrored Volumes” on page 348)	<ol style="list-style-type: none"> Database Recovery Log Recovery Log Volumes 	QUERY LOGVOLUME
Define device classes for database backups (“Backing Up the Database” on page 349)	Not available	DEFINE DEVCLASS
Set the recovery log mode (“Setting the Recovery Log Mode” on page 350)	Database Recovery Log	SET LOGMODE
Adjust the size of the recovery log (“Estimating the Size of the Recovery Log” on page 351)	Database Recovery Log	REDUCE LOG
Set a database backup trigger (“Setting a Database Backup Trigger” on page 352)	Database	DEFINE DBBACKUPTRIGGER
Request information about the backup trigger information (“Setting a Database Backup Trigger” on page 352)	Database	QUERY DBBACKUPTRIGGER
Update the database backup trigger information (“Setting a Database Backup Trigger” on page 352)	Database	UPDATE DBBACKUPTRIGGER
Delete the database backup trigger information (“Setting a Database Backup Trigger” on page 352)	Database	DELETE DBBACKUPTRIGGER
Back up volume history information (“Saving the Volume History File” on page 354)	<ol style="list-style-type: none"> Server Sequential Volume History 	BACKUP VOLHISTORY

Table 17 (Page 3 of 4). Protecting and Recovering Your Data

Task	Location in the GUI	Command Used
Display volume history information ("Saving the Volume History File" on page 354)	<ol style="list-style-type: none"> 1. Server 2. Sequential Volume History 	QUERY VOLHISTORY
Delete volume history information ("Saving the Volume History File" on page 354)	<ol style="list-style-type: none"> 1. Server 2. Sequential Volume History 	DELETE VOLHISTORY
Backup the device configuration information file ("Saving the Device Configuration Backup File" on page 355)	Server	BACKUP DEVCONFIG
Perform a full or incremental database backup ("Doing Full and Incremental Backups" on page 357)	<ol style="list-style-type: none"> 1. Database 2. Database Volumes 	BACKUP DB
Place a failing database volume offline ("Recovering by Using Mirrored Volumes" on page 358)	<ol style="list-style-type: none"> 1. Database 2. Database Volumes 	DELETE DBVOLUME
Place a failing recovery log volume offline ("Recovering by Using Mirrored Volumes" on page 358)	<ol style="list-style-type: none"> 1. Database Recovery Log 2. Recovery Log Volumes 	DELETE LOGVOLUME
Place a database volume online ("Recovering by Using Mirrored Volumes" on page 358)	<ol style="list-style-type: none"> 1. Database 2. Database Volumes 	DEFINE DBCOPY
Place a recovery log volume online ("Recovering by Using Mirrored Volumes" on page 358)	<ol style="list-style-type: none"> 1. Database Recovery Log 2. Recovery Log Volumes 	DEFINE LOGCOPY
Restore the database to a point in time or most current state ("Restoring a Database to a Point in Time" on page 359 or "Restoring a Database to its Most Current State" on page 362)	Not available	STRRSTADSM

Table 17 (Page 4 of 4). Protecting and Recovering Your Data

Task	Location in the GUI	Command Used
Audit all disk volumes and delete volumes located ("Restoring a Database to a Point in Time" on page 359)	<ol style="list-style-type: none"> 1. Storage Pools 2. Volumes 	AUDIT VOLUME
Restore damaged files after an audit is performed and delete database entries for files not found in the copy storage pool ("Restoring a Database to a Point in Time" on page 359)	Storage Pools	RESTORE STGPOOL
Delete storage pool volumes from the database if no backups are available ("Restoring a Database to a Point in Time" on page 359)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	DELETE VOLUME
Detect damaged user files and reset damaged status if error is temporary ("Maintaining the Integrity of Files" on page 363)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	AUDIT VOLUME
Restore files marked as damaged ("Maintaining the Integrity of Files" on page 363)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	RESTORE VOLUME
Identify files marked as damaged and recreate any damaged files ("Maintaining the Integrity of Files" on page 363)	Storage Pools	RESTORE STGPOOL
Display damaged files in a specific volume ("Maintaining the Integrity of Files" on page 363)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	QUERY CONTENT
Check for and replace files that develop data-integrity problems ("Maintaining the Integrity of Files" on page 363)	<ol style="list-style-type: none"> 1. Storage Pools 2. Storage Pool Volumes 	QUERY VOLUME
Create backup copies of files for new storage pool ("Maintaining the Integrity of Files" on page 363)	Storage Pools	BACKUP STGPOOL

Part 2. Configuring and Managing Server Storage

Chapter 3. Using Magnetic Disk Devices with ADSM

With ADSM, magnetic disk devices are used for two main purposes:

- To store the database and the recovery log.

For information on using disk storage for the database and recovery log, see Chapter 14, “Managing the Database and Recovery Log” on page 277.

- 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.

You may also want to use disk storage (in the form of FILE volumes) to store backups of the ADSM database and to export and import ADSM data.

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

Section	Page
Concepts:	
Using cache	57
Freeing space on disk	57
Scratch FILE volumes	57
FILE volumes used for database backups and export operations	57
Tasks:	
Using random access volumes on disk devices	55
Using disk for FILE logical devices	56

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.”
- 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 Logical Devices” on page 56.

Using Random Access Volumes on Disk Devices

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

Do the following to use random access volumes on a disk device:

1. Format a random access volume. See Chapter 10, “Managing Storage Pool Volumes” on page 179 for details on using the ADSM random access volume formatting utility.
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). See “Example: Defining a Storage Pool Hierarchy” on page 160 for details.
3. Define the DISK volumes formatted in step 1 to the storage pool. See “Defining Storage Pool Volumes” on page 183 for details.
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 11, “Managing Policies” on page 203 for details.
 - 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 161.

Using Disk for FILE Logical Devices

Another way to use magnetic disk storage is to use files as logical volumes that store data sequentially (as on tape volumes). FILE (logical) devices are often useful when transferring data as in electronic vaulting. For example, an administrator can create FILE devices that append data at the end of existing data and can be restored to actual tape devices at the receiving site.

Do the following:

1. Define a device class with device type FILE. See “Defining and Updating FILE Device Classes” on page 122.
2. Define a storage pool that is associated with the new FILE device class. See “Defining a Primary Storage Pool” on page 159 for details.
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 11, “Managing Policies” on page 203 for details.
 - 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 161.

ADSM automatically creates the files to be used as sequential access volumes.

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 143.

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 231.
- Set the server option for the expiration interval, so that expiration processing runs periodically. You can set options through the ADSM Utilities menu or by issuing the CHGSVRADSM command (see *ADSM Administrator's Reference*).

Scratch FILE Volumes

Scratch volumes are created automatically in the OS/400 library that you specified when you defined the FILE device class. When the volumes used in storage pools become empty, the files are deleted.

FILE Volumes Used for Database Backups and Export Operations

When you backup 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 354.

Chapter 4. Using Tape Devices with ADSM

ADSM can use tape devices 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:	
Summary of how to configure devices	60
Notes on operations	79
Tasks:	
Configuring devices using the device configuration utility	62
Configuring a manual library	69
Configuring an automated library	73
Managing storage volumes in automated libraries	83

Some tasks presented in this chapter can be performed using either the graphical user interface (GUI) or the command line interface. Table 7 on page 33 shows whether a task can be performed on the GUI, the command line interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

Configuring the Devices

After tape devices have been attached and defined to your AS/400 system, you must then configure them to ADSM so that ADSM can use and manage the devices and their media.

Libraries with One Drive: If you set up a library with only one drive, you will not be able to use ADSM's automatic volume reclamation and will have to reclaim volumes by a manual process. Reclamation allows reuse of volumes after data on the volumes expires. If you will be using single-drive libraries, see "Space Reclamation for Sequential Access Storage Pools" on page 149 and "Reclamation in a Single-Drive Library" on page 153.

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.

Summary: Configuring Devices

Sequential access devices typically require that the following steps be performed so that ADSM can use the devices. The numbers in the steps correspond to the numbers in Figure 11 on page 61.

1 Configure the device to the operating system.

Most devices are automatically configured by AS/400 when they are attached to the system. If necessary, see the following publications:

- For manual libraries (libraries that require an operator to mount volumes), see *AS/400 Local Device Configuration*.
- For automated libraries (libraries that have a robot or other device to automatically mount volumes), see *AS/400 Automated Tape Library Planning and Management*. For IBM 3494 and 3495 automated libraries, see also *IBM 3494 Tape Library Dataserver User's Guide: Media Library Device Driver for AS/400*.

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 11 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 6. For a description of the standard policy that is installed with ADSM, see “Using the Standard Storage Management Policies” on page 216.

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 Label tape volumes for the device. For automated libraries (AS400MLB type), add the volumes to the device's volume inventory. (This step is not illustrated.)

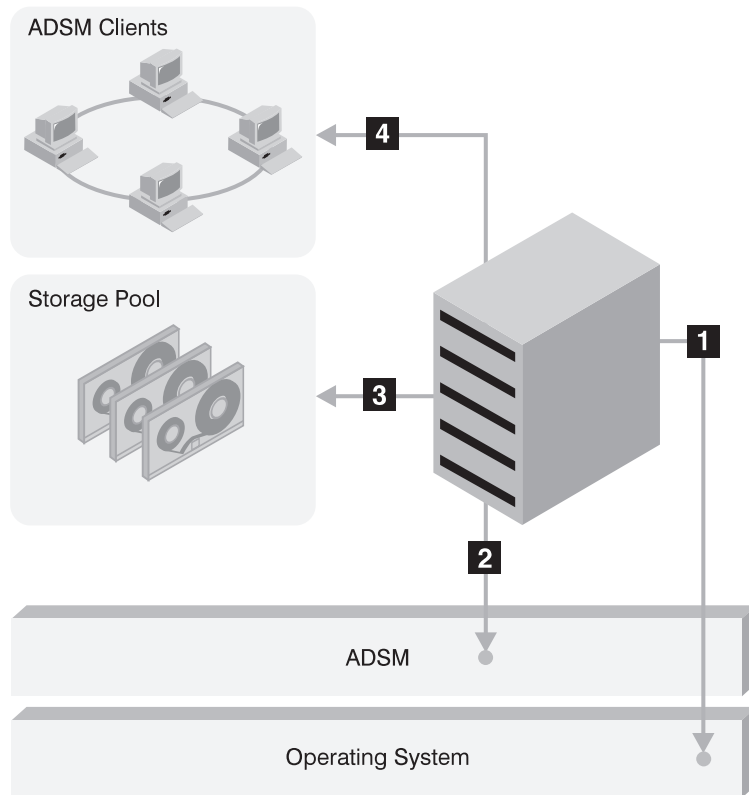


Figure 11. Overview of Sequential Device Configuration

ADSM provides a utility to assist you in configuring devices attached to your AS/400 so that ADSM can use them. For an example of setting up devices using the utility, see “Configuring Devices for ADSM Using the Device Configuration Utility” on page 62. The example shows how to set up two 3490E drives.

You can also configure the devices to ADSM manually. For an example of setting up a manual library, see “Example of a Manual Library: Setting Up Two 3490E Drives” on page 69. For an example of setting up an automated library, see “Example of an Automated Library: Setting up a 9427 Library” on page 73.

Configuring Devices for ADSM Using the Device Configuration Utility

With the device configuration utility, you can get assistance to do the following:

- Define libraries and drives to ADSM
OS/400 V3R6 and Subsequent Releases: You do not define drives for AS400MLB and USRDFN libraries.
- Define ADSM device classes
- Define ADSM storage pools
- Define and change the primary storage pool hierarchy

When you use this utility, you create a file that contains ADSM commands, called a *macro* file. The utility submits the macro to the ADSM server for processing.

To use the utility, do the following:

- 1** From the ADSM main menu on the AS/400, select 4 (Verify server status). The server must be running to use the Work with Devices utility.
Prerequisite: The AS/400 administrative client for ADSM must be defined and communications set up for it. However, you do not need to start the administrative client.
- 2** From the ADSM main menu on the AS/400, select 1 (Utilities).
- 3** From the Utilities menu, select 2 (Work with devices). (You can also use the WRKDEVADSM command on the AS/400 command line.)

The utility obtains information from the OS/400 operating system to determine the tape and automated library devices that are attached to the AS/400. The utility also checks the ADSM database to determine which of these devices are already defined to ADSM. Based on this information, the utility produces a list of storage pools and associated device classes, libraries, and drives, arranged so that you can see the associations among these objects. The objects in the list have a status of either DEFINED (known or defined to ADSM) or NEW (not known or defined to ADSM).

For example, if your system has two 3490E tape drives attached that have not been defined to ADSM, you might see a screen as shown in Figure 12 on page 63.


```

                                Work with Devices for ADSM

Macro output file . . . MACROS____
Library . . . . . QUSRADSM____
Macro output member      DEVCFG____

Type options, press Enter.
  2=Change  5=Display

Opt   Storage Hierarchy      Type           Status
--   -
--   ARCHIVEPOOL             STGPOOL        DEFINED
--   BACKUPPOOL              STGPOOL        DEFINED
--   SPACEMGPOOL             STGPOOL        DEFINED
--   CART_TAPEPOOL1          STGPOOL        NEW
--   CART_TAPECLASS1         DEVCLASS       NEW
--   MANLIB                   LIBRARY        NEW
--   TAP03                     DRIVE          NEW
--   TAP04                     DRIVE          NEW

                                Bottom

F3=Exit  F5=Refresh  F8=Submit to server  F9=Command
F10=Work with libraries  F11=Work with storage pools  F12=Cancel

```

Figure 12. Work with Devices Screen

- 4** Make changes as needed to the list. You can make changes to objects not yet defined to ADSM, and make limited changes to objects that are already defined to ADSM.

For each device not defined to ADSM, the utility creates a list of new ADSM objects (library, drives, device class, and storage pool) that the utility will ask the server to define. You can change the names and characteristics of these objects, or delete them if you do not want them defined to ADSM.

Using the Work with Devices Screen

- You cannot change or specify all parameters that apply to the objects listed in the Work with Devices screen. If you want to do more than what the utility allows, you must use ADSM commands, outside the utility.
- Standard context-sensitive help is available. Move the cursor to a field and press F1 for help on that field.
- For disk storage pools, you can only change the storage migration hierarchy. To make any other changes, you must use an ADSM administrative client.
- You can delete new storage pools and device classes if you do not want ADSM to create them. To work with the storage pools and device classes, use F11.
- You can delete new libraries and drives if you do not want ADSM to use them. Delete the associated storage pool and device class first. To work with the libraries and drives, use F10.

- For automated libraries, you can change only the category. The utility does not check whether this category is available on the AS/400 system. You must ensure that the category is unique for the library.
- The utility predicts a mount limit for a device class based on the drives that the utility detects and groups together under a device class and storage pool. If possible, use a mount limit of at least two (two drives in a library) so that you can use ADSM's automatic tape reclamation. For more information, see "Space Reclamation for Sequential Access Storage Pools" on page 149.
- To define a USRDFN library (a library managed by a media management system instead of ADSM), you can change the library type for an ADSM manual library to USRDFN. However, you still must create and define the exits as described in Chapter 5, "Using a Tape Management System with ADSM" on page 89.

With the utility, you cannot change the library type of a new automated library to USRDFN. You must use the process described in "Setting Up to Use a Tape Management System" on page 90.

- The utility produces a macro file that contains ADSM commands. You can save this file as a record of the ADSM commands needed to recreate the server storage that you set up.

The following steps illustrate how to make some changes to the configuration shown in Figure 12 on page 63. You can do the following:

- a** You will be using enhanced capacity cartridge system tapes on the 3490E drives. You need to change the estimated capacity to match the expected capacity of 720MB. In the **Opt** column next to **CART_TAPECLASS1**, type **2**. A screen similar to Figure 13 on page 65 appears. In the field labeled **Estimated capacity**, type **720** and press Enter.

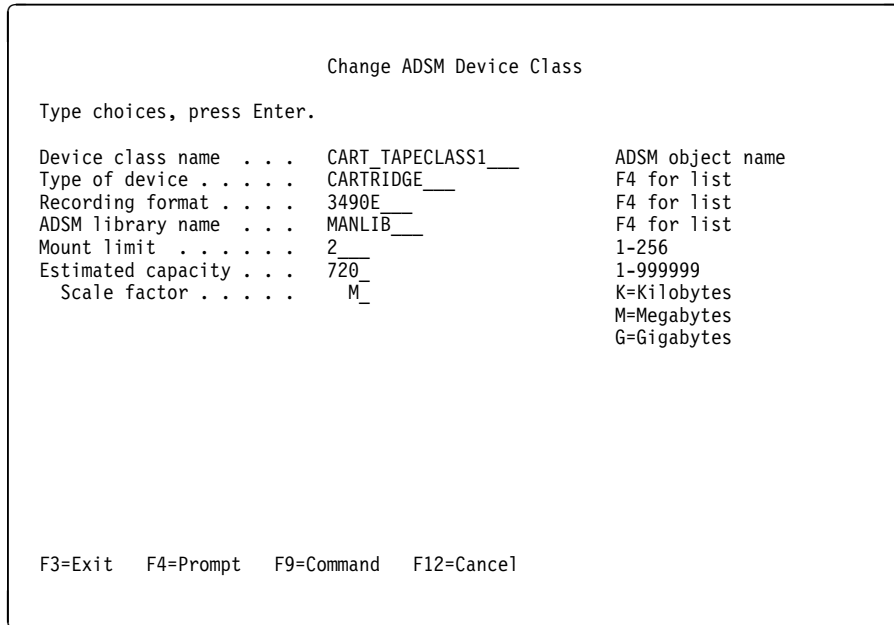


Figure 13. Change ADSM Device Class Screen

- b** You want to use scratch tapes for the CART_TAPEPOOL1 storage pool, so you need to change the tape storage pool. In the **Opt** column next to **CART_TAPEPOOL1**, type **2**. A screen similar to Figure 14 appears. In the field labeled **Maximum scratch volumes**, type in the maximum number of scratch volumes for the pool (20, for example). Press Enter.

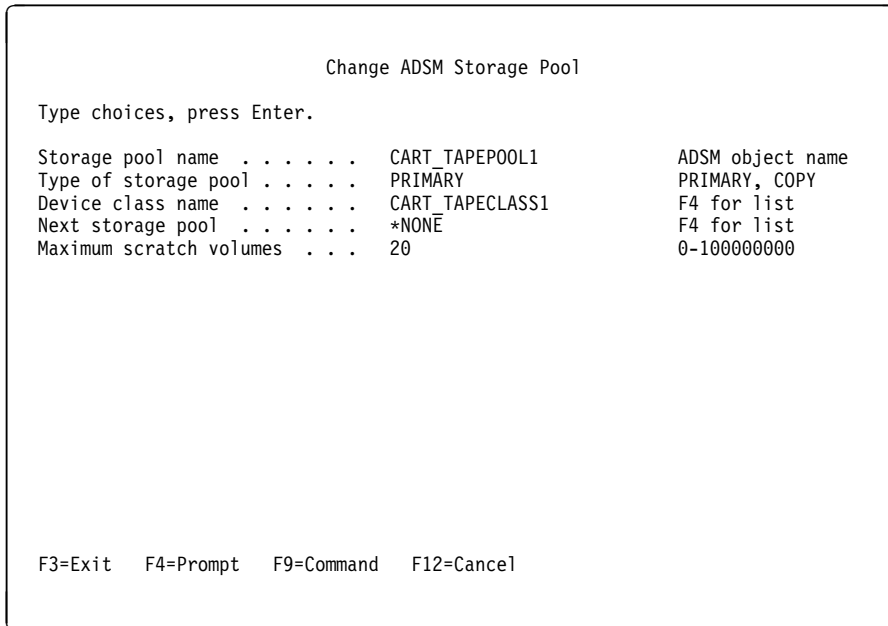


Figure 14. Change ADSM Storage Pool Screen for a Tape Storage Pool

- c** You want to have data migrate from the BACKUPPOOL disk storage pool to the new tape storage pool. To do this, you change the *next storage pool* for the BACKUPPOOL. In the **Opt** column next to **BACKUPPOOL**, type **2**. A screen similar to Figure 15 on page 67 appears. Find the field labeled **Next storage pool**. Type **CART_TAPEPOOL1** in that field or use F4 to select from a list of storage pools. Press Enter.

Press F5 to refresh the list in the Work with Devices screen. Check that the hierarchy is changed, similar to the screen shown in Figure 16 on page 67. However, the changes you see do not take effect until you use F8 to submit to the server, and the utility completes without error.

```

                                Change ADSM Storage Pool

Type choices, press Enter.

Storage pool name . . . . . : BACKUPPOOL
Type of storage pool . . . . . : PRIMARY
Device class name . . . . . : DISK
Next storage pool . . . . . : CART_TAPEPOOL1__      F4 for list

F3=Exit  F4=Prompt  F9=Command  F12=Cancel

```

Figure 15. Change ADSM Storage Pool Screen for a Disk Storage Pool

```

                                Work with Devices for ADSM

Macro output file . . . . . : MACROS__
Library . . . . . : QUSRADSM__
Macro output member . . . . . : DEVCFG__

Type options, press Enter.
2=Change  5=Display

Opt  Storage Hierarchy      Type      Status
-    ARCHIVEPOOL           STGPOOL   DEFINED
-    BACKUPPOOL            STGPOOL   DEFINED
-    CART_TAPEPOOL1        STGPOOL   NEW
-    CART_TAPECLASS1       DEVCLASS  NEW
-    MANLIB                 LIBRARY   NEW
-    TAP03                  DRIVE     NEW
-    TAP04                  DRIVE     NEW
-    SPACEMGPPOOL          STGPOOL   DEFINED

F3=Exit  F5=Refresh  F8=Submit to server  F9=Command
F10=Work with libraries  F11=Work with storage pools  F12=Cancel

Bottom

```

Figure 16. Work with Devices Screen after Changes

- 5 Select Submit to server (F8) when you are done working with the list. The utility creates a macro file that contains commands to have the ADSM objects labeled NEW in the list defined to the ADSM server, and to make changes to already-defined objects. The macro is sent to the server to be run.

Troubleshooting: If you get an error when you submit the macro, display the job log for the interactive session to understand the error. You can use F9 (Command) to bring up an AS/400 command line while still in the utility, or you can exit the utility. (If you exit the utility without successfully submitting the file to the server by using F8, all changes you made are lost.)

To display the job log, enter the following command:

```
===> dspjoblog
```

Press F10, then page up through the messages. Look for messages that have the prefix "ANS."

- 6 To see the results of running the utility, you can run the utility again, or you can use ADSM query commands from an ADSM administrative client. For example, after pressing F8 from the screen shown in Figure 16 on page 67 and starting the utility again, you might see the screen shown in Figure 17.

```

                                Work with Devices for ADSM

Macro output file . . . MACROS_____
Library . . . . . QUSRADSM____
Macro output member . . . . . DEVCFG_____

Type options, press Enter.
  2=Change  5=Display

Opt   Storage Hierarchy      Type      Status
-     ARCHIVEPOOL           STGPOOL   DEFINED
-     BACKUPPOOL            STGPOOL   DEFINED
-     CART_TAPEPOOL1        STGPOOL   DEFINED
-     CART_TAPECLASS1       DEVCLASS  DEFINED
-     MANLIB                 LIBRARY   DEFINED
-     TAP03                  DRIVE     DEFINED
-     TAP04                  DRIVE     DEFINED
-     CART_TAPEPOOL1        STGPOOL   DEFINED
-     CART_TAPECLASS1       DEVCLASS  DEFINED
-     MANLIB                 LIBRARY   DEFINED
-
F3=Exit  F5=Refresh          F8=Submit to server      F9=Command
F10=Work with libraries  F11=Work with storage pools  F12=Cancel
More...
```

Figure 17. Work with Devices Screen after Submitting the Macro

- 7 After you have successfully run the utility, you can change ADSM policy to use the new libraries and drives. For manually mounted drives, see "Update ADSM

Policy” on page 71. For drives in automated libraries, see “Update ADSM Policy” on page 76.

- 8** You must prepare volumes for use on the new drives. For manually mounted drives, see “Prepare Volumes for Use by the Library” on page 73. For drives in automated libraries, see “Prepare Volumes for Use by the Library” on page 77.

Example of a Manual Library: Setting Up Two 3490E Drives

For the following example, two 3490E drives are attached to the AS/400 and configured to the operating 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 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.

Configure the Device to the Operating System

For the details of this step for manual libraries, see *AS/400 Device Configuration Guide*.

Define the Device to ADSM

- 1** Get the drive description from your system by entering this command on an AS/400 command line:

```
==> wrkcfgsts *dev *tap
```

Record the drive description for the drives you want ADSM to use.

- 2** Define a manual library for ADSM by entering the following command on an ADSM administrative client command line. The library you are defining is *manual* because an operator must mount the tapes.

```
define library manlib libtype>manual
```

This command uses the default value for drive selection, meaning that ADSM selects the drive rather than an operator. See “Drive Selection” on page 80, “Defining Libraries” on page 110, and “MANUAL Libraries” on page 19.

- 3** Define the drives that belong to this manual library.

```
define drive manlib drive01 device=tap01
```

In this example, the ADSM name (drive01) does not match the AS/400 name for the drive (tap01). You might prefer to have the ADSM name match the AS/400 name.

Key choices: This example uses default values for the following parameters. You might want to specify different values.

AUTOLOADER

Specifies whether ADSM should try to use the next tape in an automatic cartridge loader (ACL) of a 3480 or 3490 tape drive. The default is not to use the ACL.

SHARE

Specifies whether to allow ADSM to vary the drive on and off to make drive sharing easier. The default is not to allow sharing.

See “Defining Drives” on page 112.

- 4** Classify drives according to type and format by defining ADSM device classes. For example, if you want to use the 3490E recording format, use the following command:

```
define devclass tap3490_class devtype=cartridge
format=3490e library=manlib 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, if you want drive selection done the same way for all of them. For example, you can group 8mm drives and 3490 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. If you want drive selection done differently for the different device types, you can put them into separate ADSM manual libraries.

See “Defining and Updating Device Classes for Tape” on page 116.

- 5** Check the server option MNTMSGQ to ensure that mount messages for the drives are sent where you want:
 - a. From the ADSM Main Menu, select 1 (Utilities).
 - b. Select 3 (Change server options).

For more information on server options, see *ADSM Administrator's Reference*.

6 To check what you have defined so far, use these commands:

```
query library
query drive
query devclass
```

See “Requesting Information about Libraries” on page 111, “Requesting Information about Drives” on page 113, and “Requesting Information about a Device Class” on page 123.

7 Create the storage pool to use the devices in the device class you just defined. For example:

```
define stgpool tap3490_pool tap3490_class maxscratch=20
```

Key choices:

- a. If you do not specify a value for the maximum number of scratch volumes, you must explicitly define each volume to be used in the storage pool. If you allow scratch volumes, ADSM can choose from the scratch volumes available, without action on your part.
- 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 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 144 and “How Collocation Affects Reclamation” on page 153.

See “Defining a Primary Storage Pool” on page 159.

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 216. For how to define new policy, see “Creating Your Own Storage Management Policies” on page 217.

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 to tape directly, 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.

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

```
update copygroup standard standard standard
type=backup destination=tap3490_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 223.

- 2 Activate this modified policy:

```
activate policysset standard standard
```

See “Activating Policy Sets” on page 230.

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, by using the following command:

```
update stgpool backuppool nextstgpool=tap3490_pool
```

If you have not changed the defaults for BACKUPPOOL, ADSM will migrate data from this disk pool to the TAP3490_POOL when the disk pool is 90% full. See “Defining or Updating Storage Pools” on page 159.

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. Because the STANDARD policy domain is the default, to register a client to the STANDARD policy domain, enter this command:

```
register node astro cadet
```

For information on options when registering clients, see “Administrator Registration of Client Nodes” on page 309.

Prepare Volumes for Use by the Library

- 1** Initialize and label volumes for use in the manual library. Use the OS/400 command (INZTAP), or follow these steps:
 - a. Select 1 (Utilities) from the ADSM main menu.
 - b. Select 4 (Initialize a tape) from the Utilities menu.
 - c. Repeat for all the volumes you want to label for this library.

For more information, see “Labeling Sequential Storage Pool Volumes” on page 181.

- 2** 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. For information on defining volumes, see “Defining Storage Pool Volumes” on page 183.

Example of an Automated Library: Setting up a 9427 Library

For the following example, a 9427 library containing two drives is attached to the AS/400. 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.

Configure the Device to the Operating System

For the details of this step for automated libraries, see *Automated Tape Library Planning and Management*.

Define the Device to ADSM

- 1 Get the AS/400 name for the library by using this command on an AS/400 command line:

```
===> dsptapsts
```

- 2 Select a unique category name for ADSM volumes in the library, distinct from other applications using the library. To determine the categories that are already in use, enter this command on an AS/400 command line:

```
===> dsptapcgy
```

- 3 Define the library to ADSM. For example, for a 9427 library enter the following command on an ADSM administrative client command line:

```
define library 9427 libtype=as400mlb mld=mlb01 category=adsmcat
```

For automated libraries, the library type is always AS400MLB. See “Defining Libraries” on page 110 and “AS400MLB Libraries” on page 19.

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

```
define drive 9427 tap12 device=tap12  
define drive 9427 tap13 device=tap13
```

In this example, the device is given an ADSM name that matches its AS/400 name.

OS/400 V3R6 and subsequent releases: Omit this step. ADSM drives are not defined for an automated library because the operating system handles them automatically.

See “Defining Drives” on page 112.

- 5 Classify drives according to type and recording format by defining ADSM device classes. For example:

```
define devclass 8mm_class devtype=8mm format=8700  
library=9427 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” on page 116.

- 6** Check the server option MNTMSGQ to ensure that operator intervention messages for the library are sent where you want:
 - a. From the ADSM Main Menu, select 1 (Utilities).
 - b. Select 3 (Change server options).

For more information on server options, see *ADSM Administrator's Reference*.

- 7** To check what you have defined so far, use these commands:

```
query library
query drive
query devclass
```

See “Requesting Information about Libraries” on page 111, “Requesting Information about Drives” on page 113, and “Requesting Information about a Device Class” on page 123.

- 8** Create the storage pool to use the devices in the device class you just defined. For example:

```
define stgpool 8mm_pool 8mm_class maxscratch=10
```

Key choices:

- a. If you do not specify a value for the maximum number of scratch volumes, you must explicitly define each volume to be used in the storage pool. If you allow scratch volumes, ADSM can choose from the scratch volumes available, without action on your part.
- 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 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 144 and “How Collocation Affects Reclamation” on page 153.

See “Defining a Primary Storage Pool” on page 159.

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 216. For how to define new policy, see “Creating Your Own Storage Management Policies” on page 217.

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 to tape directly, 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.

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

```
update copygroup standard standard standard
type=backup destination=8mm_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 223.

- 2 Activate this modified policy:

```
activate policysset standard standard
```

See “Activating Policy Sets” on page 230.

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, by using the following command:

```
update stgpool backuppool nextstgpool=8mm_pool
```

If you have not changed the defaults for BACKUPPOOL, ADSM will migrate data from this disk pool to the 8MM_POOL when the disk pool is 90% full. See “Defining or Updating Storage Pools” on page 159.

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. Because the STANDARD policy domain is the default, to register a client to the STANDARD policy domain, enter this command:

```
register node astro cadet
```

For information on options when registering clients, see “Administrator Registration of Client Nodes” on page 309.

Prepare Volumes for Use by the Library

Ensure that enough volumes are available to ADSM in the library. If ADSM is allowed to have 10 scratch volumes (specified in the storage pool definition), then 10 volumes must be available in the ADSM category for this library. You can make volumes available in ADSM's category either by moving volumes from other categories or by inserting new volumes into the library. If you decide to use private volumes, you must also define the volumes to ADSM.

Moving Volumes from Other Categories: You can make volumes available by moving volumes to ADSM's category.

- 1 Check for categories defined for the library. For this example, for the media library device named MLB01, enter the following command on an AS/400 command line:

```
==> wrktapctg dev(mlb01)
```

Make a note of the categories that have volumes that can be moved to ADSM's category. For example, you might have the category APPLX or *SHARE400 with volumes available.

- 2 To move volumes from another named category, such as *APPLX*, enter this command on the command line of an ADSM administrative client:

```
checkin libvolume m1b01 status=scratch search=yes category=applx
```

Inserting New Volumes That Are Initialized: You can make volumes available to ADSM by inserting new volumes that are already initialized, and moving them from the INSERT category to ADSM's category. Do the following:

- 1 Put the tapes into the library. The tapes will be put into the INSERT category.
- 2 Move the new volumes from the INSERT category by entering this command on the command line of an ADSM administrative client:

```
checkin libvolume m1b01 status=scratch search=yes category=*insert
```

Inserting New Volumes That Are Not Initialized: You can make volumes available to ADSM by inserting new volumes that are not initialized, and moving them from the INSERT category to ADSM's category. Do the following:

- 1 Put the tapes into the library. The tapes will be put into the INSERT category.
- 2 Move the tapes from the INSERT category to the ADSM category for the library. Enter this command on an AS/400 command line:

```
==> addtapctg dev(m1b01) ctg(list_of_volumes) cgy(adsmcat)
```

The tapes are moved to the ADSM category so that you can initialize them in the next step. ADSM cannot use the tapes until you complete this procedure by initializing the tapes and checking them in.

See "Informing the Server about New Volumes in a Library" on page 86.

- 3 Initialize and label the volumes. You can use the OS/400 command (INZTAP), or follow these steps:
 - a. Select 1 (Utilities) from the ADSM main menu.
 - b. Select 4 (Initialize a tape) from the Utilities menu.
 - c. Repeat for all the volumes you want to label in this library.

See "Labeling Sequential Storage Pool Volumes" on page 181.

4 Check in the volumes to ADSM.

```
checkin libvolume mlb01 status=scratch search=yes category=adsmcat
```

Using Private Volumes: 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 also be initialized and checked in. See “Defining Storage Pool Volumes” on page 183.

Configuring IBM 3590 Drives

The IBM 3590 tape drive can be used within a 3494 tape library, or outside a library (stand-alone). If you have a stand-alone 3590 drive with an automatic cartridge facility (ACF), configure the drive as an ADSM library of type AS400MLB, with one drive. By dedicating the cartridges in the ACF to ADSM, you can get unattended operations, as if it were an automated library.

Configuring non-IBM Devices

Some non-IBM devices can emulate other devices when attached to the AS/400. For example, an 8mm drive may present itself to the AS/400 as a reel device.

For devices that emulate other devices, define them to ADSM as they appear to the AS/400. If an 8mm drive appears to the AS/400 as a reel device, define its device class with a device type of REEL. Ensure that the estimated capacity of the device matches the actual capacity of the device.

To find out how a device is presenting itself to the AS/400, you can use this command:

```
==> wrkcfgsts *dev *tap
```

Create separate device classes for normal devices and emulated devices. If you have true reel devices in addition to emulated reel devices, create separate device classes for them. Each device class is likely to have a different estimated capacity.

When a non-IBM drive is improperly configured, you receive a message that no drives are available when ADSM tries to use the drive.

Notes on Operations

The following sections summarize choices and procedures you need to be aware of when operating tape 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 in two ways: by setting up a mount message queue, 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 6, “Managing Tape Operations” on page 101.

Setting Up the Message Queue for Mount Messages

If the MNTMSGQ parameter has been used in the ADSM server options, an OS/400 message queue pointed to by the MNTMSGQ parameter receives the messages.

Using the Administrative Client for Mount Messages

The server sends mount request status messages 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
```

Note: The AS/400 administrative client cannot be started in mount mode or console mode.

Handling Messages for Automated Libraries

For automated libraries, ADSM works with AS/400 and 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 the message queue, and on administrative clients that have been started with either the mount mode or console mode parameters. However, you cannot use the ADSM REPLY command to respond to these messages. For more details, see Chapter 6, “Managing Tape Operations” on page 101.

Drive Selection

For ADSM libraries of type MANUAL or USRDFN, you can decide to have ADSM or a mount operator perform drive selection. If the mount operator performs drive selection, messages are sent to the mount message queue (if defined) and to any administrative client that was started using the mount mode or console mode parameter. For information on mount messages, see “Mount Operations for Manual Libraries” and “Handling Messages for Automated Libraries.”

For an AS400MLB library, you do not have a choice about drive selection. Either ADSM selects the drive (OS/400 V3R2) or OS/400 selects the drive (OS/400 V3R6 and subsequent releases).

Collocation

Collocation is a process by which the server attempts to keep all files belonging to a client node 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 144 and “How Collocation Affects Reclamation” on page 153.

Using Automatic Cartridge Loaders

For 3480, 3490, or 3490E drives for which volumes must be mounted by an operator, you can choose whether to allow ADSM to use the drive’s automatic cartridge loader (ACL). Consider using the ACL when you know that an ADSM operation will require a certain number of scratch volumes, a number that the ACL can hold. You can define or update the drive so that ADSM can use the ACL (AUTOLOADER=YES). You can then load the ACL with scratch cartridges and ADSM will use the tapes as needed, without requiring the intervention of an operator to mount the volumes.

For example, suppose you want to back up your database using scratch volumes and you know it will require 3 tapes. You can load the ACL with initialized scratch tapes and allow ADSM to perform the backup without operator intervention.

Maintaining the Volume Inventory

You can use private or scratch volumes with ADSM. See “Using Scratch Volumes and Private Volumes.”

With ADSM, you maintain your tape volume inventory by performing the following tasks:

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

Using Scratch Volumes and Private Volumes

A scratch volume 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. 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 defined to ADSM are private volumes, and do not return to scratch when they become empty. For more information, see “Preparing Volumes for Sequential Access Storage Pools” on page 181.

For information about private and scratch volumes in automated libraries, see “Private and Scratch Volumes in Automated Libraries” on page 85.

Labeling Volumes

Sequential access volumes (other than for FILE device class) must be labeled before ADSM can use them. See “Labeling Sequential Storage Pool Volumes” on page 181.

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 the reason you want to make volumes unavailable is 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 163.

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 83.

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 83.

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 231.
- Set the server option for the expiration interval, so that expiration processing runs periodically. You can change server options by selecting 1 (Utilities) on the main ADSM menu. 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 149.

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 153.

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 354.

Managing Storage Volumes in Automated Libraries

ADSM uses a volume in a media library device (an AS400MLB library type) only if the volume is identified for use by the server and is added to the library volume inventory. When you *check in* a volume to ADSM, the volume is assigned to the server’s category in that library, and added to the library volume inventory. The server then keeps track of the volume in a *library volume inventory* that it maintains for each media library device. ADSM tracks the library volume inventory separately from the inventory of volumes for each storage pool.

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

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

To ensure that 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
Concepts:	
Using categories for volumes in automated libraries	84
Private and scratch volumes in automated libraries	85
Tasks:	
Informing the server about new volumes in a library	86
Changing the status of a volume in a library	87
Removing volumes from a library	87
Returning volumes to a library	88
Auditing a library's volume inventory	88

Using Categories for Volumes in Automated Libraries

In automated libraries such as the IBM 3494, volumes in the library are assigned a *category*. The category values are useful when multiple systems or applications share the resources of a single library. Each system or application should use a unique category for the volumes it uses and owns. Otherwise, different systems or applications may try to use the same volume, resulting in data loss or corruption.

Ensure that you assign ADSM volumes to a category that is unique for that library. For ADSM, you specify the category when you define the library to ADSM. See "Defining Libraries" on page 110.

When a volume is first inserted into the library, either manually or automatically at the convenience I/O station, the volume is assigned to an *INSERT category*. A software application, such as ADSM, can then change a volume's category name to a different value.

These category values are useful when multiple systems share the resources of a single library. Typically, a software application that uses an automated library only uses volumes that reside in a category that is reserved for the application.

When a library is defined, you must specify the category name that ADSM assigns to the volumes that it owns in the library, by specifying the `CATEGORY` parameter in the `DEFINE LIBRARY` command. If you decide to connect multiple systems that are running ADSM to a single library, you must ensure that none of the systems use the same category values. Otherwise, two or more systems may try to use the same volume, resulting in data loss or corruption.

IBM 349X Libraries: An IBM 3494 or 3495 Tape Library Dataserver has an intelligent control unit that tracks the inventory of all volumes in the library. The control unit assigns each volume to a single *category*, a hexadecimal value, which ADSM represents as the equivalent decimal value. A software application such as ADSM can interact with the library control unit to change a volume's category to a different value.

Private and Scratch Volumes in Automated Libraries

Within each ADSM category in an automated library, the library tracks 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 its category in the automated library and change its status to private, that is, in use. 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 87.

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 a scratch mount request is made in a library, the server can choose *any* volume in the library whose status code indicates that it is a scratch volume and whose category indicates it belongs to the ADSM server. After the volume is mounted, its status code is changed to private and the volume is automatically defined as part of the storage pool for which the mount request was made. 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.

A volume used to satisfy a scratch mount request automatically has its status code changed to private to prevent it from being used for ensuing scratch requests. If the volume is being used by a storage pool, it is automatically returned to the library scratch pool when all data on the volume expires or is moved to other volumes. However, if a scratch volume is used for an export, a database backup, or a database dump operation, its status code changes to private and it is not automatically returned to the scratch pool. The volume returns to the scratch pool when an 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.

Informing the Server about New Volumes in a Library

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

You inform the server that a new volume is available in a 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.

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 (manual mode), and issue the command for each volume. To check in a larger number of volumes, you can use the search capability of this command (search mode). Processing for the two modes differs:

Manual (SEARCH=NO)

ADSM checks in only a single volume. 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 drive within the library.

For a 349X library, you can use manual mode even for volumes that have already been inserted into the library through 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.

Search (SEARCH=YES)

ADSM automatically searches 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 a separate CHECKIN LIBVOLUME command for each volume. The server searches for new volumes only in the category that you specify in the command, and checks in the volumes that it finds. This restriction prevents the server from checking in volumes that are being used by another application that is accessing the library simultaneously.

When using this mode, you cannot specify a volume name.

Volume Status

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 rejects any attempt to check in a volume with scratch status when that volume already belongs to a storage pool.

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 only checks in volumes that are 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 “Labeling Sequential Storage Pool Volumes” on page 181.

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*. When you allow swapping, if 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.

Checking In Volumes for Libraries with 3590 Drives

To check in a volume for a 3590 drive, you must specify the device type for the volume (that is, 3590). If you do not specify the device type, ADSM assumes the volume is for a CARTRIDGE device (3480, 3490, or 3490E).

Check-In Processing Time

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

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, it is not possible to change the status of a volume from private to scratch if the volume is defined in a storage pool. Doing so might result in the volume being overwritten during a later scratch mount request.

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.
- You want to make a copy of a volume at a remote site.

- 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. 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 is not equipped with an entry/exit port, the mount operator is requested to 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 detects that the volume is not in the library and marks the volume as UNAVAILABLE to the storage pool.

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.
2. Update the volume's ACCESS value. You must change the access from unavailable to read/write or read-only.

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.

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).

Chapter 5. Using a Tape Management System with ADSM

You can use a media management system to manage media for ADSM storage. To do this, you define a user-defined (USRDFN) library that works with a set of user-written exit programs to integrate ADSM processing of sequential media into your media management system. Through the exits, the media management system provides volume mount, volume dismount, and volume inventory services to ADSM.

This section provides information that helps you to establish an interface between ADSM and your media management system. For the details of the interface, see Appendix, "Interface for Media Management Systems" on page 373.

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

Section	Page
Concepts:	
The ADSM exits	89
Notes on operations	97
Tasks:	
Creating an exit program	90
Setting up a media management system	90
Defining and managing exit programs	96

The Exits

ADSM communicates with tape management systems such as Backup Recovery and Media Services/400 (BRMS/400) via the four ADSM exits. The tape management system manages ADSM-defined USRDFN libraries. The exits are:

Mount

Invoked by ADSM when a volume needs to be mounted in a USRDFN library. OS/400 commands or media management system commands, or a combination, can be used in the mount exit program to perform any tasks necessary to either mount the volume or aid in the mount when the server opens the volume.

The ADSM server issues mount request messages that provide the status of the mount operation. For more information on mount messages, see "Handling Messages" on page 97.

Dismount

Invoked by ADSM when a volume that was mounted is no longer needed by ADSM. If the media management system selected a drive, this exit releases it.

Deletion

Invoked by ADSM when a volume that was acquired as scratch becomes empty, or when a volume is deleted from ADSM's database. ADSM no longer needs the volume, and the media management system takes the appropriate actions.

Expiration

Invoked by ADSM when a volume that was explicitly defined to the storage pool becomes empty, for example, after reclamation processing or data expiration. ADSM will reuse the tape from the beginning.

BRMS allows ADSM administrators to avoid dealing with checking in volumes, managing scratch volumes, and managing categories in automated libraries. The product library contains sample exits that use BRMS and its standard definitions. These sample exits can be modified if necessary and then compiled for use.

Creating an Exit Program

The exit program must be able to interpret the parameters it receives from the server and turn them into a request that your media management system can understand. If your media management system cannot use some of the parameters, you can ignore them. The exit program must also be able to understand any responses from the media management system. Finally, the exit program must be able to communicate its results to the ADSM server.

There are samples of exit programs in the ADSM product library, located in QADSM/QAANRSMP. There are two sets of samples, one written in C language and an equivalent set written in the AS/400 command language (CL). You can use them as a basis for your customized exits, or write your own exits in C, CL, or another language. Before modifying the sample exit programs, you should copy them into a different file or library. You can create a new library or you can place the exits in the work library of the server.

When you have designed your exit programs, code them in the language of your choice. For the details of the interface, see Appendix, "Interface for Media Management Systems" on page 373. Create *PGM objects from your exit program source and place them in an OS/400 library that the ADSM server has authority to access.

Setting Up to Use a Tape Management System

Before you begin the setup of the interface between ADSM and your media management system, you should understand ADSM sequential media processing and evaluate your media management system to determine what functions it can perform for ADSM. See "USRDFN Libraries" on page 20. Some key questions are:

- Who should do drive selection? The choices are:
 - ADSM (not applicable for OS/400 V3R6 and subsequent releases)
 - An operator
 - The mount exit
- What has to be done when the dismount exit is given control?

When you have decided on the design, do the following steps. For a detailed example of performing these steps, see "A Detailed Example" on page 91.

1. Create and build your exit programs. See "Creating an Exit Program."

2. Define the exit programs to ADSM. See “Defining an Exit Program to ADSM” on page 96.
3. Define one or more libraries of type USRDFN. See “Defining Libraries” on page 110.
4. For each library, define one or more drives. See “Defining Drives” on page 112.
OS/400 V3R6 and subsequent releases: Skip this step. You do not define drives.
5. Define one or more device classes that use the libraries. See “Defining and Updating Device Classes for Tape” on page 116.
6. Define one or more storage pools that use the device classes. See “Defining or Updating Storage Pools” on page 159.

A Detailed Example

Suppose you have the following system configuration:

- BRMS installed. You want BRMS to act as the media management system for the ADSM server.
- A 3494 that is shared with another system and contains two tape drives.
 - Device TAP01
 - Device TAP02
- A string of four 3480 tape drives that is *not shared*.
 - Device TAP03
 - Device TAP04
 - Device TAP05
 - Device TAP06
- A 7208-12, an 8mm tape drive.
 - Device TAP07

Setting up the exits and ADSM definitions to work with the tape management system for this example consists of these tasks:

1. “Preparing the Exit Programs” on page 92
2. “Defining the Exit Programs” on page 92
3. “Defining the Libraries” on page 93
4. “Defining the Drives” on page 94
5. “Defining the Device Classes” on page 94
6. “Defining the Storage Pools” on page 95

Preparing the Exit Programs

1. Determine the design of each of the four exit programs based on the system configuration described in “A Detailed Example” on page 91. For example, the following design decisions were made for this example:
 - The Expiration Exit program requests that BRMS expire the volume immediately.
 - The Deletion Exit program uses the BRMS CHGMEDBRM command to expire the volume, which will result in the volume returning to the scratch pool.
 - When the mount exit program is invoked:
 - The SETMEDBRM command prepares BRMS for the open volume operation.
 - When the server indicates that the exit must select a drive:
 - If the drive location is not in a 3494 library, the exit uses the device type to select a drive.

Device type	Drive name
CARTRIDGE	TAP03
8MM	TAP07
 - If the drive location is in a 3494 library, the exit allows BRMS to select the device.
 - When the dismount exit program is invoked, decide what MOVE operations to select. If the mount exit program selected a drive, release it.

The sample exit programs that come with ADSM support this example.

Optionally, BRMS V3R1 and later versions contain a set of APIs that provide capabilities that might be utilized in the exit programs. If you choose to use the BRMS APIs, you must update the QADSM job description and add QBRM to the library list.

2. Code the exit programs in the language of your choice. Create *PGM objects from your exit program source and place them in an OS/400 library that the ADSM server has authority to access. For this example, the *PGM objects are located in ADSMEXITS.

Defining the Exit Programs

1. Define the exits to the ADSM server. Each exit program requires a separate DEFINE EXIT command. From an ADSM administrative client, you can define the exit programs using the following commands:

```
define exit mount adsmexits/mount
define exit dismount adsmexits/dismount
define exit deletion adsmexits/deletion
define exit expiration adsmexits/expiration
```

- Use the QUERY EXIT command to validate the definitions of the exit programs. Figure 18 on page 93 shows the information that is displayed.

Exit Type	Exit Name	Resolved	Last Modified Date/Time
MOUNT	ADSMEXITS/MOUNT	Yes	11/01/1994 09:38:30
DISMOUNT	ADSMEXITS/DISMOUNT	Yes	10/31/1994 12:05:57
DELETION	ADSMEXITS/DELETION	Yes	10/31/1994 12:03:19
EXPIRATION	ADSMEXITS/EXPIRATION	Yes	10/31/1994 12:03:18

Figure 18. Example of an Exit Program Validation

Defining the Libraries

For this configuration, at least two ADSM libraries must be defined, one for the automated 3494 and one for the remaining manual drives.

BRMS tracks the *location* where the device is physically located. For example, the sample exits assume that the ADSM library name is also the BRMS location.

- The BRMS location of the drives in the 3494 is the name of the media library device, MLD01 for this example. MLD01 is used as the ADSM library name to create the necessary relationship between BRMS and ADSM.
- The location of the remaining drives is the BRMS default, *HOME. Because ADSM does not accept the * as part of a library name, HOME is used as the ADSM library name and the change to *HOME is handled in the mount exit program.

You can define the libraries using the following commands:

```
define library home libtype=usrdfn
define library mld01 libtype=usrdfn driveselection=exit
```

The QUERY LIBRARY command can be used to validate the definitions of the libraries. See Figure 19.

Library Name	Library Type	Drive Selection	Media Library Device	Category
HOME	USRDFN	Exit		
MLD01	USRDFN	Exit		

Figure 19. Query Library Output

Defining the Drives

OS/400 V3R6 and subsequent releases: Omit this step. You do not define drives.

Any drive you want ADSM to use must be defined, regardless of library type. For the drives in the 3494, use the SHARE option. This option permits the ADSM server to vary the devices online when their initial status is varied offline. The remaining drives are not shared so the SHARE option defaults to No.

You can define the drives in this example by using the following commands:

```
define drive mld01 tapedrive1 device=tap01 share=yes
define drive mld01 tapedrive2 device=tap02 share=yes

define drive home tapedrive3 device=tap03
define drive home tapedrive4 device=tap04
define drive home tapedrive5 device=tap05
define drive home tapedrive6 device=tap06

define drive home tapedrive7 device=tap07
```

The QUERY DRIVE command can be used to check the definitions of the drives. Figure 20 shows the output from the command.

Library Name	Drive Name	Device Type	Device	Share	Autoloader
MLD01	DRIVE01	CARTRIDGE	TAP01	YES	
MLD01	DRIVE02	CARTRIDGE	TAP02	YES	
HOME	DRIVE03	CARTRIDGE	TAP03	NO	
HOME	DRIVE04	CARTRIDGE	TAP04	NO	
HOME	DRIVE05	CARTRIDGE	TAP05	NO	
HOME	DRIVE06	CARTRIDGE	TAP06	NO	
HOME	DRIVE07	8MM	TAP07	NO	

Figure 20. Query Drive Output

Note: For USRDFN libraries, ADSM does not check whether the drive is actually in the library when the drive is defined to ADSM. If the drive is not actually in the library, the problem is not detected until a mount operation attempting to use the drive fails.

Defining the Device Classes

BRMS assigns to physical media an attribute called *media class* that denotes each unique combination of media type and capacity. The ADSM device class object includes the same characteristics of media type and capacity. To connect the BRMS media class with the ADSM device class characteristics, you can use the method in the sample exits: the name of the device class tells BRMS the media class.

To use the media class as the name of the device class for the devices in this example, define the device classes by using the following commands:


```
define devclass cart3490e devtype=cartridge format=3490e library=mld01
define devclass halfinch devtype=cartridge format=3480 library=home
define devclass fmt5gb devtype=8mm format=8500 library=home
```

Defining the Storage Pools

A storage pool name must be unique. When choosing a storage pool name, you might want to select a name that the exit program can use to determine what ADSM will use the volume for. For example, a name such as *archiveoffsite* can provide information to the exit to determine move policy, or a name can help in the selection of a drive.

You can define the storage pools using the following commands:

```
define stgpool lanservers cart3490e maxscratch=5
define stgpool archiveoffsite halfinch
define stgpool 8mmpool fmt5gb
```

Naming the storage pools: The exit programs will receive an indication only that a volume was used for an operation, not which type of operation (archive, backup, or space management of a client file, or backup of a primary storage pool). You may want to ensure that each storage pool is used for only one type of operation. To help you track what a storage pool is used for, indicate its use with the storage pool name you select.

Scratch volumes: Whether to allow scratch volumes in the storage pool is a key choice. If you do not allow scratch volumes, you must explicitly define each volume to be used in the storage pool. If you allow scratch volumes, ADSM can, via the media management system, choose from the scratch volumes available without action on your part.

Defining and Managing Exit Programs

The exit programs you write must be defined to ADSM. You can also update, query, and delete the exit program definitions.

Task	Required Privilege Class
Defining an Exit Program to ADSM	System or unrestricted storage
Updating an Exit Program	
Querying an Exit Program	
Deleting an Exit Program	

Defining an Exit Program to ADSM

To inform the server about an exit program, an administrator must issue the DEFINE EXIT command. For a USRDFN library to function properly, you must define MOUNT, DISMOUNT, DELETION, and EXPIRATION exit programs. Each exit program requires a separate DEFINE command. When issuing this command, you must provide the name of the OS/400 library which contains the exit's *PGM object, as well as the name of the exit program.

For example, to define the mount exit in the adsmexits/mount library, use the following command:

```
define exit mount adsmexits/mount
```

Updating an Exit Program

Administrators can update a defined exit program by issuing the UPDATE EXIT command. You typically use this command to notify the server of either a new exit program name or a changed exit program.

Suppose you have defined a deletion exit program to the server. You then make some modifications to the program and rebuild the *PGM object. To notify the server that it must resolve the exit program again, issue the following command:

```
update exit deletion
```

Querying an Exit Program

An administrator can query for information about one or all previously defined exit programs by using the QUERY EXIT command. Either a standard or a detailed report can be requested.

For example, the information shown in Figure 21 on page 97 might be generated if the following command is issued:

```
query exit
```

Exit Type	Exit Name	Resolved	Last Modified Date/Time
MOUNT	ADSM/QANRMOUNTC	Yes	11/01/1994 09:38:30
DISMOUNT	ADSM/QANRDSMNTC	Yes	10/31/1994 12:05:57
DELETION	ADSM/QANRDELTNC	Yes	10/31/1994 12:03:19
EXPIRATION	ADSM/QANREXPIRC	Yes	10/31/1994 12:03:18

Figure 21. Standard Query Exit Report

Deleting an Exit Program

Administrators can delete a previously defined exit program by issuing the DELETE EXIT command. Before issuing this command, however, all of the libraries of type USRDFN must be deleted.

For example, suppose you wish to delete the mount exit program. After deleting all of the USRDFN libraries, you could issue the following command to delete the exit itself:

```
delete exit mount
```

See *ADSM Administrator's Reference* for more information on this command.

Notes on Operations

The following sections summarize choices and procedures you need to be aware of when using a media management system with ADSM.

Handling Messages

The server issues mount request messages to provide the status of the mount operation. The messages are sent to all administrative clients that were started using the mount mode parameter or console mode parameter.

Another option is to use the MNTMSGQ parameter in the server options. The message queue pointed to by this parameter will also receive mount request messages. Mount operators can connect to the server from remote systems and monitor the server for required volume mount activities. However, you must use your installation's mounting procedures to respond to such mount requests.

Drive Selection

You can decide to have ADSM, a mount operator, or the mount exit program perform drive selection. If the mount operator performs drive selection, messages are sent to the mount message queue (if defined) and to any administrative client that was started using the mount mode or console mode parameter. For information on mount messages, see “Handling Messages” on page 97.

OS/400 V3R6 and Subsequent Releases: Drives are not defined to USRDFN libraries, and therefore ADSM cannot perform drive selection. Drive selection must be performed by a mount operator or by the mount exit. If the media management system cannot perform drive selection, you can use the DRIVESELECTION parameter of the DEFINE LIBRARY command to request that the ADSM server send drive selection messages to a mount operator.

Collocation

Collocation is a process by which the server attempts to keep all files belonging to a client node 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 144 and “How Collocation Affects Reclamation” on page 153.

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 99.

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 99.

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 231.
- Set the server option for the expiration interval, so that expiration processing runs periodically. You can set options through the ADSM Utilities menu or by issuing the CHGSVRADSM command (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 149.

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 153.

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 354.

Maintaining the Exits

Whenever you make a change to an exit, you must inform the ADSM server by using an UPDATE EXIT command. The changes you made then become effective.

Chapter 6. Managing Tape Operations

When the ADSM server requires a volume mount in a library, it generates a request to represent that mount. The server sends mount request status messages to all administrative clients that have been started with either the special *mount mode* or *console mode* parameter. Optionally, if the MNTMSGQ parameter has been used in the ADSM server options, an OS/400 message queue pointed to by the MNTMSGQ parameter also receives the messages.

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. It is usually not necessary for an operator 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 USRDFN and AS400MLB libraries, it is not possible for the server to automatically detect when a requested activity has been completed. Generally, an operator must perform an OS/400 or media management system action to complete the requested activity. For example, the operator may have to respond to a message on the message queue associated with a specific device description. For such requests, the ADSM REPLY command is not accepted by the server.

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.

If the MNTMSGQ parameter was used in the ADSM server options, an OS/400 message queue pointed to by the MNTMSGQ parameter receives mount request messages. Figure 22 on page 102 shows an example.

```

                                Display Messages
Queue . . . . . : ADSMMSGQ          System: TUC400H
Library . . . . : QUSRADSM         Program . . . . : *DSPMSG
Severity . . . . : 00              Delivery . . . . : *HOLD

Type reply (if required), press Enter.
ADSM request 0001: Mount volume DSM001 on device TAP01.
Reply . . . _____

F3=Exit          F11=Remove a message      F12=Cancel
F13=Remove all   F16=Remove all except unansw     F24=More keys
Bottom

```

Figure 22. Example of Messages on a Mount Message Queue

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 TAP01 (TAP01) of library
MANLIB 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 OS/400 mount message queue or by using the ADSM REPLY command.

If the MNTMSGQ parameter was used in the ADSM server options, an OS/400 message queue pointed to by the MNTMSGQ parameter receives an inquiry message that permits a reply to the server's request. Figure 22 on page 102 shows an example of such a message. If you put your cursor on the request for the volume mount and press F9 (for additional message information), the screen shown in Figure 23 appears. You can type your reply on the screen, as shown.

```

Additional Message Information
Message ID . . . . . : ANR8100      Severity . . . . . : 99
Message type . . . . . : Inquiry
Date sent . . . . . : 02/13/96      Time sent . . . . . : 16:10:11

Message . . . . . : ADSM request 0001: Mount volume DSM001 on device TAP01.
For request 0001, the ADSM server requires the mounting of 8MM volume DSM001
R/W on device TAP01 within 60 minutes. The value of the volume name can be
a specific volume or a nonspecific volume (scratch). If the volume name is
SCRATCH, EXPORT.n, or DUMP.n, mount a scratch volume. The value "n" denotes
the sequence number of the scratch volume being requested for DUMP DB or
EXPORT operations.
Do one of the following,
1) Put volume DSM001 on device TAP01, and type R to acknowledge the mount.
2) Type C to cancel the request.
3) Type P to cancel the request and to indicate the volume is
More...

Type reply below, then press Enter.
Reply . . . . r_____

F3=Exit  F6=Print  F9=Display message details  F12=Cancel
F21=Select assistance level

```

Figure 23. Responding to an Operator Request Using the Message Queue

If you are not using the mount message queue to monitor mount messages, reply to ADSM requests using 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 TAP01:

```
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 fail the operation that needed the requested volume.

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 101.

You can also specify the PERMANENT parameter if the requested volume is to be marked 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 USRDFN and AS400MLB libraries, an operator must perform an OS/400 or media management system action to cancel the requested mount. For example, the operator may have to cancel the request from the message queue associated with a specific device description. For such requests, the ADSM CANCEL REQUEST command is not accepted by the server.

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 may keep it mounted for a time to reduce the access time if it is needed again. An administrator can explicitly request that such a 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. If the associated library is of type USRDFN, the dismount exit program is also invoked.

The amount of time that the server keeps a volume mounted after the volume becomes idle is set in the device class. See "Mount Retention Period" on page 117.

Chapter 7. 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 Tape Devices with ADSM" on page 59 and Chapter 5, "Using a Tape Management System with ADSM" on page 89.

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

Section	Page
Concepts:	
How ADSM Uses Sequential Access Devices	108
Tasks:	
Defining and Managing Libraries	110
Defining and Managing Drives	112

Some tasks presented in this chapter can be performed using either graphical user interface (GUI) or the command line interface. Table 7 on page 33 shows whether a task can be performed on the GUI, the command line interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

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 24 shows these relationships.

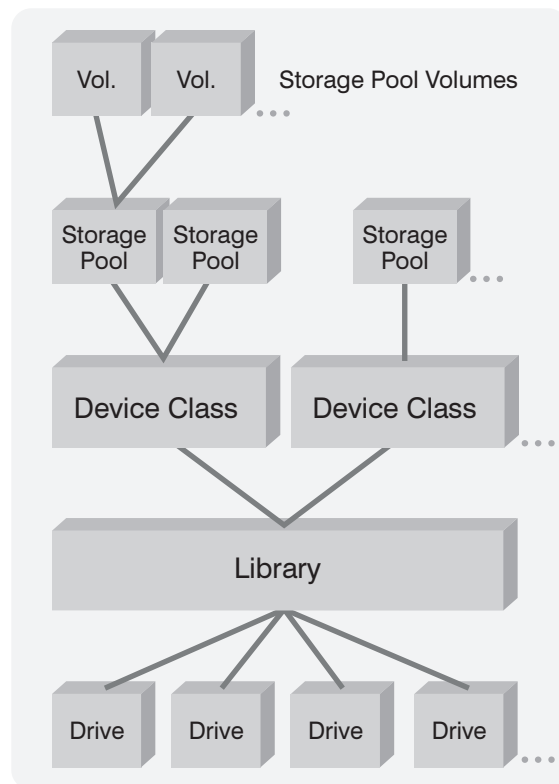


Figure 24. Relationships between Storage and Device Objects

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.

Based on the type of library, drive selection is performed by either ADSM or some predetermined external method.

- If ADSM is to perform *drive selection*, the server evaluates both ADSM and OS/400 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, provided the *SHARE* parameter of the drive allows this to occur.
 - 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.
- If ADSM is not performing *drive selection*, the server will evaluate the attributes of the drive selected by some predetermined external method. If the drive is acceptable, it will be used for the 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 user-defined, the server invokes the mount exit program. OS/400 or media management system commands can be used in the mount exit program to perform any tasks necessary to either mount the volume or aid in the mount when the server opens the volume.
- If the library is automated, the server utilizes appropriate OS/400 commands 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 code of *SCRATCH*. The volume inventory is established and managed by using the commands described in "Managing Storage Volumes in Automated Libraries" on page 83. Volume status codes are described in "Private and Scratch Volumes in Automated Libraries" on page 85. If a scratch volume is found, its volume status code 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 code 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 simply ejects the volume from the drive so that a mount operator can place it in an appropriate storage location.

- If the library is user-defined, the server ejects the volume from the drive and invokes the dismount exit program. The dismount exit program can perform any cleanup tasks required by the media management system.
- If the library is automated, the server utilizes appropriate OS/400 commands 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. Once 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 19, “AS400MLB Libraries” on page 19, and “USRDFN Libraries” on page 20.

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. 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 a library managed by a media management system installed on your OS/400, you use the DEFINE LIBRARY command to define a USRDFN library. You could define a library named USRDFNMOUNT for the drives using the following command:

```
define library usrdfnmount libtype=usrdfn
```

For automated libraries, you use the DEFINE LIBRARY command to define an AS400MLB library. For example, if you have an IBM 3494 Tape Library Dataserver with a media library device name of MLD01, you could define a library named AUTOMOUNT as follows:

```
define library automount libtype=as400mlb category=adsmcat mld=mld01
```


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, the information shown in Figure 25 might be generated if the following command is issued:

```
query library
```

Library Name	Library Type	Drive Selection	Media Library Device	Category
HOME	USRDFN	Exit		
8MMLIB	MANUAL	Automatic		
MANLIB	MANUAL	Operator		
AUTOLIB	AS400MLB	AS400MLB	MLD01	TUC400I/ADSMTAPES

Figure 25. 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.

AS400MLB Libraries

The only attribute on an AS400MLB library that can be updated is the MLD attribute. This might be necessary if your system is reconfigured, causing the MLD name to change. For example, you have defined an AS400MLB library named ROBOTMOUNT, 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 robotmount mld=mld04
```

MANUAL or USRDFN Libraries

Drive selection is the only attribute that can be updated on a MANUAL or a USRDFN library. For example, you have defined a MANUAL library named MANUALMOUNT for which ADSM is currently performing drive selection. You now decide that you would like your mount operator to perform drive selection for ADSM. You can then issue the following command to inform the ADSM server of the change:

```
update library manualmount driveselection=operator
```

You have a USRDFN library named USRDFNMOUNT for which a mount operator is performing drive selection. You now decide that you would like your mount exit program to perform drive selection for ADSM. You can then issue the following command to inform the ADSM server of the change:

```
update library usrdfnmount driveselection=exit
```

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 114.

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

OS/400 V3R6 and Subsequent Releases: The following information on defining drives applies only to MANUAL type libraries.

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, or, for manual libraries, to which you have assigned it.

Device name

The device name to be used to access the drive.

Device sharing

Whether to allow ADSM to share the device with other systems. If you allow device sharing, ADSM will vary the device on when needed, and vary it off when done. If you do not allow device sharing, ADSM will not vary on the device when it is offline. The default is not to allow sharing.

Autoloader

Whether to allow ADSM to use the next tape in an automatic cartridge loader (ACL) on a 3480 or 3490 drive. The default is not to use the ACL.

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

```
define drive manlib drive01 device=tap01
```

In this example, the ADSM name (drive01) does not match the OS/400 name for the drive (tap01). You might prefer to have the ADSM name match the OS/400 name.

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
```

Figure 26 on page 114 shows an example of the results of this command.

Library Name	Drive Name	Device Type	Device	Share	Autoloader
8MMLIB	DRIVE08	8MM	TAP08	YES	
MANLIB	DRIVE01	CARTRIDGE	TAP01	NO	NO
3490LIB	DRIVE04	CARTRIDGE	TAP04	NO	YES
QICLIB	DRIVE06	QIC	TAP06	NO	

Figure 26. Standard Query Drive Report

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.

A drive cannot be updated if it is currently in use. If a drive has a volume mounted, but the volume is currently idle, it can be explicitly dismounted as described in “Dismounting an Idle Volume” on page 105.

For example, suppose you have a drive TAP01 for which you did not allow device sharing, and you now want to allow device sharing. Enter the following command:

```
update drive manlib tap01 share=yes
```

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, it can be explicitly dismounted as described in “Dismounting an Idle Volume” on page 105.

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

Chapter 8. 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:

8MM	8mm tape drives, such as IBM 7208-2, 7208-12, and 6390 drives
QIC	Quarter-inch cartridge tape drives, such as the IBM 9346 and 6368 drives
3590	IBM 3590 tape drives
CARTRIDGE	Cartridge tape drives, such as IBM 3480, 3490, and 3490E drives
REEL	Half-inch reels, such as IBM 9347, 9348, and 2440
FILE	Storage volumes that are files in the file system of the server machine

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

Section	Page
Tasks:	
Defining and Updating Device Classes for Tape (8MM, QIC, 3590, CARTRIDGE, and REEL)	116
Defining and Updating FILE Device Classes	122
Filling Volumes to Capacity	124
Requesting Information about a Device Class	123
Deleting a Device Class	124

Most tasks presented in this chapter can be performed by using either the graphical user interface (GUI) or the command line interface. Table 8 on page 34 shows whether a task can be performed on the GUI, the command line interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

Defining and Updating Device Classes for Sequential Media

Task	Required Privilege Class
Defining and updating device classes	System or unrestricted storage

You can define and update multiple device classes for each device type. The following sections show how to define the device classes for each supported device type.

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:

- 8MM
- QIC
- 3590
- CARTRIDGE
- REEL

For information about defining drives and libraries, see Chapter 7, “Defining Drives and Libraries” on page 107.

Defining and Updating Device Classes for Tape

To use 8mm, quarter-inch, reel, IBM 3590, or IBM 34xx tape drives and cartridges, you must define a device class whose device type is 8MM, QIC, REEL, 3590, or CARTRIDGE, respectively. Do this by issuing a DEFINE DEVCLASS command with the DEVTYPE parameter. Other parameters specify how to manage server storage operations involving the new device class:

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

Mount Limit

When defining a device class, 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 given device class.

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.

- Do you want reclamation to occur?

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

- How frequently does migration or reclamation occur?

If the server is using all available drives to complete server processes such as migration or reclamation, users may have to wait until a drive becomes available before they can recover data from a storage pool. See “When Files Are Migrated” on page 139 and “Space Reclamation for Sequential Access Storage Pools” on page 149 for information on migration and reclamation.

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

Notes:

1. ADSM cannot share drives between multiple device classes.
2. Operations with high priority like database backup (BACKUP DB) or retrieve can automatically cancel lower priority operations like reclamation or backup if all of the drives associated with the device class are in use. If this happens often, 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

You can use the *MOUNTWAIT* parameter to specify 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

You can use the *MOUNTRETENTION* parameter to specify 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 via 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 operations that are using the drive and then dismount the volume. For example, you can cancel server migration or backup operations. For how to cancel processes and dismount volumes, see “Canceling Server Processes” on page 269 and “Dismounting an Idle Volume” on page 105.

Tape Label Prefix

By using the *PREFIX* parameter, you can specify a prefix value that is used to construct the *data set name* string that is stored in the label area of each tape volume. This data set name field is not used by the ADSM server, but it may facilitate the use of ADSM tapes on foreign systems, such as MVS, with tape management systems that use the data set name field. This process may be valuable when ADSM tapes are being used to export data from one system to another.

The prefix string is used as the high-level qualifier of the data set 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 show the values supported for the tape device classes.

Device Class	Reference
8MM	Table 18 on page 119
QIC	Table 19 on page 119
3590	Table 20 on page 120
CARTRIDGE	Table 21 on page 120
REEL	Table 22 on page 121

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.

8MM Device Classes

Table 18. Recording Format and Default Estimated Capacity for 8mm Tape Volumes

Recording Format	Estimated Capacity	Description
8200	2472MB	Basic recording format for an 8mm tape drive. This format yields a capacity of approximately 2.3GB on a standard 112-meter tape cartridge.
8500	5.0GB	Enhanced recording format for 8mm tape drives. This format yields a capacity of approximately 5.0GB on a standard 112-meter tape cartridge.
8700	7.0GB	Enhanced recording format for 8mm tape drives. This format yields a capacity of approximately 7.0GB on a standard 112-meter tape cartridge.
DRIVE	—	Lets the server select the recording format to use based on the drive on which the volume is mounted.

QIC Device Classes

Table 19. Recording Format and Default Estimated Capacity for QIC Tape Volumes

Recording Format	Estimated Capacity	Description
120	150MB	Specifies the 120 recording format that is used for quarter-inch cartridge tapes, resulting in a capacity of approximately 150MB
525	525MB	Specifies the 525 recording format that is used for quarter-inch cartridge tapes, resulting in a capacity of approximately 525MB
1000	1.0GB	Specifies the 1000 recording format that is used for quarter-inch cartridge tapes, resulting in a capacity of approximately 1GB
2000	2.0GB	Specifies the 2000 recording format that is used for quarter-inch cartridge tapes, resulting in a capacity of approximately 2GB
DRIVE	—	Lets the server select the recording format to use based on the drive on which the volume is mounted.

3590 Device Classes

Table 20. Recording Format and Default Estimated Capacity for 3590 Tape Volumes

Recording Format	Estimated Capacity	Description
3590*	10GB	Specifies that ADSM writes data using the basic format or, if available, compacted recording format. This results in a tape capacity of approximately 10GB. If the compacted recording format is used, the actual capacity may be greater, depending on the effectiveness of compression.
DRIVE	—	Lets the server select the recording format to use based on the drive on which the volume is mounted.

Note: An asterisk * indicates recording formats that support compaction. Because ADSM cannot determine the extent to which compaction increases the capacity of a particular recording format, ADSM does not increase the estimated capacity for recording formats that support compaction.

CARTRIDGE Device Classes

Table 21. Recording Format and Default Estimated Capacity for CARTRIDGE Tape Volumes

Recording Format	Media Type	Estimated Capacity	Description
3480	CST	180MB	18-track basic recording format
3490	CST	180MB	18-track basic recording format
3490E	CST	360MB	36-track basic recording format
3490E	ECCST	720MB	36-track basic recording format
DRIVE	—	—	Lets the server select the recording format to use based on the drive on which the volume is mounted.

REEL Device Classes

Table 22. Recording Format and Default Estimated Capacity for REEL Tape Volumes

Recording Format	Estimated Capacity	Description
1600	44MB	Specifies that the reel tape device can read and write 1600 bits per inch
3200	82MB	Specifies that the reel tape device can read and write 3200 bits per inch
6250	156MB	Specifies that the reel tape device can read and write 6250 bits per inch
DRIVE	—	Lets the server select the recording format to use 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 server uses the estimated capacity to determine how much data to write on a volume. 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 “Filling Volumes to Capacity” on page 124.

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
8MM	Table 18 on page 119
QIC	Table 19 on page 119
3590	Table 20 on page 120
CARTRIDGE	Table 21 on page 120
REEL	Table 22

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 FILE Device Classes

The FILE device type is used for special device classes whose storage volumes are not physical units, such as tape 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.

Because each volume in a FILE device class is actually a file, a volume name is a fully qualified file name string.

Note: Do not use the CRTVOLADSM command to create and format the volumes to be used by a device class with a device type of FILE. These volumes are created automatically.

When you define the FILE device class, you can supply additional parameters. The following parameters are used to instruct ADSM how to manage server storage operations for the new device class:

- MOUNTLIMIT
- MAXCAPACITY
- OLIBRARY

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 116.

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.

Specifying an Object Library for Volumes

By using the OLIBRARY parameter of the DEFINE DEVCLASS command, you can specify the library in the file system where scratch volumes are created. This library

name is also used to generate a fully-qualified file name for a volume when only a partially qualified name is provided to the server in an administrative command.

Requesting Information about a Device Class

Task	Required Privilege Class
Request information about device classes	Any administrator

You can choose to view a standard or detailed report. The default is a standard report.

To query the server to view a standard report on device classes, enter:

```
query devclass
```

Figure 27 is an example of a standard report for device classes.

Device Class Name	Device Access Strategy	Storage Pool Count	Device Type	Format	Est/Max Capacity (MB)	Mount Limit
DISK	Random	9				
TAPE8MM	Sequential	1	8MM	8200	2,472.0	2

Figure 27. Example of a Standard Device Class Report

To query a server to view a detailed report for the TAPE8MM device class, enter:

```
query devclass tape8mm format=detailed
```

Figure 28 on page 124 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: 01/05/1994 16:02:13

```

Figure 28. 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 when:

- No storage pools are assigned to the device class. For information on deleting storage pools, see “Deleting a Storage Pool” on page 175.
- The device class is not being used by an export or import process.

Note: You cannot delete the DISK device class from the server.

Filling Volumes to Capacity

The device class contains an ESTCAPACITY parameter that indicates the estimated capacity for sequential volumes associated with the device class through the storage pool. If the ESTCAPACITY parameter is not specified on the DEFINE DEVCLASS command, a default value is set based on the DEVTYPE parameter of the device class.

When a volume is mounted, ADSM checks the device class that is associated with the storage pool to determine the estimated capacity for the volume. The server attempts to write to the volume until it has reached this estimated capacity.

You can either accept the default estimated capacity for a given device class, or explicitly specify an estimated capacity. You may want to change the estimated capacity if:

- Data compression is being performed by the drives
- You have volumes of nonstandard size

- You want to restrict the amount of data on each volume

You can initially use the default estimated capacity for a device class, then begin checking volumes when ADSM has filled them to this estimated capacity. If ADSM is not filling volumes to their actual capacity, you may want to increase the estimated capacity by updating the device class.

Be careful not to specify an estimated capacity that exceeds the actual capacity of the volumes in the device class. If you choose an estimated capacity that permits the server to write to the physical end of the volume, an error occurs when the server is writing to a volume and reaches the end of the tape. The server stops writing to the volume and sets the status of the volume to read-only. ADSM considers the volume as still filling with data, when actually the volume is full. A message, ANR8263W, is issued to notify you of the volume's condition.

For how to recover from the ANR8263W error, see "Recovering from Error ANR8263W" on page 127.

Tape Volume Capacity and 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, and by the device where the file is finally stored.

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 29 on page 126 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. What the results are depends on the estimated capacity you have specified in the device class and whether client compression and drive compression are used. Table 23 on page 126 summarizes the results. In two cases, there is a conflict between what ADSM reports in the database and the actual status of the tapes.

Table 23. Tape Capacity Variations When a Client Stores a 2.4GB File

Tape Physical Capacity	Tape Estimated Capacity (in Device Class)	Client Compression	Drive Compression	ADSM Database: File Size	ADSM Database: Tape Status
1.2GB	2.4GB	On	On or off	1.2GB	One tape, half full (Tape is actually full)
1.2GB	1.2GB	On	On or off	1.2GB	One tape, full
1.2GB	2.4GB	Off	On	2.4GB	One tape, full
1.2GB	1.2GB	Off	On	2.4GB	Two tapes, full (Tapes are actually filled to only one-half of physical capacity)

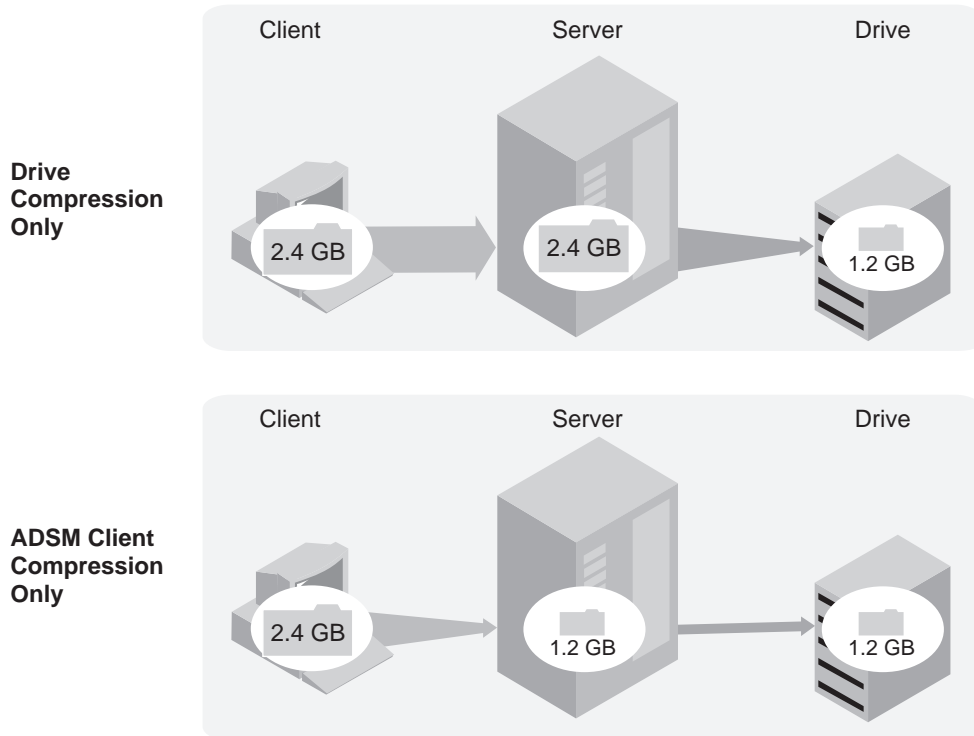


Figure 29. Comparing Compression at the Client and Compression at the Device

For how to set up compression on the client, see “User Registration of Client Nodes” on page 309 and “Administrator Registration of Client Nodes” on page 309.

Recovering from Error ANR8263W

If you receive message ANR8263W, you chose an estimated capacity for a device class that permits the server to write to the physical end of a volume. The error appears when the server reaches the end of a tape while writing to it. The server stops writing to the volume and sets the status of the volume to read-only. The message ANR8263W is issued to notify you of the volume’s condition.

To correct the problem, you must reduce the estimated capacity for the device class that the volume is associated with through the storage pool. The following is a summary of the steps to take:

1. Use the QUERY VOLUME command to view the actual capacity of the volume after it is full, and the estimated capacity for the device class.
2. Calculate the new estimated capacity based on the actual capacity of the volume.
3. Use the UPDATE DEVCLASS command to reduce the estimated capacity for the device class.

Do not use the UPDATE VOLUME command to change the access mode of the volume that caused the error. The changed access mode does not fix the problem, and only results in an immediate dismount of the volume when the server attempts to write to it. The volume eventually becomes empty as files expire or are deleted. When the empty volume is reused, the new estimated capacity takes effect.

4. If you receive the ANR8263W message for another volume, then you must step through this process again. This time you should consider a larger reduction in the computed estimated capacity.

An Example of Recovering from ANR8263W

For this example, assume that the ADSM server has mounted volume QIC1, which has an estimated capacity of 200MB. This capacity is greater than the volume’s actual capacity of 120MB and eventually the physical end of the volume is encountered. Message ANR8263W is issued by the server and the volume is dismounted.

The following describes how to respond to this situation.

1. Issue the QUERY VOLUME command to determine the status and estimated utilization of the volume:

```
query volume qic1 format=detailed
```

Figure 30 on page 128 shows the information that is displayed.

```

Volume Name: QIC1
Storage Pool Name: BRMS
Device Class Name: QIC120
Estimated Capacity (MB): 200.0
    %Util: 61.4
Volume Status: Filling
    Access: Read-Only
Pct. Reclaimable Space: 0.0
    Scratch Volume?: Yes
    In Error State?: Yes
Number of Writable Sides: 1
Number of Times Mounted: 109
    Write Pass Number: 1
Approx.Date Last Written: 03/17/1995 11:50:08
    Approx. Date Last Read: 03/17/1995 11:18:10
    Number of Write Errors: 1
    Number of Read Errors: 0
Last Update by (administrator):
    Last Update Date/Time: 03/17/1995 11:17:50

```

Figure 30. Detailed Query Volume Report

2. Use the estimated capacity value and the utilization of the volume listed in the QUERY VOLUME command output to calculate a new ESTCAPACITY value:

```
Estimated Capacity * %Util = New estimated capacity
```

For the example shown in Figure 30, the calculation is as follows:

```
200.0MB * .614 = 122.8MB
```

You should consider lowering the resulting value to allow for some variation from volume to volume because of data compression or volume size. In this example, we choose a new estimated capacity of 120MB.

3. Use the UPDATE DEVCLASS command to reduce the ESTCAPACITY value for the device class.

```
update devclass qic120 estcapacity=120m
```

After you change the estimated capacity of the device class, no further actions are necessary for the volume that initially encountered the warning condition. The volume's status and estimated capacity remain unchanged, but normal ADSM processing eventually returns the volume to an empty state as files on the volume expire or are deleted. Once empty, the volume can be filled again, this time based on the new estimated capacity.

Note: Do not use the UPDATE VOLUME command to change the access mode of the volume.

To understand this situation, use the following command:

```
query volume devclass=qic120
```

Figure 31 shows the information that is displayed. The volume QIC1 still has an estimated capacity of 200MB, but the new volume QIC2 has the new estimated capacity of 120MB.

Volume Name	Storage Pool Name	Device Class Name	Estimated Capacity (MB)	%Util	Volume Status
QIC1	BRMS	QIC120	200.0	61.4	Filling
QIC2	BRMS	QIC120	120.0	100.0	Full

Figure 31. Standard Query Volume Report

Chapter 9. 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.

Section	Page
Concepts:	
Storage pools	132
Assigning volumes in storage pools	135
Storage pool hierarchy	135
Server migration of files	139
Cache on disk storage pools	143
Collocation on sequential access storage pools	144
Space reclamation on sequential access storage pools	149
Expiration processing	154
How restore processing works	155
Tasks:	
Estimating space needs for storage pools	159
Defining or updating storage pools	159
Backing up storage pools	163
Using copy storage pools to improve data availability	165
Monitoring the use of storage pools	166
Deleting storage pools	175
Restoring storage pools	176

Most tasks presented in this chapter can be performed by using either the graphical user interface or the command-line interface. Table 9 on page 35 shows whether a task can be performed on the graphical user interface, the command line interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

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 163 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 363)
- The primary file is stored on a volume that is UNAVAILABLE or DESTROYED
- The 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 176, “Using Copy Storage Pools to Improve Data Availability” on page 165, “Recovering a Lost or Damaged Storage Pool Volume” on page 369, and “Maintaining the Integrity of Files” on page 363.

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 32 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 135.

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 163.

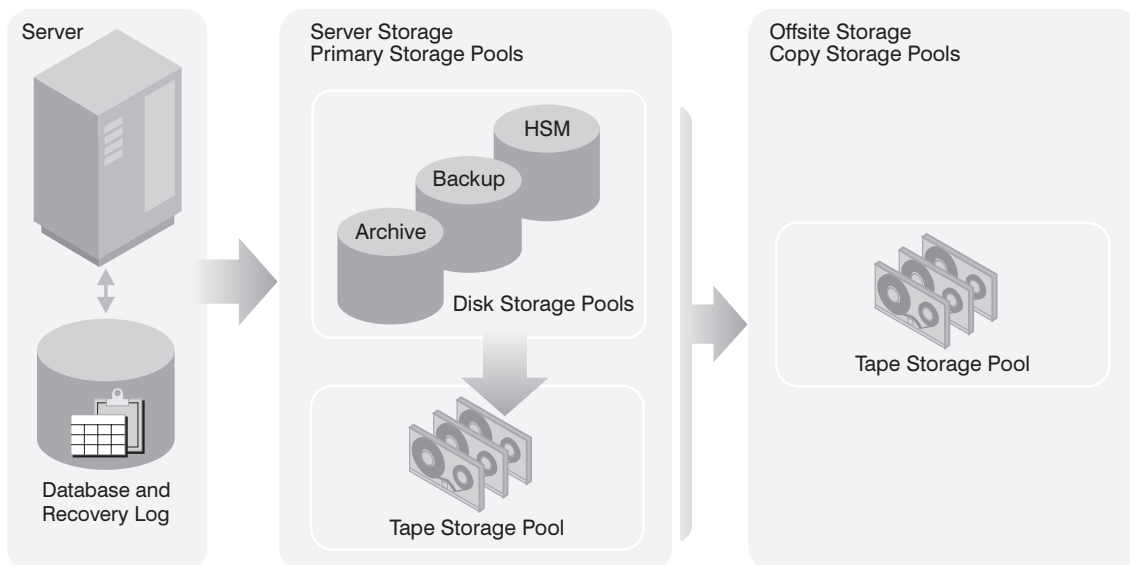


Figure 32. Example of Server Storage

Comparing Primary and Copy Storage Pools

Table 24 compares the characteristics of primary and copy storage pools.

Table 24. 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 151.
File deletion	Files are deleted: <ul style="list-style-type: none"> • 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 are damaged and no other copies of the file exist 	Files are deleted: <ul style="list-style-type: none"> • 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 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 10, “Managing Storage Pool Volumes” on page 179 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. This process may be useful when you want to establish a volume naming scheme for ADSM volumes. See Chapter 10, “Managing Storage Pool Volumes” on page 179 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.

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. This prevents the volumes from being deleted from the database and makes it easier to determine which volumes should be returned to the onsite location. The volume is not returned to the scratch pool until the access value is changed to READWRITE, READONLY, or UNAVAILABLE. This allows the administrator to query ADSM for empty offsite copy storage pool volumes and return them to the onsite location.

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 the maximum file size allowed is in the storage pool
- What the high migration threshold is for the storage pool
- If 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 33 on page 137.

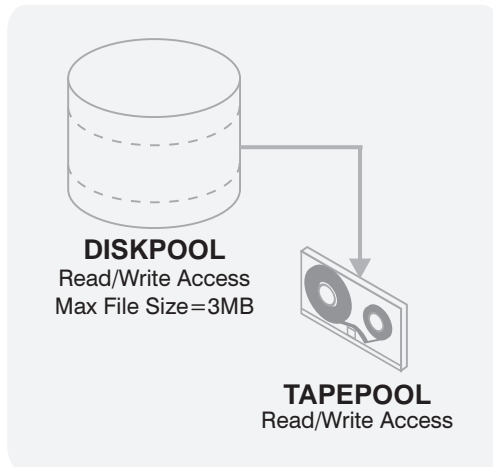


Figure 33. Storage Hierarchy, Read/Write Access, and Maximum File Size

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 33.

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 specified archive storage destination.
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 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 345 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 useful 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.

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 primary sequential storage pools 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 156 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 159 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 143 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 25 on page 140 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 25. 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

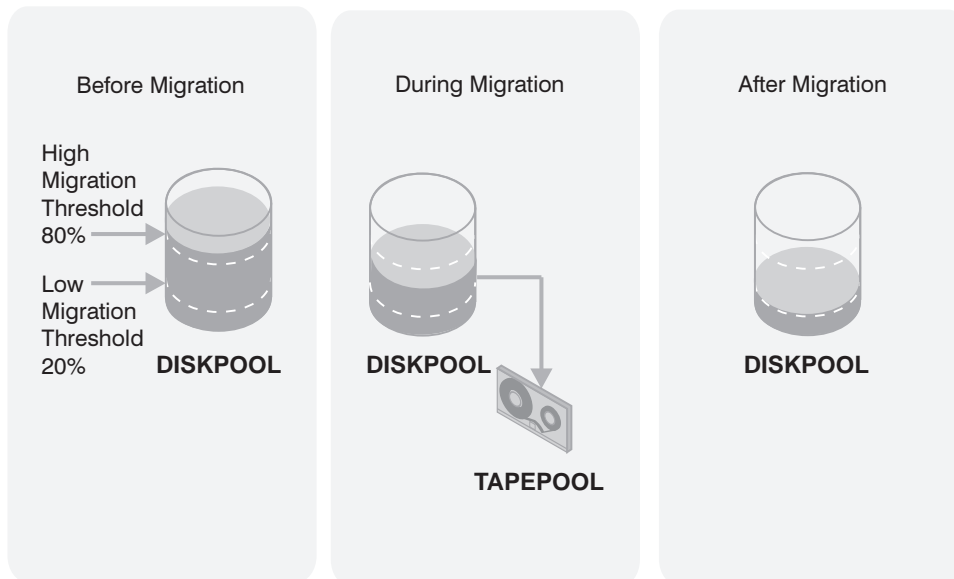


Figure 34. The Migration Process and Migration Thresholds

Figure 34 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 143.

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, without having migration occur so frequently that the device is unavailable for other use.

To calculate the high-migration threshold, consider:

- The amount of storage capacity provided for each storage pool
- The amount of free storage needed for users to add to existing files, without having migration occur

To calculate the low-migration threshold, consider:

- The amount of free disk storage space needed for normal daily processing
- Whether to use cache on disk storage pools to improve the retrievability of data
- How frequently you want migration to occur, based on the availability of sequential access storage devices and mount operators

An alternative to choosing a single high migration threshold value is to set the value to 100%, then schedule a command to lower the threshold to control when migration occurs. By scheduling when migration occurs, you can choose a time when your tape drives and mount operators are available for the operation.

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.

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 underlying devices
- 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 159.

- 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.

For information about enabling or disabling collocation for sequential access storage pools, see “Collocation on Sequential Access Storage Pools” on page 144.

- 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 149.

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 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.

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: When caching is used, the occupancy value that the server compares against the migration thresholds (%Migr) does *not* include space occupied by cached copies of files. The space utilization of the pool (%Util), however, *does* include the space used by any cached files in the storage pool. For more information on storage pool statistics, see “Monitoring the Use of Storage Pools” on page 166.

When Not to Use Cache

Do not use cache if you have limited database space. 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.

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.

If you disable cache, you may want to set higher migration levels for the disk storage pool to keep files on disk longer by preventing migration from occurring too frequently.

Collocation on Sequential Access Storage Pools

Collocation is a process in which the server attempts to keep all files belonging to a client node on a minimal number of sequential access storage volumes.

To have ADSM collocate data when files from different client nodes are mixed in the same storage pool, set collocation to *yes* when you define or update a sequential storage pool. 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 35 on page 145 shows an example of collocation enabled, with three clients having separate volumes assigned for each client.

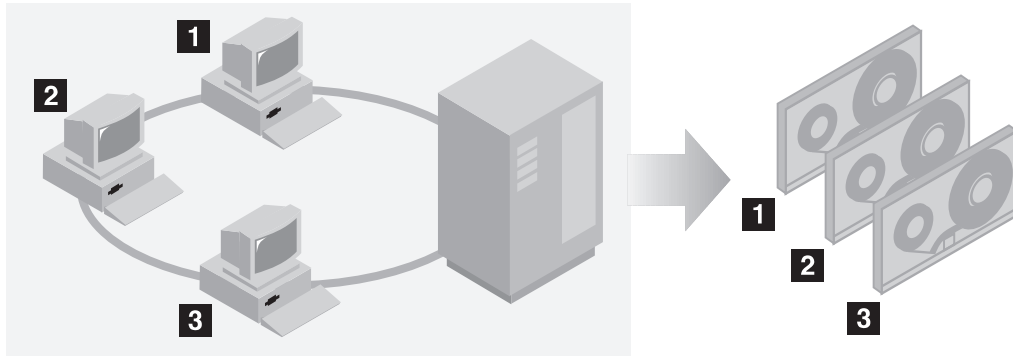


Figure 35. Example of Collocation Enabled

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 36 shows an example of collocation disabled, with three clients sharing space on a volume.

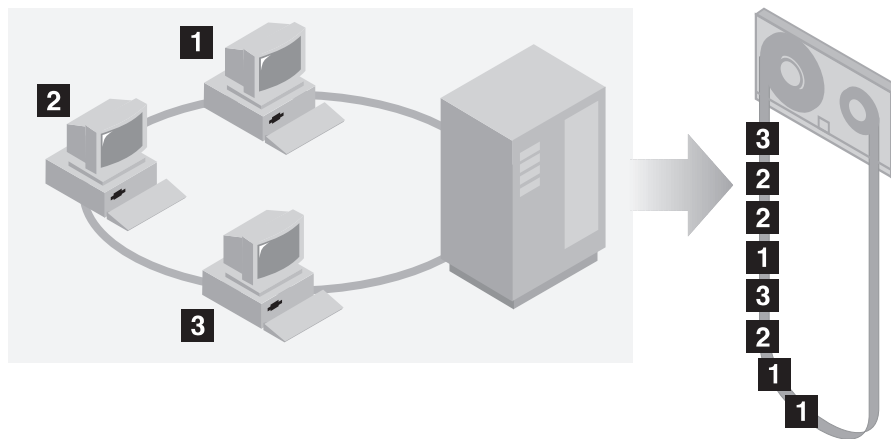


Figure 36. Example of Collocation Disabled

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 underlying device is a tape, the access time is long, because volumes must be inserted into the appropriate type of drive via manual intervention or robotic load. 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 need to restore or retrieve a large number of files, 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 which 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 and users back up, archive, or migrate files to sequential access storage, the server attempts to select a volume that already contains files from file spaces belonging to the client node.

If no such volume exists, the server attempts to select an empty volume. The server first selects volumes that have been explicitly defined in the storage pool. If no predefined volumes exist, but scratch volumes are supported for the storage pool, the server attempts to select a scratch volume.

If no empty volume exists and no scratch volume can be obtained, the server selects the emptiest volume that already contains 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 without mixing user files on multiple volumes. For example, Figure 37 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 data for a single node.

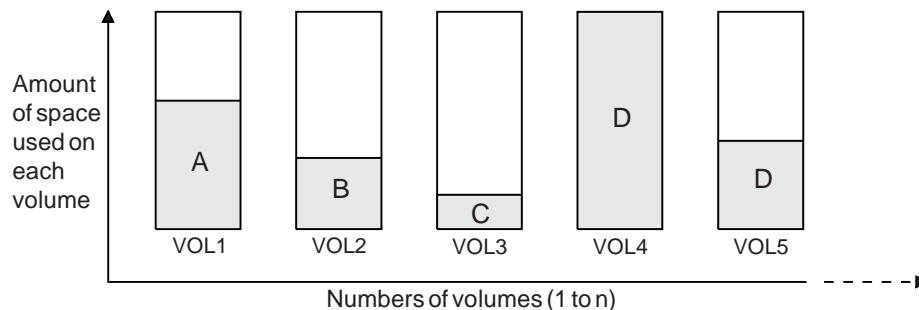


Figure 37. Using All Available Sequential Access Storage Volumes with Collocation Enabled

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 has been disabled, the server first attempts to select a previously used sequential volume with available space.

If none exists, the server selects the volume that contains the most data so that each volume is fully utilized. If no partially full volume exists, the server selects 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 38 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 user files on individual volumes.

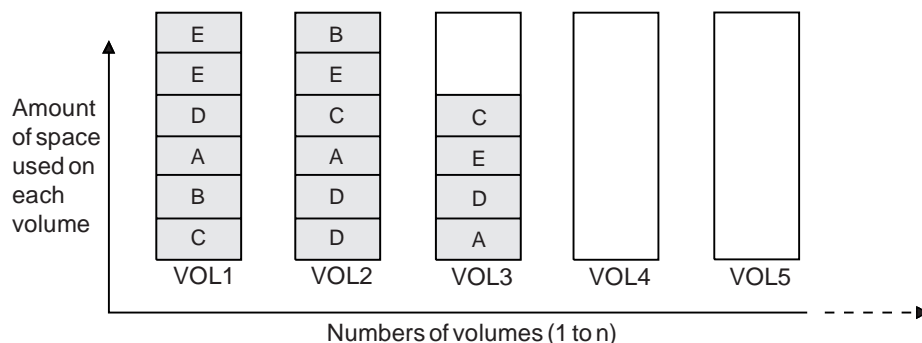


Figure 38. Using All Available Space on Sequential Volumes with Collocation Disabled

Turning Collocation On or Off

After you define a storage pool, you can turn collocation on or off 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. Direct client recovery is typically done from the primary pools while copy storage pools are usually used to recover the primary pool data. 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. 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.

Careful consideration should be given before using collocation for copy storage pools. Even customers using collocation for their primary storage pools 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 144 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.

During space reclamation, the server copies active files from the candidate volume to other volumes in the storage pool. For example, Figure 39 on page 150 shows the active files from tapes 1, 2, and 3, being consolidated on tape 4.

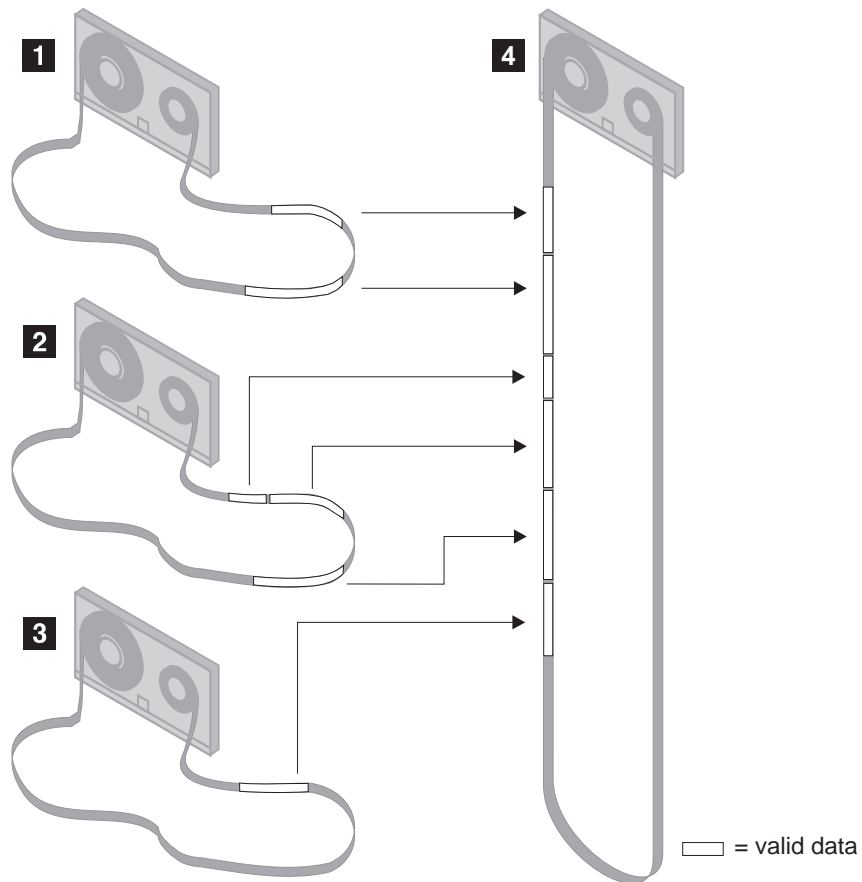


Figure 39. Tape Reclamation

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 lower the reclamation threshold, the more frequently the server tries to reclaim space occupied by obsolete files. 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 116.

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 reclaim 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 152.

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.

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 154.

How Collocation Affects Reclamation

If collocation is enabled and reclamation occurs, the server tries to reclaim each user's files 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 each client's data.
- Mount and dismount multiple volumes to allow the server to select the most appropriate volume on which to move each client data. The server tries to select a volume in the following order:
 1. A volume that already contains files belonging to the 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."

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 365. For more information about expiration, see “Expiration Processing” on page 154.

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 176 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 199 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 DISCARDATA=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, 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 prematurely for any reason, processing could 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 172.

Estimating Space Needs in Random Access Storage Pools

To estimate the amount of storage space required for each random access (disk) storage pool:

- Decide what percentage of this storage you want to keep on disk storage space:
 - 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.
- 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 141 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:

$$\text{Backup space} = \text{AvgWkstSize} * \text{Utilization} * \text{VersionExpansion} * \text{NumWkst}$$

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 compression is used, the amount of space required will be less than the total.

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 than 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 149 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 144 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 26. Some information applies only to random access storage pools or only to sequential access storage pools.

Table 26 (Page 1 of 2). 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
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

Table 26 (Page 2 of 2). Information for Defining a Storage Pool

Information	Explanation	Applies to Random Access	Applies to Sequential Access
Name of the next storage pool	Specifies the name of the next storage pool where files can be moved.	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	<i>Collocation</i> is a process in which the server attempts to keep all files belonging to a client file space on a minimal number of sequential access storage volumes.	—	Yes
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.
 - 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 acquire scratch volumes.

Collocation

Collocation is a process in which the server attempts to keep all files belonging to a client node 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 cartridges so the maximum number of scratch volumes is set to an appropriate value.

```
define stgpool disaster-recovery cartridge 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 ease of storage volume management.

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 run as long as required to back up the primary storage pool or be cancelled if desired. Reissuing the BACKUP STGPOOL command allows the backup to continue from the spot the backup was cancelled. For example, to back up the ARCHIVEPOOL primary pool to the RECOVERYPOOL copy pool, enter:

```
backup stgpool archivepool recoverypool
```

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 multiple, 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 12, “Automating Operations” on page 239 for information about scheduling commands.

Notes:

1. Backing up storage pools places additional space requirements on the 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.
4. Cached files are not backed up.
5. When setting the MAXPROCESS parameter, consider the number of mount points and drives that can be dedicated to this operation.

See “Backing Up Storage Pools” on page 345 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 cartridge tape storage pool (CART-POOL, with device class CARTRIDGE). The files stored in the random access storage pool are migrated to the cartridge 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:

1. Create a copy storage pool called CART-BACKUP, with the same device class as the CART-POOL primary storage pool, by issuing the following command:

```
define stgpool cart-backup cartridge 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 CART-BACKUP 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 CART-BACKUP by issuing the following commands:

```
backup stgpool disk-pool cart-backup  
backup stgpool cart-pool cart-backup
```

3. Define schedules to automatically run the commands for backing up the primary storage pools to the copy storage pool. These are the commands 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 12, "Automating Operations" on page 239.

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 40 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 Pool Name	Device Class Name	Estimated Capacity (MB)	%Util	%Migr	High Mig%	Low Mig%	Next Storage 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 40. 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 slightly 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 40 on page 166 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 either the ENGBACK1 or ENGBACK2 storage pools, since both storage pools show 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 left blank for copy storage pools.

High Migr%

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 left blank for copy storage pools).

Low Migr%

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 left blank for copy storage pools).

Next Storage Pool

Specifies the primary storage pool destination to which data is migrated (this field is left 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 40 on page 166 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 41 on page 169 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 Pool Name	Device Class Name	Estimated Capacity (MB)	%Util	%Migr	High Mig%	Low Mig%	Next Storage Pool
BACKTAPE	TAPE	180.0	95.2	100.0	90	70	
BACKUPPOOL	DISK	80.0	51.6	28.8	50	30	BACKTAPE

Figure 41. Information on Backup Storage Pools

You can query the server to monitor the migration process by entering:

```
query process
```

A message similar to Figure 42 is displayed:

Process Number	Process Description	Status
2	Migration	Disk Storage Pool BACKUPPOOL, Moved Files: 1086, Moved Bytes: 25555579, Unreadable Files: 0, Unreadable Bytes: 0

Figure 42. 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” on page 170 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 170 for additional information.
- Provide additional space. See “Providing Additional Space for the Migration Process” on page 171 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 43 is displayed:

Process Number	Process Description	Status
1	Migration	ANR1113W Migration suspended for storage pool BACKUPPOOL - insufficient space in subordinate storage pool.

Figure 43. 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 168, 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 168, 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 168, 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 44 on page 172 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 44. 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 313.

Figure 45 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	Filespace Name	Storage Pool Name	Number of Files	Space Occupied (MB)
SSTEINER	/home	ENGBACK1	513	3.52

Figure 45. 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 46 displays a report on the occupancy of tape storage pools assigned to the TAPE8MM device class.

Node Name	Filespace Name	Storage Pool Name	Number of Files	Space Occupied (MB)
HTANG	OS2C	ARCTAPE	5	.92
HTANG	OS2C	BACKTAPE	21	1.02
PEASE	/home/pease/dir	ARCTAPE	492	18.40
PEASE	/home/pease/dir	BACKTAPE	33	7.60
PEASE	/home/pease/dir1	BACKTAPE	2	.80
TOMC	/home/tomc/driver5	ARCTAPE	573	20.85
TOMC	/home	BACKTAPE	13	2.02

Figure 46. A Report on the Occupancy of Storage Pools by Device Class

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 156 and “Estimating Space for Archived Files in a Random Access Storage Pool” on page 158 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=backuppools type=backup
```

Figure 47 on page 175 displays a report on the amount of server storage used for backed up files.

Node Name	Filespace Name	Storage Pool Name	Number of Files	Space Occupied (MB)
HTANG	OS2C	BACKUPPOOL	513	23.52
HTANG	OS2D	BACKUPPOOL	573	20.85
PEASE	/marketing	BACKUPPOOL	132	12.90
PEASE	/business	BACKUPPOOL	365	13.68
TOMC	/	BACKUPPOOL	177	21.27

Figure 47. 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 47:

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 198 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 166.

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 159 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 155.

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 363 and “Backup and Recovery Scenarios” on page 365 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. 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.

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 (DISCARDATA=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 10. Managing Storage Pool Volumes

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

Section	Page
Concepts:	
Storage pool volumes	179
Access modes for storage pool volumes	180
Tasks:	
Preparing volumes for random access storage pools	181
Preparing volumes for sequential access storage pools	181
Defining storage pool volumes	183
Updating storage pool volumes	183
Monitoring the use of storage pool volumes	185
Auditing a storage pool volume	189
Moving files from one volume to another volume	193
Deleting storage pool volumes	197
Restoring storage pool volumes	199

Most tasks presented in this chapter can be performed by using either the graphical user interface or the command-line interface. Table 10 on page 36 shows whether a task can be performed on the graphical user interface, the command line-interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

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. A volume in a random access storage pool is always a fixed-size file that must be created using the create and format volume option (CRTVOLADSM) from the ADSM menu.

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:

8MM A volume is a single 8mm tape cartridge.

QIC A volume is a single 1/4" inch tape cartridge.

REEL	A volume is a half-inch tape reel.
CARTRIDGE	A volume is a single 3480 or 3490 cartridge system tape.
3590	A volume is a single 3590 cartridge tape (for IBM 3590 tape drives).
FILE	A volume is a file in the file system of the server machine

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	<p>Allows files to be read from or written to a volume in the storage pool.</p> <p>If the server cannot write to a read/write access volume, the server automatically changes the access mode to read-only.</p>
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	<p>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.</p> <p>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.</p> <p>Only volumes in primary storage pools can be updated to destroyed.</p> <p>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.</p> <p>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.</p>
Offsite	<p>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:</p>

- 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. Name and format the file using the create and format volume option (CRTVOLADSM command) from the ADSM Utilities menu. This is the file that the server will access for serve storage operations for the volume.
2. Define the disk storage volume using the DEFINE VOLUME command from the ADSM administrative session. 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 183.

Preparing Volumes for Sequential Access Storage Pools

Volumes are automatically created for sequential access storage pools with device type of FILE. You do not need to prepare volumes for these storage pools. The volumes are created in the OS/400 library specified when the device class was defined.

For sequential access storage pools with other than FILE device type, you must prepare volumes for use. The steps are:

1. Label the volume using the OS/400 command INZTAP.
2. 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 183. You can skip this step if you have enabled scratch volume allocation by specifying a nonzero MAXSCRATCH parameter for the storage pool.
3. For storage pools in AS400MLB libraries, use the CHECKIN LIBVOLUME command to check the volumes into the library.

Labeling Sequential Storage Pool Volumes

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

- 8MM
- QIC
- CARTRIDGE
- REEL
- 3590

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.

Use the INZTAP command to label volumes. The INZTAP command can be accessed through the ADSM menu by first selecting Utilities (option 1), then selecting Initialize a tape for ADSM (option 6).

```

ADSM                ADSTAR Distributed Storage Manager

Select one of the following:

    1. Utilities
    2. Recover ADSM
    3. Display ADSM messages
    4. Verify server status
    5. Start administrative client for ADSM

    10. Start server
    11. End server

Selection or command
====>

    F3=Exit  F4=Prompt  F9=Retrieve  F12=Cancel  F13=Information Assist
  
```

The following example shows the screen that appears after selecting Initialize a tape for ADSM.

```

                          Initialize Tape (INZTAP)

Type choices, press Enter.

Tape device . . . . .
New volume identifier . . . . . *NONE          Name
New owner identifier . . . . . > QADSM         Character value, *NONE
Volume identifier . . . . . *MOUNTED          Character value, *BLANK
Check for active files . . . . . *YES         Character value, *MOUNTED
Tape density . . . . . > *DEVTYPE            *YES, *NO, *FIRST
Code . . . . . *EBCDIC                       *DEVTYPE, *QIC120, *QIC525...
End of tape option . . . . . *REWIND         *EBCDIC, *ASCII
Clear . . . . . > *NO                        *REWIND, *UNLOAD
                                          *NO, *YES
  
```

Overwriting Existing Volume Labels

You can overwrite existing volume labels for tapes that no longer contain active data by using the INZTAP command and specifying *NO on the "Check for active files" option.

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

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. The volumes are automatically defined as they are acquired. The volumes are also automatically deleted from the storage pool when the server no longer needs them.

Before a scratch volume can be used, it must have a standard label. See "Labeling Sequential Storage Pool Volumes" on page 181.

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 might be if you accidentally damage VOL1, you can change the access mode to unavailable so that no data can be written to or read 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
```

Figure 48 shows the output of this standard query. The example illustrates that data is being stored on the 8mm tape volume named ADSTM01, as well as on several other volumes in various storage pools.

Volume Name	Storage Pool Name	Device Class Name	Estimated Capacity (MB)	%Util	Volume Status
QUSRADSM/DISKPOOL (VOL1)	AIXPOOL1	DISK	240.0	26.3	On-Line
QUSRADSM/DISKPOOL (VOL2)	AIXPOOL2	DISK	240.0	36.9	On-Line
QUSRADSM/DISKPOOL (VOL3)	DOSPOOL1	DISK	240.0	72.2	On-Line
QUSRADSM/DISKPOOL (VOL4)	DOSPOOL2	DISK	240.0	74.1	On-Line
QUSRADSM/DISKPOOL (VOL5)	OS2POOL1	DISK	240.0	55.7	On-Line
QUSRADSM/DISKPOOL (VOL6)	OS2POOL2	DISK	240.0	51.0	On-Line
ADSTM00	TAPEPOOL	TAPE8MM	2,472.0	0.0	Filling
ADSTM01	TAPEPOOL	TAPE8MM	2,472.0	2.2	Filling

Figure 48. Standard Information about Storage Pool Volumes

Requesting Detailed Information about Storage Pool Volumes

To query the server for a detailed report on volume ADSTM01 in the storage pool named TAPEPOOL, enter:

```
query volume adsm01 format=detailed
```

Figure 49 on page 186 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
```

Figure 49. Detailed Information about a Storage Pool Volume

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.

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 189 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 189 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

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 50 displays a standard report which shows the first seven files from file space /usr on TOMC stored in ADSM01.

Node Name	Type	Filespace Name	Client's Name for File
TOMC	Bkup	/usr	/bin/ acctcom
TOMC	Bkup	/usr	/bin/ acledit
TOMC	Bkup	/usr	/bin/ ac1put
TOMC	Bkup	/usr	/bin/ admin
TOMC	Bkup	/usr	/bin/ ar
TOMC	Bkup	/usr	/bin/ arcv
TOMC	Bkup	/usr	/bin/ banner

Figure 50. 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 51 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/ut1/ 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/ut1/ 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/ut1/ appl.scr
Stored Size: 1,013
Segment Number: 1/1
Cached Copy?: No
```

Figure 51. 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 any administrative client running in console mode, and to the console message queue (`CSLMSGQ` set in the server options)
- 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. 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 273.

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.

Auditing a Volume in a Disk Storage Pool

To audit a disk volume named QUSRADSM/DISKPOOL(VOL1) and have only summary messages sent, for example, enter:

```
audit volume qusradsm/diskpool(vol1) quiet=yes
```

The audit volume process is run in the background and the server returns an informational message as follows:

```
ANR2313I Audit Volume NOFIX process started for volume
QUSRADSM/DISKPOOL(Vol1) (process id 4).
```

Messages are sent to the activity log, to any administrative client running in console mode, and to the console message queue (CSLMSGQ in the server options).

To view the status of the audit volume process, enter:

```
query process
```

Figure 52 displays an example of the audit volume process report.

Process Number	Process Description	Status
4	Audit (Inspect only) Volume	Storage Pool BACKUPPOOL, Volume QUSRADSM/DISKPOOL(Vol1) , Files Processed: 680, Irretrievable Files Found: 0, Partial Files Skipped: 0

Figure 52. Information on the Audit Volume Process

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 53 shows five volumes defined to ENGBACK2. In this example, File A spans VOL1 and VOL2, and File D spans VOL2, VOL3, VOL4, and VOL5.

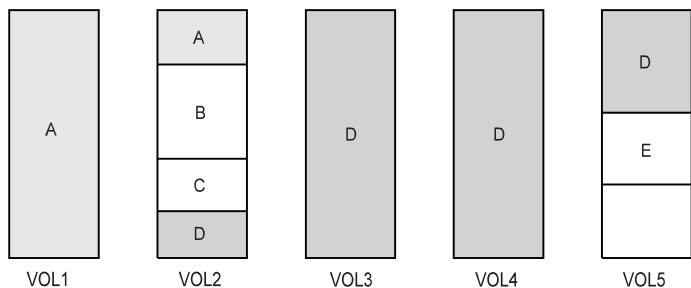


Figure 53. Tape Volumes with Files A, B, C, D, and E

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 may entail 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 53 on page 192 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 ADISM 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 197 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 its 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 183 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 185 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 166 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 183 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 123 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 volume QUSRADSM/DATA(VOL1) to any available volume in the STGTMP1 storage pool, enter:

```
move data qusradm/data(vol1) 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 QUSRADSM/DATA(VOL1)
(process ID 32).
```

Requesting Information about the Data Movement Process

To request information on the data movement process, enter:

```
query process
```

Figure 54 shows an example of the report that you receive about the data movement process.

Process Number	Process Description	Status
32	Move Data	Volume QUSRADSM/DATA(VOL1), (storage pool BACKUPPOOL2), Target Pool STGTMP1, Moved Files: 20, Moved Bytes: 1,302,528, Unreadable Files: 0, Unreadable Bytes: 0. Current File (bytes): 299,008

Figure 54. 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 qusradsm/data(vol1) stgpool=backuppool2
```

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:
 - Policy-based file expiration
 - File space deletion
 - Deletion of the primary storage pool volume

Note: If you are deleting many volumes, it is recommended that you 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 ADSTM03, 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 AD5M03
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 AD5M03 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 cancelled 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 193 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. (AD5M 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 369 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. 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

Chapter 11. 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.

Section	Page
Concepts:	
Policy operations	204
Policy objects	205
Management classes	207
Expiration processing	211
File eligibility for policy operations	212
How client migration works with backup and archive	215
Tasks:	
Using the standard storage management policies	216
Creating your own storage management policies	217
Defining a policy domain	220
Defining a policy set	221
Defining a management class	222
Defining a backup copy group	223
Defining an archive copy group	227
Assigning a default management class	229
Validating and activating policy sets	229
Starting expiration processing	231
Querying policy objects	231
Deleting policy objects	234

Most tasks presented in this chapter can be performed by using either the graphical user interface or the command-line interface. Table 11 on page 37 shows whether a task can be performed on the graphical user interface, the command-line interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

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 55 on page 206 shows the objects and their relationships.

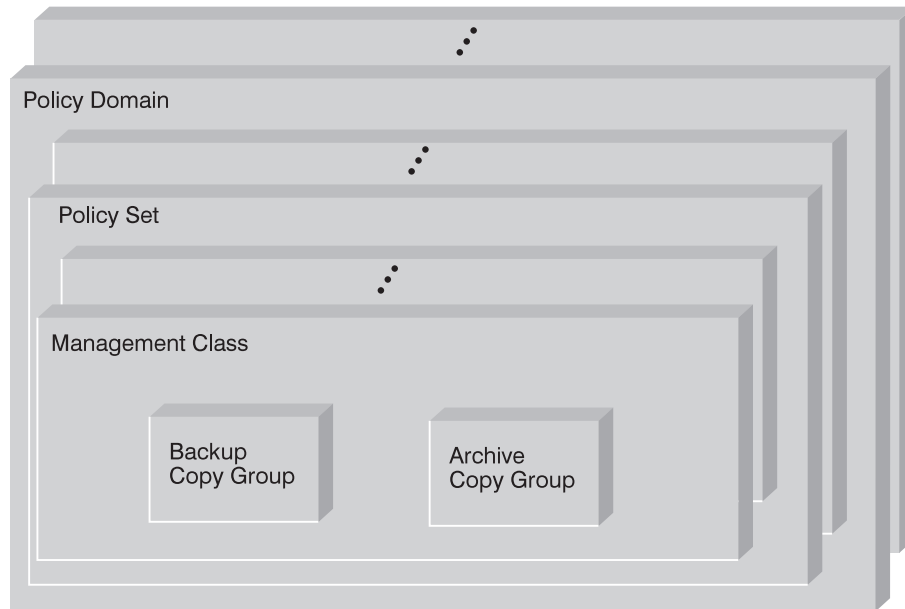


Figure 55. ADSM Policy Objects

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 309 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 209.
- 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 56 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 56 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 56 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 56. 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. Expiration processing occurs (see "Expiration Processing" on page 211 for details). REPORT.TXT is still bound to the default management class, which now includes new retention criteria. Therefore, the oldest inactive version is expired, 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 231 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 (<i>modified</i>) or whenever a backup is requested (<i>absolute</i>). |
| 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.

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: <ul style="list-style-type: none"> • 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: <ul style="list-style-type: none"> • 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 220.
2. Define policy sets for different storage management policies. See page 221.

3. Define management classes to match users' storage management requirements. See page 222.
4. Define backup copy groups to specify which files can be backed up and how to manage backup versions. See page 223.
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 227.
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 229.
7. Validate all policy sets, and activate one policy set for each policy domain. See page 230.
8. Start expiration processing. See page 231.

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 12, "Automating Operations" on page 239.

Example: Sample Policy Objects

Figure 57 on page 219 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.

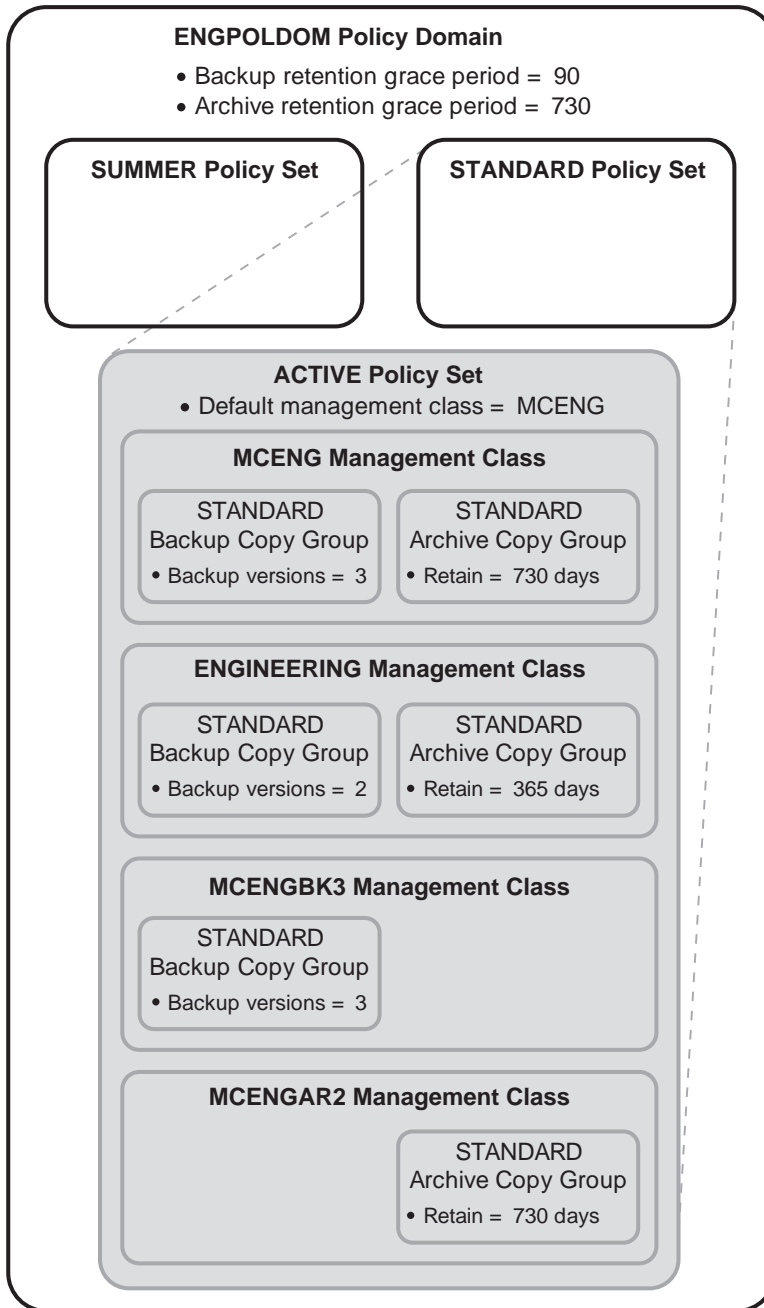


Figure 57. An Example of Policy Objects Defined for an Engineering Department

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 with 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 or 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 58 on page 226, 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.

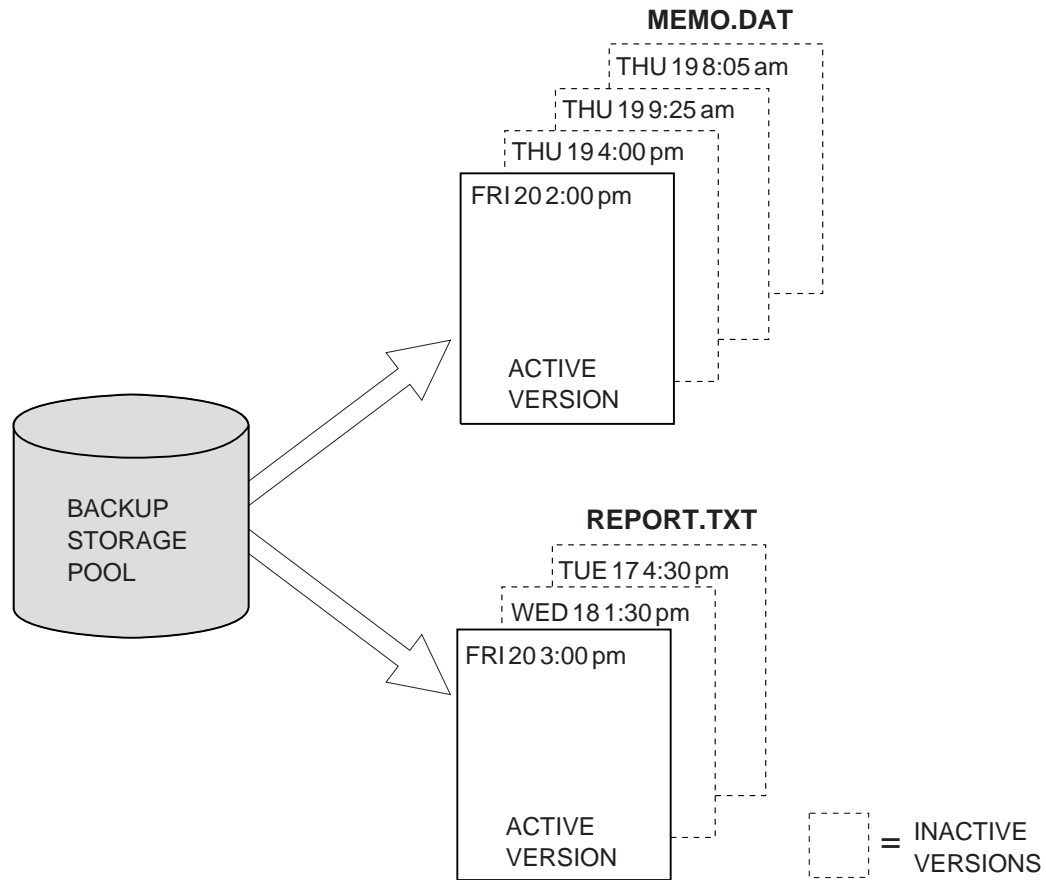


Figure 58. Example of Active and Inactive Versions of Backed Up Files

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 the retention time, in days, for all but the most recent backup version. 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 208 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 221). 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 209 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 220 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 220.

- 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 invoke expiration processing either automatically or by command. You control automatic expiration processing by using the expiration interval specified in the server options. You can set options through the ADSM Utilities menu or by issuing the CHGSVRADSM command (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).

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 figure is 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 Domain Name	Policy Set Name	Mgmt Class Name	Copy Group Name	Retain Version
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

To request information about archive copy groups in the ENGPOLDOM engineering policy domain, enter:

```
query copygroup engpoldom * type=archive
```

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 Domain Name	Policy Set Name	Mgmt Class Name	Default Mgmt Class ?	Description
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 Domain Name	Policy Set Name	Default Mgmt Class Name	Description
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 Domain Name	Activated Policy Set	Activated Default Mgmt Class	Number of Registered Nodes	Description
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

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
```

Note: 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
```

Note: 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 209.

- Delete each node from the STANDARD policy domain.

3. Delete the policy domain by entering:

```
delete domain standard
```

Note: 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

Chapter 12. 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. 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
- Macros 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	240
Automating client operations	241
Coordinating client schedules	244
Tailoring schedules	250
Copying schedules	254
Deleting schedules	255
Managing client node associations	258
Managing scheduled events	255

Most tasks presented in this chapter can be performed using either the graphical user interface or the command-line interface. Table 12 on page 39 shows whether a task can be performed on the graphical user interface, the command-line interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

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. However, 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 250.

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 250.

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_ARCHIVEPOOL	11/15/1995 14:08:11	1 H	1 D	Any

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 255.

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 250.

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. After client nodes have been associated with a schedule, the client must 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 250.

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, see 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 255.

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
<ul style="list-style-type: none"> • 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, see 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 223. 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 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, see *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 querieschedperiod 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 255.

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 starttime=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
starttime=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.

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 service 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 or a macro, you must specify the entire command or the macro file name.

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
- Issue a macro

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 for a restore, retrieve, command, or macro operation. A Macintosh client cannot run a schedule for a restore, retrieve, or macro operation.

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 of incremental backups for the ENGPOLDOM policy domain, enter:

```
define schedule engpoldom engweekly action=incremental  
period=1 perunits=weeks
```

This command sets the incremental backup period for schedule ENGWEEKLY to one week to match 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 242 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 242.

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 exceptiononly=yes
```

Figure 59 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 59. Exception Report of Events

Figure 60 shows an example of a general report for client node GOODELL that is displayed after you enter:

```
query event standard weekly_backup node=goode11
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 60. General Report of Events

To query an event for an administrative command schedule, you must specify the TYPE=ADMINISTRATIVE parameter. Figure 61 shows an example of the results of the following command:

```
query event * type=administrative
```

Scheduled Start	Actual Start	Schedule Name	Status
11/17/1995 14:08:11	11/17/1995 14:08:14	BACKUP_ARCHI- VEPOOL	Completed

Figure 61. 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, 1995, enter:

```
delete event 06/30/1995 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 from 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

Chapter 13. 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	263
Managing client sessions	265
Disabling or enabling server access	267
Managing server processes	268
Varying disk volumes online or offline	270
Requesting information about server status	270
Setting the server name	271
Querying server options	271
Managing the activity log	272
Monitoring accounting records	274
Getting help on commands and error messages	275

Most tasks presented in this chapter can be performed by using either the graphical user interface or the command line interface. Table 13 on page 41 shows whether a task can be performed on the graphical user interface, the command line interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

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 manually or have it started automatically when you start your AS/400 system. To start the server manually, complete the following steps:

1. Before the server starts, the communication methods specified in the server options should already be active. If they are not active, start them.

2. Go to the ADSM main menu by typing at any AS/400 command line:

```
===> go adsm
```

3. Select option 10 (Start server) from the ADSM main menu or use the command STRSVRADSM. The screen in Figure 62 is displayed:

```
                Start Server for ADSM (STRSVRADSM)
Type choices, press Enter.
Work library . . . . . QUSRADSM      Name, *CURLIB
                -----
```

Figure 62. Start Server Screen

4. When the server starts, it enables the communication methods specified in the server options and uses the volumes specified for the database and recovery log.
The server takes a few minutes to start. When the startup is complete, this message is sent to the message queue of the user who initiated the server (issued the STRSVRADSM command):
Server is ready to communicate with clients
5. The server is now ready to communicate with clients.

Note: Select option 4 (Verify server status) from the ADSM main menu to determine server status or use the command VFYSVRADSM.

To start the server automatically when you start your AS/400 system, add the STRSVRADSM command to your system startup program, defined in the QSTRUPPGM system value.

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. 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 267.

2. Query for session information to identify any existing administrative and client node sessions, as described in “Requesting Information about Client Sessions” on page 265.
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 267.
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. If you started the server or have all object authority, you can use the ENDSVRADSM command from an AS/400 command line instead.

Restarting the Server

To start the server after it has been halted, follow the instructions in “Starting the Server” on page 263.

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 307. For information on locking or unlocking client nodes from the server, see “Locking and Unlocking Client Nodes” on page 311.

Requesting Information about Client Sessions

When administrators or users log on to the server, 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 63 shows a sample client session report.

Sess Number	Comm. Method	Sess State	Wait Time	Bytes Sent	Bytes Recvd	Sess Type	Platform	Client Name
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 63. 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 cancelled 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 cancelled, those persons must log on to the server again. If they were in the process of performing a function when the session was cancelled, they must reissue their last command.

If the session you cancel is currently waiting for a media mount, the mount request is automatically cancelled.

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 initiates a process, such as 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 licenses
- Auditing a volume
- Backing up the database
- Backing up a storage pool
- Defining a database copy
- Defining a recovery log copy
- Deleting a file space
- Deleting a database volume
- 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 64 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 Number	Process Description	Status
2	DELETE FILESPACE	Deleting filespace DRIVE_D for node CLIENT1: 172 files deleted.

Figure 64. 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 cancelled.

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 DASD, you can vary a disk volume offline. For example, to vary the disk volume named QUSRADSM/POOL001 offline, enter:

```
vary offline qusradm/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 QADSM/POOL001 available to the server, enter:

```
vary online qusradm/pool001
```

It is only necessary to specify the library if the volume resides some place other than the work library.

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 through the ADSM Utilities menu or by issuing the CHGSVRADSM command (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 console message queue during server operation. 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 message queue 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, allowing 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 console message queue, if one is specified in the server options. 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 282.

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.

For example, to review messages generated on May 30 between 8 a.m. and 5 p.m., enter:

```
query actlog begindate=05/30/1995 enddate=05/30/1995  
begintime=08:00 endtime=05:00
```

To request information about messages related to the expiration of files from the server storage inventory, enter:

```
query actlog msgno=0813
```

See the *ADSM Messages* for message numbers.

To request information about messages generated from the IMPORT NODE command, enter:

```
query actlog search='import node'
```

Monitoring Accounting Records

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 the storage 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 an OS/400 accounting file named QAANRACTLG located in the work library for the server. The file is written out as text records that can be viewed directly. There are a total of 24 fields. Individual fields are delimited by commas (,), and records are terminated with the newline 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

Example Records:

```
0,2,adsm,9/27/94,16:33:55,dsmuser1,,os/2,1,tcp/ip,1,0,0,0,0,0,0,0,0,2,36,36,0,0  
0,2,adsm,9/27/94,16:35:15,dsmuser1,,os/2,1,tcp/ip,1,1,5,1,5,1,5,1,5,23,59,57,0,0
```

_____ 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, such as ANR0992I for example, enter:

```
help 0992
```

Chapter 14. 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	277
Tasks:	
Estimating database or recovery log space requirements	280
Adding space to the database or recovery log	282
Deleting space from the database or recovery log	286
Optimizing the performance of the database or recovery log	290

Most tasks presented in this chapter can be performed by using either the graphical user interface or the command-line interface. Table 14 on page 43 shows whether a task can be performed on the graphical user interface, the command-line interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

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.

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 17, “Protecting and Recovering Your Data” on page 341 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 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 65, the database for SERVER1 consists of four volumes: VOL1 through VOL4. ADSM tracks the database as a single logical volume.



Figure 65. A Server Database

To manage the database and recovery log effectively, you must understand the following concepts:

- Available space, page 279
- Assigned capacity, page 279
- Utilization, page 279

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 282 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 285 for details.

Utilization

Utilization is the percent of assigned capacity used at a specific point in 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 66 on page 280 shows that utilization statistics were reset at 9 p.m. the previous evening and the maximum percent utilized for the recovery log occurred at 12 a.m.

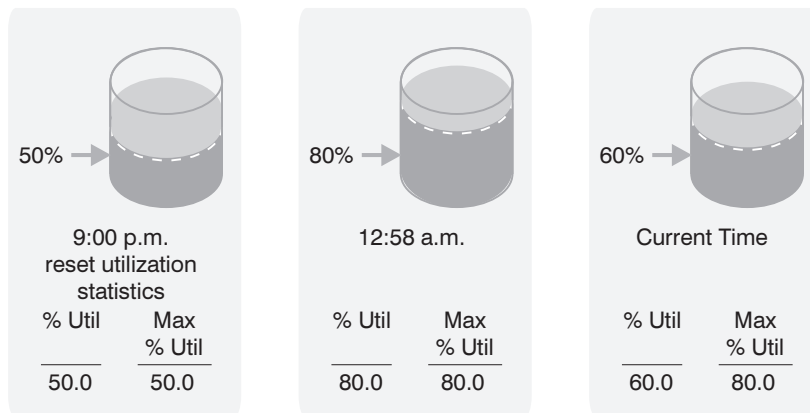


Figure 66. An Example of Recovery Log Utilization

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

As a general guideline, you should allocate to the database from 5% to 10% of the space required for server storage. For example, if you need 10GB of server storage, your database should be between 500MB and 1GB. See “Estimating Space Needs for Storage Pools” on page 156 for details.

If you back up primary storage pools to copy storage pools, the database also requires about 200 bytes of overhead space for each file in a copy storage pool.

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 343 and “Estimating the Size of the Recovery Log” on page 351 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 need to add or delete space.

Resetting the maximum utilization counters for the database and recovery log lets you 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, enter:

```
query db
```

The server displays a report, like this :

Available Space (MB)	Assigned Capacity (MB)	Maximum Extension (MB)	Maximum Reduction (MB)	Page Size (bytes)	Total Pages	Used Pages	%Util	Max. %Util
96	96	0	92	4,096	24,576	86	0.3	0.3

To display information about the recovery log, enter:

```
query log
```

The server displays a report, like this:

Available Space (MB)	Assigned Capacity (MB)	Maximum Extension (MB)	Maximum Reduction (MB)	Page Size (bytes)	Total Pages	Used Pages	%Util	Max. %Util
12	12	0	8	4,096	3,072	68	2.2	2.2

See the indicated pages for details about the following entries:

- Available space, page 279
- Assigned capacity, page 279
- Utilization and maximum utilization, page 279

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” on page 282 or “Deleting Space from the Database or Recovery Log” on page 286.

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.

Attention: You must not change the size of allocated database or recovery log volumes. If you change the size of a volume, ADSM may not initialize correctly, and data may be lost. 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.

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 283
- “Step 3: Extending the Capacity of the Database or Recovery Log” on page 285

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 98MB.

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 67 on page 283.


Allocated Space on Physical Volumes			Available Space for the Database	
25 MB			24 MB	
25 MB			24 MB	
25 MB			24 MB	
25 MB			24 MB	
Totals	100 MB		96 MB	

Figure 67. An Example of Available Space

Notes:

1. To protect database and recovery log volumes from media failure, you can use the mirroring feature. You should also place the mirrored volumes in separate auxiliary storage pools (ASP). See “Mirroring the Database and Recovery Log” on page 347 for information on the mirroring feature.
2. 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 ADSTM 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 286.

For example, to allocate an additional 101MB to the database, define volume VOL5 by issuing CRTVOLADSM from an AS/400 session:

```
===> crtvoladsm vol((qusradsm/vo15 *file 101))
```

Step 2: Defining Database or Recovery Log Volumes to ADSTM

To define a database volume named VOL5, enter:

```
define dbvolume vo15
```

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 ADSTM cannot use the space until the capacity

is extended (see “Step 3: Extending the Capacity of the Database or Recovery Log” on page 285).

After you define your volumes, you can verify the change by querying the database or recovery log. To query the database, enter:

```
query db
```

The server displays a report, like this:

Available Space (MB)	Assigned Capacity (MB)	Maximum Extension (MB)	Maximum Reduction (MB)	Page Size (bytes)	Total Pages	Used Pages	%Util	Max. %Util
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 68 illustrates these changes.

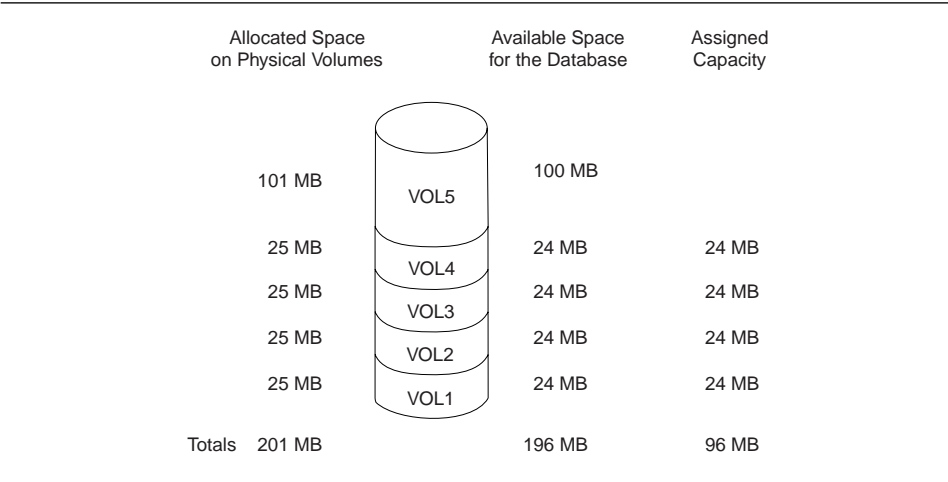


Figure 68. Adding Volumes Increases Available Space

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 69.

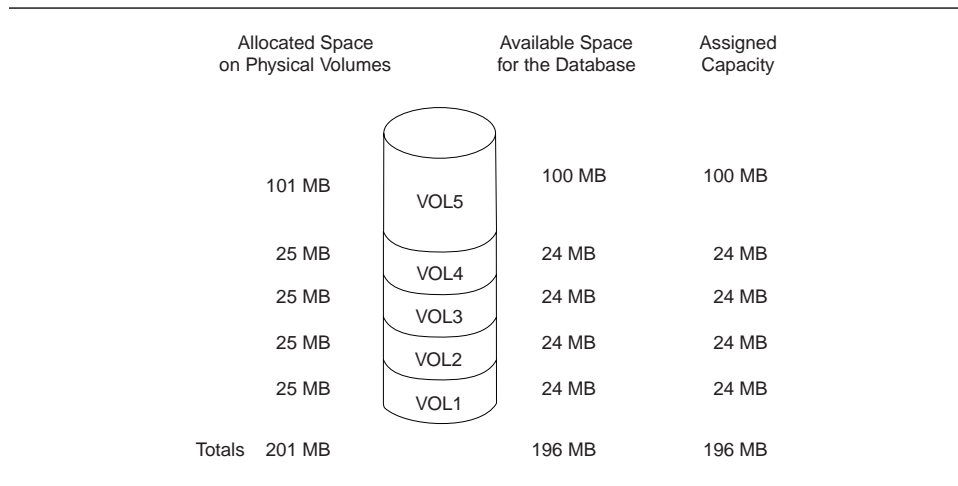


Figure 69. Extending the Capacity of the Database

You can issue a QUERY DB or QUERY LOG command to verify the assigned capacity of the database or recovery log. The server would display a report, like this:

Available Space (MB)	Assigned Capacity (MB)	Maximum Extension (MB)	Maximum Reduction (MB)	Page Size (bytes)	Total Pages	Used Pages	%Util	Max. %Util
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 282 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 286).
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 288
3. Delete the volume (page 288).

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): QUSRADM/VOL04(VOL04)
      Copy Status: Sync'd
Volume Name (Copy 2): QUSRADM/VOLD(VOLD)
      Copy Status: Sync'd
Volume Name (Copy 3): QUSRADM/VOL300(VOL300)
      Copy Status: Sync'd
Available Space (MB): 24
Allocated Space (MB): 24
      Free Space (MB): 0

Volume Name (Copy 1): VOL5
      Copy Status: Sync'd
Volume Name (Copy 2): VOLE
      Copy Status: Sync'd
Volume Name (Copy 3): VOL200
      Copy Status: Sync'd
Available Space (MB): 100
Allocated Space (MB): 100
      Free Space (MB): 0

more ...

```

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 Space (MB)	Assigned Capacity (MB)	Maximum Extension (MB)	Maximum Reduction (MB)	Page Size (bytes)	Total Pages	Used Pages	%Util	Max. %Util
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 282 and “Step 2: Defining Database or Recovery Log Volumes

to ADSM” on page 283 and continue with “Step 2: Reducing the Capacity of the Database or Recovery Log” on page 288.

Step 2: Reducing the Capacity of the Database or Recovery Log

The *maximum reduction* identifies the number of megabytes by which 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 the utilization of the database, VOL5 alone could contain all the data. To reduce the database by the amount of available space in VOL1-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 Space (MB)	Assigned Capacity (MB)	Maximum Extension (MB)	Maximum Reduction (MB)	Page Size (bytes)	Total Pages	Used Pages	%Util	Max. %Util
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 volumes 1 through 4 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 70 shows data moved from VOL1, VOL2, VOL3, and VOL4 to available space on VOL5.

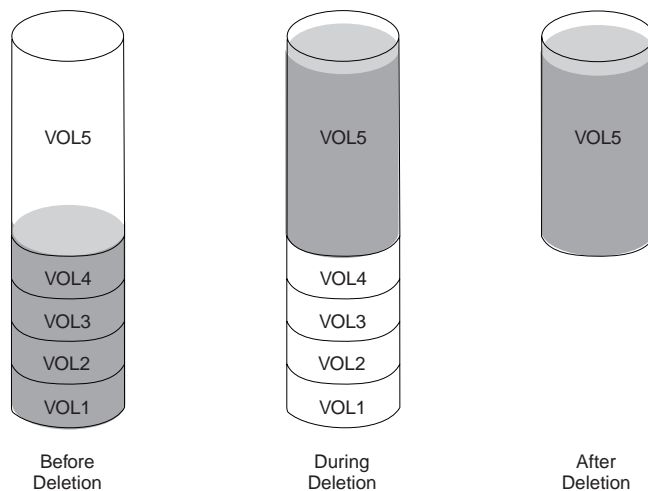


Figure 70. Deleting Database Volumes

After the data has been moved, these volumes are deleted from the server.

Now you can delete the physical file members and free up space on the AS/400 by entering:

```
====> rmvm file(qusradsm/vo11) mbr(vo11)
====> rmvm file(qusradsm/vo12) mbr(vo12)
====> rmvm file(qusradsm/vo13) mbr(vo13)
====> rmvm file(qusradsm/vo14) mbr(vo14)
```

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 since the last reset of the buffer pool. 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. You can set options through the ADSM Utilities menu or by issuing the CHGSRVADSM command (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 *log pool percent utilization* (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 log pool size option. You can set options through the ADSM Utilities menu or by issuing the CHGSVRADSM command (see *ADSM Administrator's Reference*).

Chapter 15. 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	295
Ensuring client/server authentication	299
Registering administrators or updating information	300
Granting administrative authority	300
Revoking or reducing administrative authority	304
Managing administrator access	305
Managing client node registration	308
Registering an application programming interface to the server	315
Managing client node access	310
Requesting information about client nodes	311
Requesting information about file spaces	313
Deleting file spaces and client nodes from the server	314

Most tasks presented in this chapter can be performed using either the graphical user interface or the command-line interface. Table 15 on page 45 shows whether a task can be performed on the graphical user interface, the command line interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

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:

- 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 298. In either case, you must contact your IBM account representative or authorized reseller to modify your agreement.

Licensed Features

The base ADSM OS/400 server license supports an unlimited number of administrative clients, one DESKTOP backup-archive client, and a specified set of removable media devices.

Note: In this licensing section, the term *client* is used to refer to backup-archive clients, unless otherwise noted.

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 DESKTOP backup-archive client. If you want to add clients in an environment other than DESKTOP, 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 DESKTOP. The base license allows only for backup-archive clients on DESKTOP. See “Registering Clients Other Than DESKTOP Clients” on page 297.
- Add support for storage devices not covered by the existing agreement. See “Registering Device Support Modules” on page 297.
- Add a second server attachment to an already supported library. See “Registering Secondary Server Attachment” on page 298.

You can register licenses by using the REGISTER LICENSE command, which is described in this section. You can also register licenses by using the CHGSVRADSM command, which lets you change your server options, including license terms. However, if you change the server options, you must then stop and restart the server so that it can read the updated options. For details about the CHGSVRADSM, see *ADSM Administrator's Reference*.

Registering Additional Clients

You register the server to support a specified number of clients beyond the one DESKTOP 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 DESKTOP Clients” on page 297).

For example, to register three additional clients, enter:

```
register license clients 3
```

Note: 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 DESKTOP Clients

You can obtain licenses for environment support features that allow the server to support clients other than DESKTOP clients.

Environment support features are:

DESKTOP

OS/2, DOS, Macintosh, Novell NetWare, Windows, or Windows 32-bit

UNIX

Any UNIX clients such as AIX, HP-UX, or SunOS

OPENEDITION

OpenEdition for MVS

SPACEMGMT

HSM clients

The *ADSM Licensed Program Specifications* and *License Information* list the supported clients. However, client support is continually expanded. For current information about supported clients, check with IBM or your authorized reseller, or call the IBM Information Support Center at 1-800-IBM-3333 and ask for STAR 20. To register more than one environment support feature, issue a separate REGISTER LICENSE command for each feature. For example, to allow AIX clients and HSM clients, issue:

```
register license unix
register license spacemgmt
```

Registering Device Support Modules

You can obtain licenses for device support modules that allow the server to 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 OS/400 base license includes Device Support Modules 1 and 2.

To let the server attach storage devices in Device Support Module 3, enter:

```
register license devicemod3
```

Any attempt to define a library or drive that requires a device support module fails if the module is not registered. If you try to mount a volume requiring a library or drive that is not licensed, the operation also fails.

The *ADSM Licensed Program Specifications* and *License Information* list the devices and libraries supported by each device support module. However, device support is continually expanded. For current information about supported devices, check with IBM

or your authorized reseller, or call the IBM Information Support Center at 1-800-IBM-3333 and ask for STAR 20.

Registering Secondary Server Attachment

You can obtain a license for attaching a secondary server to a library. For example, if you have a license for Device Support Module 4, you can get a license that lets you attach a secondary server to a library in that module. Register that license by entering:

```
register license secondaryserverattach
```

Licensing Example

You must issue a separate REGISTER LICENSE command for each feature or device support module. For example, to license a server for one or more features or device support modules:

1. Obtain the additional licenses for the features or device support modules from your IBM account representative or authorized reseller.
2. Issue a REGISTER LICENSE command for each feature or device support module.

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 commands or the GUI to compare the current configuration with the current licenses.

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'
```

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
```

You can reset the password for the default AS/400 administrative client, `ADSMADMIN`, when you start the server. After you select option 10 (Start server) from the `ADSM` main menu or use the `STRSVRADSM` command, you can press F10 to see additional parameters. One of these parameters resets the password for the `ADSMADMIN` administrative client to the default password, `ADSMADMIN`.

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 71 on page 301. An administrator with system privilege can perform any `ADSM` function. Administrators with policy, storage, operator, or analyst privileges can perform subsets of `ADSM` functions.

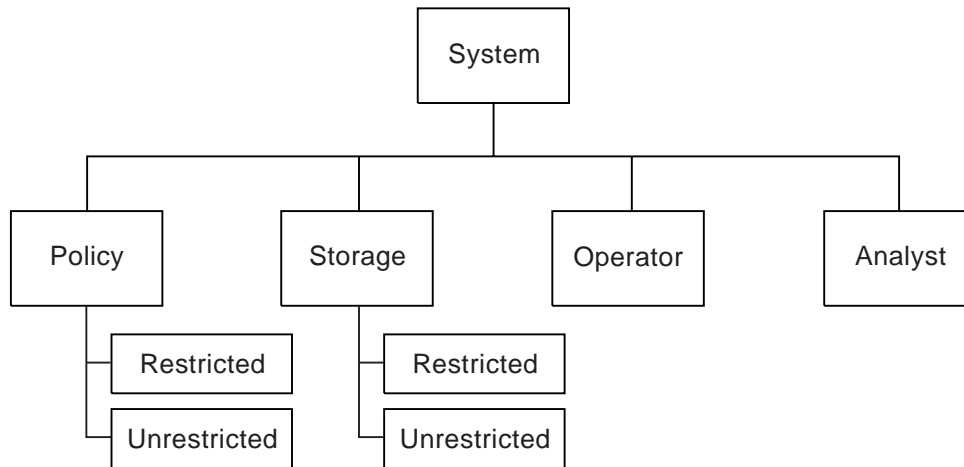


Figure 71. Administrative Privilege Classes

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, 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 perform some storage management operations only for the storage pools to which they have been authorized. They do not have authority to manage the database or recovery log.

An administrator with restricted storage privilege can:

- Define volumes to authorized disk or tape storage pools
- Move data from a volume to another volume in an authorized storage pool
- Delete volumes from an authorized storage pool
- Audit volumes belonging to an authorized storage pool

For example, an installation has the following storage pools beginning with the name "ADSM": ADSM.BFS.TAPE1, ADSM.BFS.TAPE2, and ADSM.BFS.TAPE3. To grant restricted storage privilege for those storage pools to administrator HOLLAND, enter:

```
grant authority holland stgpools=adsm*
```

HOLLAND is restricted to managing storage pools beginning with ADSM 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, ADSM.BFS.TAPEX, to HOLLAND's authority, enter:

```
grant authority holland stgpools=adsm.bfs.tapex
```

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 Remove other administrators from the server Temporarily prevent other administrators from accessing the system	System privilege
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"
```

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
```

Note: You cannot remove the last system administrator from the system.

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.

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 72 displays a detailed report.

```
Administrator Name: DAVEHIL
Last Access Date/Time: 02/09/1995 19:49:46
Days Since Last Access: 1
Password Set Date/Time: 02/08/1995 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: 02/09/1995 19:00:00
Registering Administrator: REES
```

Figure 72. 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
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, entering:

```
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 310).

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.

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. 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 set in the COMPRESSION option

The COMPRESSION option can be set in the client system options file or in 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 314).

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 11, “Managing Policies” on page 203.)

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 Name	Days Since Last Access	Days Since Password Set	Locked?
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: 02/21/1995 10:58:36
Days Since Last Access: 3
Password Set Date/Time: 02/09/1995 10:02:00
Days Since Password Set: 12
Locked?: No
Contact:
Compression: Yes
Archive Delete Allowed?: No
Backup Delete Allowed?: No
Registration Date: 02/09/1995 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 filesystem 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 Name	Platform	Filespace Type	Capacity (MB)	%Util
PEASE	/home/peas-e/dir	AIX	JFS	196.0	91.7
PEASE	/home/peas-e/dir1	AIX	JFS	328.0	81.0
PEASE	/home/peas-e/dir2	AIX	JFS	46.9	96.0

Deleting File Spaces and Client Nodes

You can delete a client node from a server, but first you must delete any 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 308 and "Administrator Registration of Client Nodes" on page 309 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 filesystem 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 filesystem 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 314), 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 16. Exporting and Importing Data

Task	Required Privilege Class
Export: Copy some or all server information to sequential volumes Import: Copy server information from sequential volumes to a server	System
Display information about export and import processes	Any administrator

ADSM provides an export-import facility that allows you to copy all or part of a server to removable media so that data can be transferred to another server.

This section takes you through the task of exporting data to sequential media and importing data to create a new ADSM server. The sections listed in the following table begin at the indicated pages.

Section	Page
Concepts:	
Data that can be exported and imported	317
Tasks:	
Preparing to export or import data	318
Monitoring export and import processes	320
Exporting data to sequential media volumes	324
Importing data from sequential media volumes	328

Most tasks presented in this chapter can be performed by using either the graphical user interface or the command-line interface. Table 16 on page 47 shows whether a task can be performed on the graphical user interface, the command-line interface, or both.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

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
 - 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

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

Upon successful completion of an import operation, 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 tape 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 console output message queue (usually ADSMMSGQ) 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 console output message queue (ADSMMSGQ) 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 320. If necessary, you can cancel an export or import process, as described in “Canceling Server Processes” on page 269.

Planning for Sequential Media Used to Export Data

To export data, you must specify a device class that supports removable 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 8, “Defining Device Classes” on page 115.

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 Tapes to Label

To estimate the number of tapes required to store export data, divide the number of bytes to be moved by the estimated capacity of a volume.

For example, cartridge system tape volumes used with 3490 tape devices have an estimated capacity of 360MB. If the preview shows that you need to transfer 720MB 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.

Labelling Tapes

During an import process, you must specify the order in which tape volumes will be mounted. This order must match the order in which tapes have been mounted during the export process.

To ensure that tapes are mounted in the correct order, label tapes 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 tape. 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 console output message queue (usually ADSMMMSGQ) or from an administrative client running in console mode.
- You can query the activity log for status information when a process has completed, by using an administrative client 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 269.

Viewing Information from an Administrative Client

When you issue an IMPORT or EXPORT command from an administrative client, a process runs in the background to execute the command. To see the results of that process as it executes, you can start an administrative client session in console mode. Figure 73 on page 322 shows an example of the information that is displayed on an administrative client console session after issuing an EXPORT SERVER command.

```

ANR0610I EXPORT SERVER started by ADSMADMIN 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 ADSMADMIN as process 1 has ended.

```

Figure 73. Sample Export Server Output Sent to the Administrative Client Console Session

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:

```
> dsmadm -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*.

Note: Console mode is not supported on an AS/400 administrative client. However, you can simulate it by specifying a console message queue in the server options. You can set options through the ADSM Utilities menu or by issuing the `CHGSVRADSM` command (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 74 on page 324 displays a sample activity log report.

Date/Time	Message
05/03/1995 10:50:28	ANR0610I EXPORT SERVER started by ADSMADMIN as process 1.
05/03/1995 10:50:28	ANR0639I EXPORT SERVER: Processing domain ENGPOLDOM.
05/03/1995 10:50:28	ANR0640I EXPORT SERVER: Processing policy set ACTIVE in policy domain ENGPOLDOM.
05/03/1995 10:50:28	ANR0640I EXPORT SERVER: Processing policy set STANDARD in policy domain ENGPOLDOM.
05/03/1995 10:50:29	ANR0641I EXPORT SERVER: Processing management class STANDARD in domain ENGPOLDOM, set ACTIVE.
05/03/1995 10:50:29	ANR0641I EXPORT SERVER: Processing management class STANDARD in domain ENGPOLDOM, set STANDARD.
05/03/1995 10:50:29	ANR0643I EXPORT SERVER: Processing archive copy group in domain ENGPOLDOM, set STANDARD, management class ACTIVE.
05/03/1995 10:50:29	ANR0643I EXPORT SERVER: Processing archive copy group in domain ENGPOLDOM, set STANDARD, management class STANDARD.
05/03/1995 10:50:29	ANR0642I EXPORT SERVER: Processing backup copy group in domain ENGPOLDOM, set STANDARD, management class ACTIVE.
05/03/1995 10:50:29	ANR0642I EXPORT SERVER: Processing backup copy group in domain ENGPOLDOM, set STANDARD, management class STANDARD.
05/03/1995 10:50:29	ANR0604I EXPORT SERVER: No schedules were found in policy domain * for exporting.
05/03/1995 10:50:29	ANR0635I EXPORT SERVER: Processing node TOMC.
05/03/1995 10:50:29	ANR0605I EXPORT SERVER: No schedule associations were found in policy domain * for exporting.
05/03/1995 10:50:29	ANR0637I EXPORT SERVER: Processing file space DRIVED for node TOMC.
05/03/1995 10:50:29	ANR0637I EXPORT SERVER: Processing file space OS2 for node TOMC.
05/03/1995 10:50:29	ANR0637I EXPORT SERVER: Processing file space OS2VDISK for node TOMC.
05/03/1995 10:50:32	ANR0617I EXPORT SERVER: Processing completed successfully.
05/03/1995 10:50:32	ANR0620I EXPORT SERVER: Copied 1 domain(s).
05/03/1995 10:50:32	ANR0621I EXPORT SERVER: Copied 2 policy set(s).
05/03/1995 10:50:32	ANR0622I EXPORT SERVER: Copied 2 management class(es).
05/03/1995 10:50:32	ANR0623I EXPORT SERVER: Copied 4 copy group(s).
05/03/1995 10:50:32	ANR0626I EXPORT SERVER: Copied 1 node definition(s).
05/03/1995 10:50:32	ANR0627I EXPORT SERVER: Copied 3 file space(s), 16 export file(s) and 0 backup file(s).
05/03/1995 10:50:32	ANR0629I EXPORT SERVER: Copied 3045632 bytes of data.
05/03/1995 10:50:32	ANR0611I EXPORT SERVER started by ADSMADMIN as process 1 has ended.

Figure 74. 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 “Labelling Tapes” on page 320 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 267. For more information on locking or unlocking administrators from the server, see “Locking and Unlocking Administrators from the Server” on page 307.

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 267. 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 267.

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 311.

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 267.
- Unlocking client nodes, as described in “Locking and Unlocking Client Nodes” on page 311

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 CARTRIDGE 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=cartridge  
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 CARTRIDGE 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=cartridge
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 CARTRIDGE device class

To issue this command, enter:

```
export node filespace=* domains=engpoldom
filedata=backupactive devclass=cartridge
```

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 CARTRIDGE device class, but allow the server to use scratch tape volumes if necessary. To issue this command, enter:

```
export policy engpoldom
devclass=cartridge 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=cartridge preview=yes
volumenames=dsm001,dsm002,dsm003,dsm004
```

Figure 75 on page 330 shows an example of the messages sent to the console output message queue and the activity log.

```

ANR0402I Session 3 started for administrator ADSMADMIN (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
05/07/1995 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 ADSMADMIN as process 2 has ended.

```

Figure 75. 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 75 shows that 88 536 358 bytes of data will be imported. Ensure that you have at least 88 536 358 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 333 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 318. For information on reviewing the results of a preview operation, see “Monitoring Export and Import Processes” on page 320.

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 332
- 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 332.

Then import the server control information from specified tape volumes. See “Importing Server Control Information” on page 333.

Determining Whether to Replace Existing Definitions

By using the `REPLACEDFS` 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 `REPLACEDFS=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:

```
> dsmadm -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 CARTRIDGE device class.

From an administrative client session, enter:

```
import server filedata=none devclass=cartridge  
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 332.
 - 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 159.
 - 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 221.

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 console output message queue (usually ADSMMSGQ) 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 CARTRIDGE device class. To issue this command, enter:

```
import server filedata=all replacedefs=no
devclass=cartridge 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 27 on page 337 shows the commands you can use to import a subset of exported information to a target server.

Table 27. 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 console output message queue (ADSMMSGQ) 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 335.

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

Chapter 17. 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.

Concepts	
Levels of data protection provided by ADSM	Page 341
Protecting Data	
How to back up storage pools	Page 345
How to mirror the database and recovery log	Page 345
How to back up the database	Page 349
Recovering Data	
How to recover the database and recovery log from mirrored copies	Page 358
How to recover the database from backups	Page 358
How to recover damaged files	Page 358
Backup and recovery scenarios	Page 365

Most tasks presented here can be performed through the graphical user interface, the command line interface, or both. Table 17 on page 49 shows where each task can be performed.

For information about issuing ADSM commands, see *ADSM Administrator's Reference*, or issue the HELP command from the command line of an ADSM administrative client. For help performing a task on the graphical user interface, refer to the online help.

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
- Your mirrored volumes must be in a separate AS/400 Auxiliary Storage Pool (ASP)

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.

Normal	Roll-forward
Level of Protection	
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: <ul style="list-style-type: none"> 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)

Normal	Roll-forward
Storage Requirements	
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: <ul style="list-style-type: none"> • 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. <p>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.</p>

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 365.

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 from a Disaster

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 copy volumes as unavailable
3. Bring the copy volumes onsite and mark them as read/write
4. Restore the destroyed files
5. Mark the copy 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 76). 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.

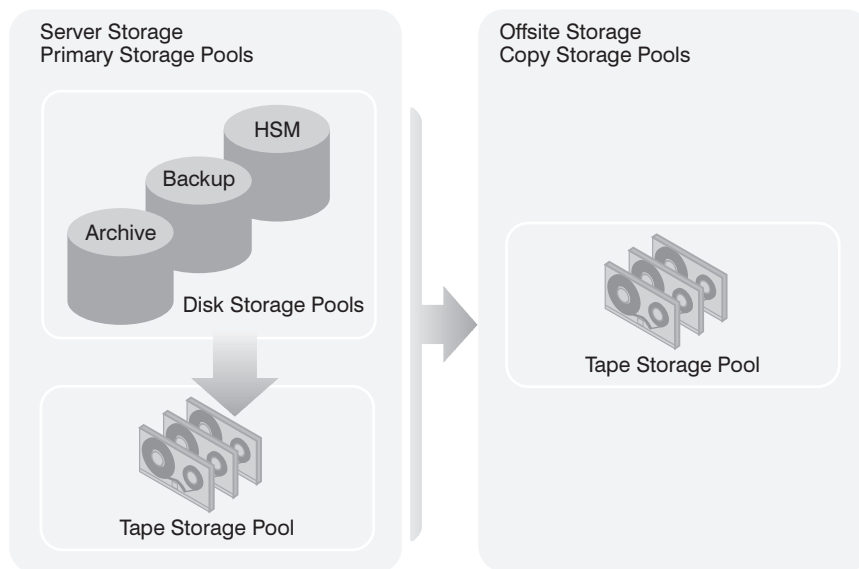


Figure 76. Copy Storage Pools

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 367 and “Recovering a Lost or Damaged Storage Pool Volume” on page 369.

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

Note: Mirrored volumes must be in a separate AS/400 ASP. to provide protection from a crash of your primary storage pool data.

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 Auxiliary Storage Pools

By keeping volume copies in auxiliary storage pools (ASP) you protect the server against disk failure and increase the availability of the database and recovery log. ADSM mirrored volumes should have at least the same capacity as the original volumes.

Defining Database or Recovery Log Mirrored Volumes

To mirror the entire database or recovery log, define a volume copy for each volume in the database or recovery log. Ensure that ADSM mirrored volumes are in AS/400 ASPs separate from the original copy.

If the database consists of five volumes named, for example, VOL1, VOL2, VOL3, VOL4, and VOL5, you must define five volume copies to mirror the database. Figure 77 on page 348 shows a mirrored database. In this example, VOL3 and VOLC are a group of mirrored volumes with the same portion of the database.

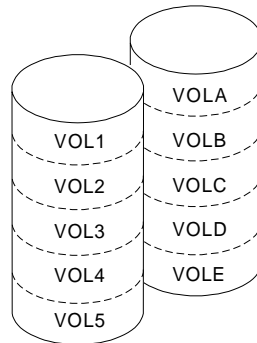


Figure 77. Mirrored Volumes

Format the space by using the CRTVOLADSM command. Then define the group of mirrored volumes by entering, for example, the following commands:

```
define dbcopy VOL1 VOLA
define dbcopy VOL2 VOLB
define dbcopy VOL3 VOLC
define dbcopy VOL4 VOLD
define dbcopy VOL5 VOLE
```

Note: You can use a macro to define volume copies from the command line.

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 1)	Copy Status	Volume Name (Copy 2)	Copy Status	Volume Name (Copy 3)	Copy Status
-----		-----		-----	
VOL1/DB(DB)	Sync'd	VOLA/DB(VOL1)	Sync'd		Undef-
VOL2/DB(DB)	Sync'd	VOLB/DB(VOL1)	Sync'd		ined
VOL3/DB(DB)	Sync'd	VOLC/DB(VOL1)	Sync'd		
VOL4/DB(DB)	Sync'd	VOLD/DB(VOL1)	Sync'd		
VOL5/DB(DB)	Sync'd	VOLE/DB(VOL1)	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 357) 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 (QOPTADSM in the server work library)
- Database and recovery log set up (the output from detailed queries of your database and recovery log volumes)

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 can avoid the situation in which a backup based on the database backup trigger is run but no mount point is available. A database backup is a high priority operation; if you share the device class with other operations and all the mount points are in use, ADSM automatically cancels lower priority operations, such as reclamation. 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 355).

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 343 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 on page 351 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 282 or “Deleting Space from the Database or Recovery Log” on page 286.

Note: If the recovery log runs out of space, you may not be able to start the server for normal operation. You can include the EXTENDLOG parameter on the STRSVRADSM command to create an additional recovery log volume if needed to start the server and perform the needed database backup.

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 351 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.

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 355 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.

The default in ADSM for AS/400 is to create a volume history file, named VOLHISTORY, containing the information in the QUSRADSM library. To ensure the availability of the information, you can do any of the following:

- Store at least one copy of the volume history information on a disk separate from the database, or offsite
- Store a copy of the volume history file offsite with your database backups
- Store a remote copy, for example, on an NFS-mounted file system
- Store a printout of the information stored offsite
- Include the volume history in the nightly AS/400 backup

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 VOLHSTFILE server option lets you specify a file in a different library and create additional backup volume history files (see the CHGSVRADSM command in 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 CHGSVRADSM command.

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 VOLHSTFILE 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
```

Saving the Device Configuration Backup File

The device configuration file contains information about the device classes, libraries, and drives needed to read backup data. For libraries of type USRDFN, the definitions for exits that interface with a media management system are also written to the device configuration file. You must create and build your exit programs before using the device configuration file for a restore operation. 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.

The default in ADSM for AS/400 is to create device configuration file, named DEVCONFIG, containing the information in the QUSRADSM library. To ensure the availability of the 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 stored offsite with your volume history file and database backups
- Store a remote copy, for example, on an NFS-mounted file system
- Store a printout of the information stored offsite

The DEVCFGFILE server option lets you specify a file in a different library and create additional backup device configuration files (see the CHGSVRADSM command in 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" 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 DEVCFGFILE 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:

- DSPVOLADSM
- STRRSTADSM

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 EXIT
 - DEFINE DRIVE

You need to provide only 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:

```
/* IBM AdStar Distributed Storage Manager Device Configuration */
define devclass 8mmtape devtype=8mm library=8mmlib
define library 8mmlib libtype=manual
define drive 8mmlib tapedrive3 device=tap03
```

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 (CRTVOLADSM)
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. 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:

- 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)

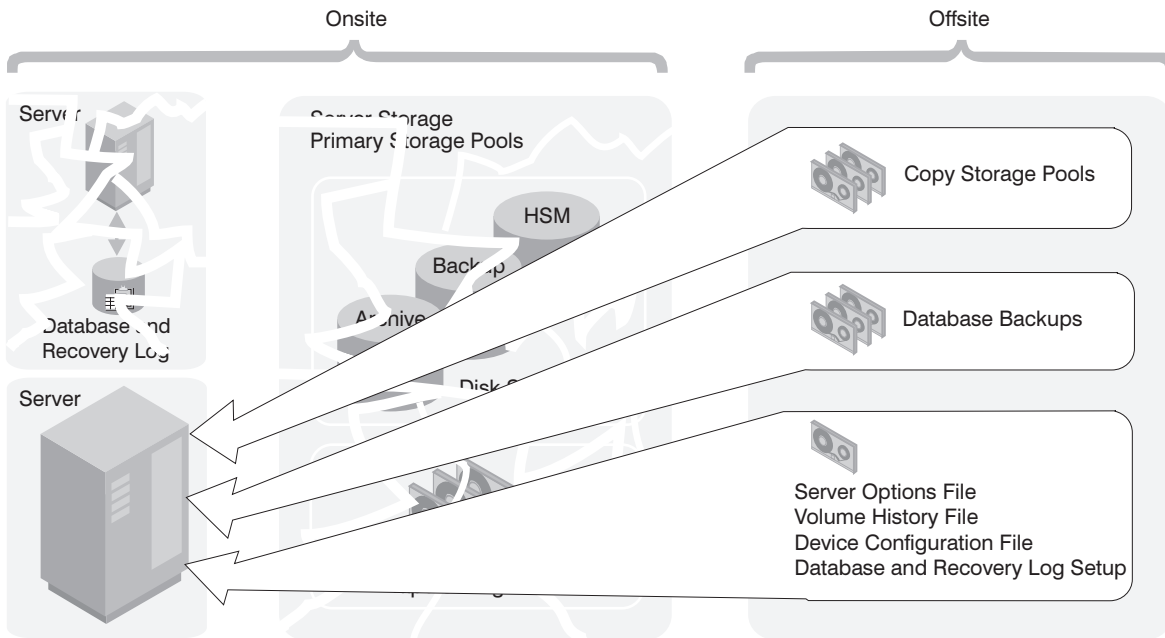


Figure 78. Recovery from a Disaster

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 356). Put the recreated file in the server work library. You can do the same with the server options file.
3. Issue the STRRSTADSM command. For example, to restore the database to a backup series that was created on April 19, 1996, enter:

```
===> strrstadsm type(*rstdate) rstdate(04/19/96)
```

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 STRRSTADSM 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, you need a list of all the volumes that were reused, added, and deleted since the original backup. Use this list to perform the following steps.

4. Audit all disk volumes, all reused volumes (STGREUSE), and any deleted volumes that you could locate with FIX=YES.

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.

5. 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 to delete database entries for files not found in the copy storage pool.
6. 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 DISCARDATA=YES.
7. 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, 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 STRRSTADSM command in the sequence in which they were written to. First, however, issue the DSPVOLADSM command to read your backup volumes and display the information needed to arrange them in order (backup series, backup operation, and volume sequence):

```
===> dspvoladsm type(*dbbackup) devclass(tape8mm)
      vol(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:

```
===> strrstadsm type(*vollist) vollist(dsm023 dsm037 dsm087)
      devclass(tape8mm) commit(*no)

===> strrstadsm type(*vollist) vollist(dsm012)
      devclass(tape8mm) commit(*no)

===> strrstadsm type(*vollist) vollist(dsm038 dsm058)
      devclass(tape8mm) commit(*yes)
```

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 STRRSTADSM 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:

```
===> strrstadsm type(*rcylog)
```

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” on page 364.

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 165.

If a client tries to access a file stored in CART-POOL and a data integrity error occurs, the file in CART-POOL is automatically marked as damaged. Future accesses to the file automatically use the copy in CART-BACKUP as long as the copy in CART-POOL is marked as damaged.

To restore any *damaged* files in CART-POOL, you can define a schedule that issues the following command periodically:

```
restore stgpool cart-pool
```

You can check for and replace any files that develop data-integrity problems in CART-POOL or in CART-BACKUP. For example, every three months, query the volumes in CART-POOL and CART-BACKUP by entering the following commands:

```
query volume stgpool=cart-pool  
query volume stgpool=cart-backup
```

Then issue the following command for each volume in CART-POOL and CART-BACKUP:

```
audit volume <volname> fix=yes
```

If a data integrity error occurs on a file in CART-POOL, that file is marked *damaged* and an error message is produced. If a data integrity error occurs on file in CART-BACKUP, that file is deleted and a message is produced.

Restore *damaged* primary files by entering:

```
restore stgpool cart-pool
```

Finally, create new copies in CART-BACKUP by entering:

```
backup stgpool cart-pool cart-backup
```

Backup and Recovery Scenarios

This section presents scenarios for protecting and recovering an ADSM server. You can modify the procedures to meet your needs.

These scenarios assume a storage hierarchy consisting of:

- The default random access storage pools (BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL)
- TAPEPOOL, a storage pool that uses cartridge media

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 cartridges 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 cartridge 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=devclassname 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 on page 366 onsite and updates their access to read-write.

Recovering to a Point in Time from a Disaster

In this scenario, the company's processor on which ADSM resides, the database, and all onsite storage pool volumes are destroyed by fire. An administrator restores the server to the point in time of the last backup by doing 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.
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.

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 STRRSTADSM command (see "Recovering by Using Database and Storage Pool Backups" on page 358).
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 DISCARDATA 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 using 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 178 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 365 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 copy volumes identified to 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.

Part 7. Appendix, Glossary, and Index

Appendix. Interface for Media Management Systems

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 define a USRDFN library, create the required four exits described in this appendix, and define the exits to ADSM. For guidance, see Chapter 5, "Using a Tape Management System with ADSM" on page 89.

Mount Exit Program

Parameters			
Required Parameter Group:			
1	Mount Information	Input	Char(*)
2	Completion Information	Output	Char(*)

The Mount Exit program is invoked when the ADSM server attempts to open a sequential media volume. When the user exit program is given control, it performs any tasks necessary to do one of the following actions:

- Immediately mount the volume.
- Prepare a media management system for the open volume.

If the exit program performs these tasks without error, it returns a completion value to the ADSM server indicating success. If the exit program encounters problems performing these tasks, it returns a completion value to the ADSM server indicating failure. When the exit program indicates a failure, the ADSM operation that initiates the mount request stops.

Required Parameter Group

Mount Information

INPUT; CHAR(*)

Information about the mount volume operation at the time the exit program is called. For details, see "Format of Mount Information."

Completion Information

OUTPUT; CHAR(*)

Information to be returned to the ADSM server when the exit program completes. For details, see "Format of Completion Information" on page 376.

Format of Mount Information

The following table shows the format of the mount information. For a description of the fields in this format, see "Mount Information Field Descriptions" on page 374.

Mount Exit Program

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Mount information length
4	4	BINARY(4)	Completion information length
8	8	CHAR(30)	Storage pool name
38	26	CHAR(32)	Device class name
70	46	CHAR(30)	Library name
100	64	CHAR(33)	Volume name
133	85	CHAR(8)	Volume prefix
141	8D	CHAR(3)	Volume use
144	90	CHAR(17)	File label
161	A1	CHAR(10)	Device type
171	AB	CHAR(10)	Format
181	B5	CHAR(1)	IDRC required
182	B6	CHAR(1)	Scratch mount
183	B7	CHAR(3)	Mount mode
186	BA	CHAR(1)	Select drive
187	BB	CHAR(4)	Mount wait

Mount Information Field Descriptions

Completion information length

The length, in bytes, of the completion information.

Device class name

The name of the ADSM device class.

Device type

The type of device required for this mount. Valid values are:

- 3590 A 3590 type device is required.
- CARTRIDGE A CARTRIDGE type device is required.
- REEL A REEL type device is required.
- 8MM An 8MM type device is required.
- QIC A QIC type device is required.

File label

The fully qualified file label that the server places in the labels on the volume.

Mount Exit Program

Format

The ADSM recording format to be used when writing data to this volume. Refer to the DEFINE DEVCLASS command for additional information about this field. Based on device type, the valid values are:

Device Type	Valid Values
3590	DRIVE Use highest supported format. 3590 Use 3590 Basic format.
CARTRIDGE	DRIVE Use highest supported format. 3480 Use 3480 Basic format. 3490 Use 3490 Basic format. 3490E Use 3490E Basic format.
REEL	DRIVE Use highest supported format. 1600 Use 1600 bits per inch format. 3200 Use 3200 bits per inch format. 6250 Use 6250 bits per inch format.
8MM	DRIVE Use highest supported format. 8200 Use 8200 format. 8500 Use 8500 format. 8700 Use 8700 format.
QIC	DRIVE Use highest supported format. 120 Use 120 quarter-inch format. 525 Use 525 quarter-inch format. 1000 Use 1000 quarter-inch format. 2000 Use 2000 quarter-inch format.

IDRC required

Indicates whether the server requires IDRC to be present on the drive used for this mount. Valid values are:

- 0 IDRC is *not* required for this mount.
- 1 IDRC is required for this mount.
- 2 Do *not* care about the presence of IDRC.

Library name

The name of the ADSM library that contains the drives that can be used for this mount.

Mount information length

The length, in bytes, of the mount information.

Mount mode

Indicates how the server intends to access the volume. Valid values are:

- R/W The volume will be mounted in Read-Write mode.
- R/O The volume will be mounted in Read-Only mode.

Mount Exit Program

Mount wait

The maximum length of time, in minutes, to wait for a volume to be mounted. Unlike other ADSM device classes, the MOUNTWAIT value is not used by the server to automatically cancel this mount request.

Scratch mount

Indicates whether the server desires to mount a scratch volume. Valid values are:

- 0 This is *not* a scratch mount.
- 1 This is a scratch mount.

Select drive

Indicates when the exit must select a drive for the server. Valid values are:

- 0 The exit does *not* select a drive.
- 1 The exit must select a drive.

Storage pool name

The name of a specific ADSM storage pool or, for volumes that do not belong to a storage pool, the valid values are:

- EXPORT Volume used for IMPORT/EXPORT operations.
- DUMPDB Volume used for DUMP/LOAD operations or for database BACKUP/RESTORE operations.

Volume name

Volume identifier of the volume to be mounted.

Volume prefix

The high-level qualifier of the file label that the server places in the labels on the volume.

Volume use

Indicates what use ADSM makes of this volume. Valid values are:

- BFS Volume used for BACKUP/ARCHIVE operations.
- EXP Volume used for IMPORT/EXPORT operations.
- DMP Volume used for DUMP/LOAD operations or for database BACKUP/RESTORE operations.
- OTH Volume used for OTHER operations.

Format of Completion Information

The following table shows the format of the completion information. For a description of the fields in this format, see "Completion Information Field Descriptions" on page 377.

Offset			
Dec	Hex	Type	Field
0	0	CHAR(1)	Return code
1	1	CHAR(33)	Volume name
34	22	CHAR(10)	Device name

Mount Exit Program

Completion Information Field Descriptions

Device name

If the exit is performing drive selection, this field should be set to the device name to be used by the server. See "Mount Information Field Descriptions" on page 374.

Return code

Indicates to the server whether or not mount exit processing has completed successfully. Valid values are:

- 0 Mount exit processing completed successfully.
- 1 An error occurred during mount exit processing.

Note: If the user exit program specifies 1 for this field, the server ignores the other fields. Additionally, the ADSM operation which initiated the mount request will terminate.

Volume name

For a scratch mount, the exit may provide a volume identifier to be used by the server. If this field is set to a non-blank value, the server attempts to use that value as the volume identifier for a scratch mount. See "Mount Information Field Descriptions" on page 374.

Error Messages

- ANR8274E MOUNT EXIT cannot be located.
- ANR8281E MOUNT EXIT cannot be accessed.
- ANR8285E MOUNT EXIT returns a non-zero result.
- ANR8286E MOUNT EXIT overwrites server memory.
- ANR8287E Exception occurs in MOUNT EXIT.

Dismount Exit Program

Dismount Exit Program

Parameters

Required Parameter Group:

1	Dismount Information	Input	Char(*)
2	Completion Information	Output	Char(*)

The Dismount Exit program is invoked when the ADSM server is finished with a sequential media volume that it had previously mounted. When the user exit program is given control, it performs any tasks related to the dismount volume operation.

If the exit program performs these tasks without error, it returns a completion value to the ADSM server indicating success. If the exit program encounters problems performing these tasks, it returns a completion value to the ADSM server indicating failure. When the exit program indicates a failure, the ADSM server issues an error message indicating that the dismount failed. However, the ADSM operation that initiated the dismount request will complete normally.

Required Parameter Group

Dismount Information

INPUT; CHAR(*)

Information about the dismount volume operation at the time the exit program is called. For details, see "Format of Dismount Information."

Completion Information

OUTPUT; CHAR(*)

Information to be returned to the ADSM server when the exit program completes. For details, see "Format of Completion Information" on page 381.

Format of Dismount Information

The following table shows the format of the dismount information. For a description of the fields in this format, see "Dismount Information Field Descriptions" on page 379.

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Dismount information length
4	4	BINARY(4)	Completion information length
8	8	CHAR(30)	Storage pool name
38	26	CHAR(32)	Device class name
70	46	CHAR(30)	Library name
100	64	CHAR(33)	Volume name
133	85	CHAR(8)	Volume prefix

Dismount Exit Program

Offset			
Dec	Hex	Type	Field
141	8D	CHAR(3)	Volume use
144	90	CHAR(17)	File label
161	A1	CHAR(10)	Device type
171	AB	CHAR(10)	Format
181	B5	CHAR(3)	Mount mode
184	B8	CHAR(1)	Select drive
185	B9	CHAR(10)	Device name

Dismount Information Field Descriptions

Completion information length

The length, in bytes, of the completion information.

Device class name

The name of the ADSM device class.

Device name

The device name of the drive on which the volume was mounted.

Device type

Indicates the type of device on which the volume is mounted. Valid values are:

3590	A 3590 type device is required.
CARTRIDGE	A CARTRIDGE type device is required.
REEL	A REEL type device is required.
8MM	An 8MM type device is required.
QIC	A QIC type device is required.

Dismount information length

The length, in bytes, of the dismount information.

File label

The fully qualified file label that the server places in the labels on the volume.

Format

The ADSM recording format to be used when writing data to this volume. See the DEFINE DEVCLASS command for additional information about this field.

Based on device type, the valid values are:

Device Type Valid Values

3590	DRIVE	Use highest supported format.
	3590	Use 3590 Basic format.
CARTRIDGE	DRIVE	Use highest supported format.
	3480	Use 3480 Basic format.
	3490	Use 3490 Basic format.
	3490E	Use 3490E Basic format.

Dismount Exit Program

REEL	DRIVE	Use highest supported format.
	1600	Use 1600 bits per inch format.
	3200	Use 3200 bits per inch format.
	6250	Use 6250 bits per inch format.
8MM	DRIVE	Use highest supported format.
	8200	Use 8200 format.
	8500	Use 8500 format.
	8700	Use 8700 format.
QIC	DRIVE	Use highest supported format.
	120	Use 120 quarter-inch format.
	525	Use 525 quarter-inch format.
	1000	Use 1000 quarter-inch format.
	2000	Use 2000 quarter-inch format.

Library name

The name of the ADSM library that contains the drive on which the volume is mounted.

Mount mode

Indicates how the server accessed the volume. Valid values are:

R/W The volume was mounted in Read-Write mode.

R/O The volume was mounted in Read-Only mode.

Select drive

Indicates whether the mount exit selected a drive for the server. Valid values are:

0 The mount exit did *not* select a drive.

1 The mount exit selected a drive.

Storage pool name

The name of a specific ADSM storage pool or, for volumes that do not belong to a storage pool, the valid values are:

EXPORT Volume used for IMPORT/EXPORT operations.

DUMPDB Volume used for DUMP/LOAD operations or for database BACKUP/RESTORE operations.

Volume name

Volume identifier of the volume being dismounted.

Volume prefix

The high-level qualifier of the file label that the server places in the labels on the volume.

Volume use

Indicates what use ADSM makes of this volume. Valid values are:

BFS Volume used for BACKUP/ARCHIVE operations.

EXP Volume used for IMPORT/EXPORT operations.

Dismount Exit Program

DMP Volume used for DUMP/LOAD operations or for database
BACKUP/RESTORE operations.
OTH Volume use for OTHER operations.

Format of Completion Information

The following table shows the format of the completion information. For a description of the fields in this format, see "Completion Information Field Descriptions."

Offset			
Dec	Hex	Type	Field
0	0	CHAR(1)	Return code

Completion Information Field Descriptions

Return code

Indicates to the server whether or not dismount exit processing has completed successfully. Valid values are:

- 0 Dismount exit processing completed successfully.
- 1 An error occurred during dismount exit processing.

Note: If the user exit program specifies 1 for this field, the server issues an error message indicating that the dismount has failed.

Error Messages

ANR8274E DISMOUNT EXIT cannot be located.
ANR8281E DISMOUNT EXIT cannot be accessed.
ANR8285E DISMOUNT EXIT returns a non-zero result.
ANR8286E DISMOUNT EXIT overwrites server memory.
ANR8287E Exception occurs in DISMOUNT EXIT.

Deletion Exit Program

Deletion Exit Program

Parameters

Required Parameter Group:

1	Deletion Information	Input	Char(*)
2	Completion Information	Output	Char(*)

The Deletion Exit program is invoked when the ADSM server has deleted a sequential media volume from its database. The ADSM server no longer wishes to retain ownership of the volume. When the user exit program is given control, it performs any tasks related to the delete volume operation.

If the exit program performs these tasks without error, it returns a completion value to the ADSM server indicating success. If the exit program encounters problems performing these tasks, it returns a completion value to the ADSM server indicating failure. When the exit program indicates a failure, the ADSM server issues a message stating the exit program has encountered an error. However, the ADSM operation will complete normally because the volume has been deleted from the server database before invoking the deletion exit.

Required Parameter Group

Deletion Information

INPUT; CHAR(*)

Information about the delete volume operation at the time the exit program is called. For details, see "Format of Deletion Information."

Completion Information

OUTPUT; CHAR(*)

Information to be returned to the ADSM server when the exit program completes. For details, see "Format of Completion Information" on page 381.

Format of Deletion Information

The following table shows the format of the deletion information. For a description of the fields in this format, see "Deletion Information Field Descriptions" on page 383.

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Deletion information length
4	4	BINARY(4)	Completion information length
8	8	CHAR(30)	Storage pool name
38	26	CHAR(32)	Device class name
70	46	CHAR(30)	Library name
100	64	CHAR(33)	Volume name

Deletion Exit Program

Offset			
Dec	Hex	Type	Field
133	85	CHAR(8)	Volume prefix
141	8D	CHAR(3)	Volume use
144	90	CHAR(10)	Device type

Deletion Information Field Descriptions

Completion information length

The length, in bytes, of the completion information.

Deletion information length

The length, in bytes, of the deletion information.

Device class name

The name of the ADSM device class.

Device type

Indicates the type of device on which the volume is mounted. Valid values are:

3590	A 3590 type device is required.
CARTRIDGE	A CARTRIDGE type device is required.
REEL	A REEL type device is required.
8MM	An 8MM type device is required.
QIC	A QIC type device is required.

Library name

The name of the ADSM library that contains the drive on which the volume is mounted.

Storage pool name

The name of a specific ADSM storage pool or, for volumes that do not belong to a storage pool, the valid values are:

EXPORT	Volume used for IMPORT/EXPORT operations.
DUMPDB	Volume used for DUMP/LOAD operations or for database BACKUP/RESTORE operations.

Volume name

Volume identifier of the volume being dismounted.

Volume prefix

The high-level qualifier of the file label that the server places in the labels on the volume.

Volume use

Indicates what use ADSM makes of this volume. Valid values are:

BFS	Volume used for BACKUP/ARCHIVE operations.
EXP	Volume used for IMPORT/EXPORT operations.
DMP	Volume used for DUMP/LOAD operations or for database BACKUP/RESTORE operations.
OTH	Volume use for OTHER operations.

Deletion Exit Program

Format of Completion Information

The following table shows the format of the completion information. For a description of the fields in this format, see “Completion Information Field Descriptions.”

Offset			
Dec	Hex	Type	Field
0	0	CHAR(1)	Return code

Completion Information Field Descriptions

Return code

Indicates to the server whether or not deletion exit processing completed successfully. Valid values are:

- 0 Deletion exit processing completed successfully.
- 1 An error occurred during deletion exit processing.

Error Messages

- ANR8274E DELETION EXIT could not be located.
- ANR8281E DELETION EXIT could not be accessed.
- ANR8285E DELETION EXIT returned a non-zero result.
- ANR8286E DELETION EXIT has overwritten server memory.
- ANR8287E Exception occurred in DELETION EXIT.

Expiration Exit Program

Expiration Exit Program

Parameters			
Required Parameter Group:			
1	Expiration Information	Input	Char(*)
2	Completion Information	Output	Char(*)

The Expiration Exit program is invoked when the ADSM server determines that it must expire the active data on a sequential media volume, but does not delete the volume from its database. The ADSM server must retain ownership of the volume. When the user exit program is given control, it performs any tasks related to the expire volume operation.

If the exit program performs these tasks without error, it returns a completion value to the ADSM server indicating success. If the exit program encounters problems performing these tasks, it returns a completion value to the ADSM server indicating failure. When the exit program indicates a failure, the ADSM operation that initiated the expiration request will terminate.

Required Parameter Group

Expiration Information

INPUT; CHAR(*)

Information about the expire volume operation at the time the exit program is called. For details, see "Format of Deletion Information" on page 382.

Completion Information

OUTPUT; CHAR(*)

Information to be returned to the ADSM server when the exit program completes. For details, see "Format of Completion Information" on page 387.

Format of Expiration Information

The following table shows the format of the expiration information. For a description of the fields in this format, see "Expiration Information Field Descriptions" on page 386.

Offset			
Dec	Hex	Type	Field
0	0	BINARY(4)	Expiration information length
4	4	BINARY(4)	Completion information length
8	8	CHAR(30)	Storage pool name
38	26	CHAR(32)	Device class name
70	46	CHAR(30)	Library name
100	64	CHAR(33)	Volume name

Expiration Exit Program

Offset			
Dec	Hex	Type	Field
133	85	CHAR(8)	Volume prefix
141	8D	CHAR(3)	Volume use
144	90	CHAR(10)	Device type
154	9A	CHAR(10)	Format
164	A4	CHAR(10)	Device name

Expiration Information Field Descriptions

Completion information length

The length, in bytes, of the completion information.

Device class name

The name of the ADSM device class.

Device name

The device name of the drive on which the volume is mounted.

Device type

Indicates the type of device on which the volume is mounted. Valid values are:

3590	A 3590 type device is required.
CARTRIDGE	A CARTRIDGE type device is required.
REEL	A REEL type device is required.
8MM	An 8MM type device is required.
QIC	A QIC type device is required.

Expiration information length

The length, in bytes, of the expiration information.

Format

The ADSM recording format to be used when writing data to this volume. See to the DEFINE DEVCLASS command for additional information about this field.

Based on device type, the valid values are:

Device Type	Valid Values
3590	DRIVE Use highest supported format.
	3590 Use 3590 Basic format.
	CARTRIDGE
CARTRIDGE	DRIVE Use highest supported format.
	3480 Use 3480 Basic format.
	3490 Use 3490 Basic format.
	3490E Use 3490E Basic format.
REEL	DRIVE Use highest supported format.
	1600 Use 1600 bits per inch format.
	3200 Use 3200 bits per inch format.
	6250 Use 6250 bits per inch format.

Expiration Exit Program

8MM	DRIVE	Use highest supported format.
	8200	Use 8200 format.
	8500	Use 8500 format.
	8700	Use 8700 format.
QIC	DRIVE	Use highest supported format.
	120	Use 120 quarter-inch format.
	525	Use 525 quarter-inch format.
	1000	Use 1000 quarter-inch format.
	2000	Use 2000 quarter-inch format.

Library name

The name of the ADSM library that contains the drive on which the volume is mounted.

Storage pool name

The name of a specific ADSM storage pool or, for volumes that do not belong to a storage pool, the valid values are:

EXPORT	Volume used for IMPORT/EXPORT operations.
DUMPDB	Volume used for DUMP/LOAD operations or for database BACKUP/RESTORE operations.

Volume name

Volume identifier of the volume being dismounted.

Volume prefix

The high-level qualifier of the file label that the server places in the labels on the volume.

Volume use

Indicates what use ADSM makes of this volume. Valid values are:

BFS	Volume used for BACKUP/ARCHIVE operations.
EXP	Volume used for IMPORT/EXPORT operations.
DMP	Volume used for DUMP/LOAD operations or for database BACKUP/RESTORE operations.
OTH	Volume use for OTHER operations.

Format of Completion Information

The following table shows the format of the completion information. For a description of the fields in this format, see "Completion Information Field Descriptions" on page 388.

Offset			
Dec	Hex	Type	Field
0	0	CHAR(1)	Return code

Expiration Exit Program

Completion Information Field Descriptions

Return code

Indicates to the server whether or not expiration exit processing completed successfully. Valid values are:

- 0 Expiration exit processing completed successfully.
- 1 An error occurred during expiration exit processing.

Note: If the user exit program specifies 1 for this field, the ADSM operation that initiated the expiration request will terminate.

Error Messages

- ANR8274E EXPIRATION EXIT cannot be located.
- ANR8281E EXPIRATION EXIT cannot be accessed.
- ANR8285E EXPIRATION EXIT returns a non-zero result.
- ANR8286E EXPIRATION EXIT overwrites server memory.
- ANR8287E Exception occurs in EXPIRATION EXIT.

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.

action choice. A choice in a pull-down menu that causes an action. See also *routing choice*.

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.

ADSM application programming 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 programming 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*.

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.

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*.

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.

bus converter. A device that translates between different Hewlett-Packard internal I/O bus architectures.

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. A sequential storage media that contains magnetic tape in a protective housing. Contrast with *tape reel*.

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 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 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 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.

command line interface. A type of user interface where commands are specified on the command line when the backup-archive or administrative client is started. Contrast with *graphical user interface*.

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 SAA architectural 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.

contextual help. A type of online help that provides specific information for each selectable object, menu choice, notebook tab, field, and control or push button in a window.

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. A utility that copies database entries to media for later reload in case a catastrophic error should occur.

database load. A utility that copies 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*.

data 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). Synonymous with *server storage*. See *primary storage pool*, *copy storage pool*, *storage pool volume*, and *volume*.

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.

definition. Server control information that includes administrator, client node, and policy definitions. A definition can be exported from a source server to external media so that it can be imported to a target server at a later date.

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. On-screen representation of a desk top.

desktop client. The group of clients supported by ADSM that are not UNIX-based and are not OpenEdition MVS. For example, a DOS client is a desktop client.

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 BACKUPOOL, 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 AIX 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.

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.

diskette. A small, magnetic disk enclosed in a jacket.

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.

dsm.opt file. See *client user options file*.

dsm.serv.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

general help. A type of online help that provides an overview of the function of the window.

graphical user interface (GUI). A type of user interface that takes advantage of a high-resolution monitor, including some combination of graphics, the object-action paradigm, the use of pointing devices, menu bars, overlapping windows, and icons. See *windowed interface*. Contrast with *command line interface*.

group of mirrored volumes. One, two, or three volume copies defined to the database or recovery log.

Each volume copy in the group contains exactly the same portion of the database or recovery log. See *mirroring*.

GUI. Graphical user interface.

H

handle. A data structure that is a temporary local identifier for an object. A handle identifies an object at a specific location by binding it.

HDA. Head-disk assembly.

head-disk assembly (HDA). A field replaceable unit in a direct access storage device containing the disks and actuators.

help index. A type of online help that provides an alphabetic listing of all help topics.

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).

inconsistencies. Any discrepancy between the information recorded in the database about backed up or archived files and the actual data associated with backed up or archived files residing in server storage.

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

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, 2²⁰ 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. See *group of mirrored volumes*.

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*.

notebook. A graphical representation that resembles a spiral-bound notebook that contains pages separated into sections by tabbed divider-pages. A user can “turn” the pages of a notebook to move from one section to another.

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 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 disk. A disk that contains data readable by optical techniques.

optical drive. A drive mechanism that rotates an optical disc.

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.

programmable workstation communication services (PWSCS). A product that provides transparent high performance communications between programs running on workstations or on host systems.

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.

PWSCS. Programmable workstation communication services.

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.

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. Defines a client node or administrator who can access the server. See *registration*.

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*.

RLIO. Record Level Input/Output.

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*.

routing choice. A choice in a pull-down menu that, when selected, brings up another window. See also *action choice*.

S

SAA. Systems Application Architecture.

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.

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. A program that provides services to other programs (clients).

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 program. 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-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). Synonymous with *data storage*. 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.

signal. (1) A simple method of communication between two processes. One process can inform the other process when an event occurs. (2) In operating system operations, a method of inter-process communication that simulates software interrupts.

signal handler. A subroutine called when a signal occurs.

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. A recording medium consisting of a long, narrow, flexible strip with a magnetic coating wound onto a reel or into a cartridge. See *cartridge* and *tape reel*.

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 reel. A cylinder with flanges on which magnetic tape is wound. Devices such as the 3420 9-track tape device support tape reels. Contrast with *cartridge*.

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.

task help. A type of online help that provides a list of tasks that can be completed with a selected object. When you select a task, the help provides step-by-step information on how to complete the task.

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* 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*.

window. A part of a display screen with visible boundaries in which information is displayed.

windowed interface. A type of user interface that is either a graphical user interface or a text based interface. The text based interface maintains a close affinity to the graphical user interface, including action bars and their associated pull-downs and windows. See *graphical user interface*.

workstation. A personal computer system capable of maintaining data files.

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

Numerics

3490 drive, example setup 69
349X library 84
3590 drive
 configuring 79
 device class, defining 19
9427 library, example setup 73

A

absolute mode, description of 224
access mode 180
accounting record
 description of 274
 determining for storage pool 159, 162
 monitoring 274
ACTIVATE POLICYSET command 229
activity log
 adjusting the size 272
 description of 272
 monitoring 272
 querying 273
 setting the retention period 273
administrative client
 description of 3
 viewing information after IMPORT or EXPORT 321
administrative commands
 AUDIT LIBVOLUME 88
 AUDIT LICENSE 298
 AUDIT VOLUME 191
 BACKUP DB 357
 BACKUP DEVCONFIG 356
 BACKUP STGPOOL 345
 BACKUP VOLHISTORY 355
 CHECKIN LIBVOLUME 86
 CHECKOUT LIBVOLUME 87
 DEFINE DEVCLASS 122
 DEFINE DRIVE 113
 DEFINE LIBRARY 19, 80, 84, 110
 DEFINE SCHEDULE 250
 DEFINE STGPOOL 161
 DEFINE VOLUME 85, 183
 DELETE DEVCLASS 124
 DELETE DRIVE 114
 DELETE LIBRARY 112
 administrative commands (*continued*)
 DELETE LOGVOLUME 288
 DELETE VOLHISTORY 354, 355
 DELETE VOLUME 197, 198
 DISMOUNT VOLUME 105
 EXTEND LOG 351
 GRANT AUTHORITY 300
 HELP 275
 MOVE DATA 194
 QUERY ACTLOG 273
 QUERY DB 287
 QUERY DBBACKUPTRIGGER 353
 QUERY DEVCLASS 123
 QUERY DRIVE 113
 QUERY LIBRARY 111
 QUERY LICENSE 299
 QUERY MOUNT 104
 QUERY OCCUPANCY 172, 173, 174
 QUERY OPTION 271
 QUERY PROCESS 196
 QUERY REQUEST 101
 QUERY STGPOOL 166, 167, 331
 QUERY VOLHISTORY 354
 QUERY VOLUME 185, 197
 REGISTER ADMIN 300
 REGISTER LICENSE 298
 RENAME ADMIN 306
 RESET DBMAXUTILIZATION 279, 280
 RESET LOGCONSUMPTION 351
 RESET LOGMAXUTILIZATION 279, 280
 RESTORE STGPOOL 346, 367
 RESTORE VOLUME 369
 SET ACCOUNTING 274
 SET ACTLOGRETENTION 272
 SET AUTHENTICATION 299
 SET LICENSEAUDITPERIOD 299
 SET LOGMODE 354
 SET PASSEXP 299
 SET SCHEDMODES 246
 SET SERVERNAME 271
 UPDATE ADMIN 300
 UPDATE DBBACKUPTRIGGER 353
 UPDATE DEVCLASS 127
 UPDATE DRIVE 114
 UPDATE LIBRARY 111
 UPDATE LIBVOLUME 85, 87

- administrative commands (*continued*)
 - UPDATE SCHEDULE 250
 - UPDATE VOLUME 183
 - administrative privilege class
 - analyst 304
 - description of 300
 - granting authority 300
 - operator 303
 - policy 301, 302
 - revoking all 305
 - storage 302, 303
 - system 301
 - administrator
 - authorizing to manage a policy domain 300
 - description of 3
 - locking 307
 - managing registration 295
 - querying 307
 - registering 300
 - removing 306
 - renaming 306
 - unlocking 307
 - updating 300
 - viewing information about 307
 - analyst privilege class
 - changing administrative authority 305
 - description of 304
 - granting 304
 - API
 - See application programming interface
 - application client, registering 315
 - application programming interface
 - compression option 315
 - deletion option 316
 - description of 3
 - registering to server 315
 - archive
 - amount of space used 174
 - defining criteria 218
 - description of 26
 - processing 214
 - archive copy group
 - defining 227, 229
 - deleting 234
 - description of 206
 - archive file management 204
 - archiving a file 204, 214
 - AS400MLB libraries
 - about 19
 - adding volumes to 77
 - AS400MLB libraries (*continued*)
 - updating 111
 - ASSIGN DEFMGMTCLASS command 229
 - assigned capacity 279, 285
 - association
 - defining 242
 - deleting 259
 - description of 250
 - querying 258
 - viewing information about 258
 - audit license 301
 - AUDIT LICENSE command 298
 - AUDIT VOLUME command 189, 191
 - auditing
 - library's volume inventory 88
 - license, automatic by server 298
 - multiple volumes in sequential access storage pool 192
 - single volume in sequential access storage pool 193
 - volume in disk storage pool 191
 - authentication, client/server 299
 - authority
 - granting to administrators 300
 - revoking 304
 - automated library
 - auditing 88
 - categories for volumes 84
 - changing volume status 87
 - informing server of new volumes 86
 - removing volumes 87
 - returning volumes 88
 - scratch and private volumes 85
 - automatic cartridge loader, using 81
 - automating
 - client operations 241
 - server operations 240
 - auxiliary storage pools 283, 347
- ## B
- backup
 - amount of space used 174
 - database 353, 357
 - defining criteria 218
 - description of 26
 - file management 204
 - file, by client 204, 212, 214
 - incremental 204, 212
 - selective 204, 214

- backup (*continued*)
 - storage pool 345
 - when to perform for database 350
- backup copy group
 - defining 223, 227
 - deleting 234
 - description of 206
 - frequency 212
 - mode 212
 - serialization 212
- BACKUP DB command 357
- BACKUP DEVCONFIG command 356
- backup period, specifying for incremental 246
- Backup Recovery and Media Services 89
- BACKUP STGPOOL command 345
- BACKUP VOLHISTORY command 354
- backup-archive client
 - description of 3
 - registering 309
- binding
 - description of 209
 - file to a management class 209
- BRMS
 - See Backup Recovery and Media Services
- buffer pool 290
- BUFPOOLSIZE option 291

C

- cache
 - description of 25
 - disabling for disk storage pools 143
 - enabling for disk storage pools 143, 160
 - monitoring utilization on disk 171
- CANCEL PROCESS command 170, 269
- CANCEL SESSION command 267
- capacity, assigned 279, 285
- cartridge 116
- category, 349X library 84
- central scheduling
 - controlling the workload 246
 - coordinating 244
 - description of 26, 239
- CHGSVRADSM command 296
- class, administrator privilege
 - analyst 304
 - description of 300
 - granting authority 300
 - operator 303
 - policy 301, 302

- class, administrator privilege (*continued*)
 - revoking all 305
 - storage 302, 303
 - system 301
- class, device
 - amount of space used 173
 - defining for database backup 349
 - deleting 124
 - description of 25
 - DISK 115
 - FILE 115
 - requesting information about 123
 - selecting for import and export 319
 - sequential 116
 - tape 116
- class, policy privilege
 - changing administrative authority 305
 - description of 302
 - granting 302
- class, storage privilege
 - changing administrative authority 305
 - description of 303
 - granting 303
- client
 - administrative 3
 - application 315
 - backup-archive 26
 - space management 3
- client migration 215
- client node
 - amount of space used 173
 - locking 311
 - managing registration 295, 308
 - querying 311
 - registering 309, 315
 - removing 315
 - renaming 311
 - setting password authentication 299
 - setting scheduling mode 246
 - unlocking 311
 - updating 310
 - viewing information about 311
- client queries to the server, setting the frequency 249
- client session
 - canceling 267
 - managing 265
 - querying 265
 - viewing information about 265
- client system options file 316

- client-polling scheduling 245, 248
- client/server, description of 3
- closed registration
 - description of 309
 - setting 308
- collocation
 - changing, effect of 148
 - definition 144, 160, 163
 - description of 25
 - determining whether to use collocation 144, 160, 163
 - enabling for sequential storage pool 144, 160, 163
 - how it affects reclamation 153
 - how the server selects volumes when disabled 148
 - how the server selects volumes when enabled 147
 - migration thresholds 142
- command retry attempts
 - setting the amount of time between 249
 - setting the number of 249
- commands, administrative
 - AUDIT LIBVOLUME 88
 - AUDIT LICENSE 298
 - AUDIT VOLUME 191
 - BACKUP DB 357
 - BACKUP DEVCONFIG 356
 - BACKUP STGPOOL 345
 - BACKUP VOLHISTORY 355
 - CHECKIN LIBVOLUME 86
 - CHECKOUT LIBVOLUME 87
 - DEFINE DEVCLASS 122
 - DEFINE DRIVE 113
 - DEFINE LIBRARY 19, 80, 84, 110
 - DEFINE SCHEDULE 250
 - DEFINE STGPOOL 161
 - DEFINE VOLUME 85, 183
 - DELETE DEVCLASS 124
 - DELETE DRIVE 114
 - DELETE LIBRARY 112
 - DELETE LOGVOLUME 288
 - DELETE VOLHISTORY 354, 355
 - DELETE VOLUME 197, 198
 - DISMOUNT VOLUME 105
 - EXTEND LOG 351
 - GRANT AUTHORITY 300
 - HELP 275
 - MOVE DATA 194
 - QUERY ACTLOG 273
 - QUERY DB 287
 - QUERY DBBACKUPTRIGGER 353
 - QUERY DEVCLASS 123
- commands, administrative (*continued*)
 - QUERY DRIVE 113
 - QUERY LIBRARY 111
 - QUERY LICENSE 299
 - QUERY MOUNT 104
 - QUERY OCCUPANCY 172, 173, 174
 - QUERY OPTION 271
 - QUERY PROCESS 196
 - QUERY REQUEST 101
 - QUERY STGPOOL 166, 167, 331
 - QUERY VOLHISTORY 354
 - QUERY VOLUME 185, 197
 - REGISTER ADMIN 300
 - REGISTER LICENSE 298
 - RENAME ADMIN 306
 - RESET DBMAXUTILIZATION 279, 280
 - RESET LOGCONSUMPTION 351
 - RESET LOGMAXUTILIZATION 279, 280
 - RESTORE STGPOOL 346, 367
 - RESTORE VOLUME 369
 - SET ACCOUNTING 274
 - SET ACTLOGRETENTION 272
 - SET AUTHENTICATION 299
 - SET LICENSEAUDITPERIOD 299
 - SET LOGMODE 354
 - SET PASSEXP 299
 - SET SCHEDMODES 246
 - SET SERVERNAME 271
 - UPDATE ADMIN 300
 - UPDATE DBBACKUPTRIGGER 353
 - UPDATE DEVCLASS 127
 - UPDATE DRIVE 114
 - UPDATE LIBRARY 111
 - UPDATE LIBVOLUME 85, 87
 - UPDATE SCHEDULE 250
 - UPDATE VOLUME 183
- compression
 - option for API 315
 - setting at client registration 309
 - tape volume capacity, effect on 125
- configuration file, device
 - backup 355
 - example 357
 - information 355
 - recreating 356
- configuring
 - devices for ADSM with configuration utility 62
 - devices, automated library example 73
 - devices, manual library example 69
 - IBM 3590 drives 79

- configuring (*continued*)
 - non-IBM devices 79
 - planning your storage environment 16
- console mode 321
- COPY DOMAIN command 220
- copy group
 - deleting 234
 - description of 26
- COPY MGMTCLASS command 222
- COPY POLICYSET command 221
- COPY SCHEDULE command 254
- CRTVOLADSM command 181

D

- data
 - considering user needs for recovering 17
 - exporting 317
 - importing 317
 - protection, methods 341
- data movement, querying about the process 196
- data storage
 - client files, process for storing 4
 - concepts overview 4
 - considering user needs for recovering 17
 - deleting files from 197
 - evaluating 16
 - managing 4
 - monitoring 189
 - planning 16
 - tailoring definitions 333
 - using disk devices 55
 - using media management interface 89
 - using tape devices 59
- database
 - adding space to (CRTVOLADSM) 282
 - available space 279, 280
 - backup 357
 - backup trigger 352
 - buffer pool 290, 291
 - committing data to 291
 - defining a volume 283
 - defining mirrored volumes 347
 - deleting a volume 288
 - deleting space 286
 - description of 26, 277
 - determining how much space is allocated 278, 280
 - ensuring integrity of 27
 - estimating the amount of space needed 280
 - logical volume 278, 280

- database (*continued*)
 - managing 277
 - mirroring 347
 - monitoring space 279, 280
 - monitoring the buffer 291
 - optimizing performance 290
 - querying the buffer pool 290
 - recovering 358
 - reducing capacity 288
 - resetting buffer pool statistics 290
 - restoring 351
 - storage pool size effect 277
 - transactions 277, 278
 - viewing information about 290
 - volume placement 283
- database backup and recovery
 - defining device classes 349
 - full backup 351
 - incremental backup 351
 - point-in-time 359
 - roll-forward 343, 363
 - to most current state 363
 - trigger 352
- database backup trigger and roll-forward mode 363
- database recovery
 - example recovery procedures 365
 - general strategy 341
 - methods 341
 - providing 341
 - when to backup 350
- day of the week, description of 250
- default management class, description of 205
- default management classes 208
- DEFINE ASSOCIATION command 242
- DEFINE COPYGROUP command 223, 227, 229
- DEFINE DBBACKUPTRIGGER 349, 352
- DEFINE DBCOPY command 348
- DEFINE DBVOLUME command 283
- DEFINE LOGCOPY command 358
- DEFINE LOGVOLUME command 283
- DEFINE MGMTCLASS command 222
- DEFINE POLICYSET command 221
- DEFINE SCHEDULE command 250
- DEFINE STGPOOL command 161
- DEFINE VOLUME command 183
- delete
 - empty volume 197, 355
 - files 197
 - scratch volume 135, 355
 - storage volume 198

- delete (*continued*)
 - volume with residual data 198
- DELETE ASSOCIATION command 259
- DELETE COPYGROUP command 234
- DELETE DBBACKUPTRIGGER 352
- DELETE DBVOLUME command 288
- DELETE DOMAIN command 235
- DELETE EVENT command 258
- DELETE FILESPACE command 314
- DELETE LOGVOLUME command 288
- DELETE MGMTCLASS command 235
- DELETE POLICYSET command 235
- DELETE SCHEDULE command 255
- DELETE STGPOOL command 175
- DELETE VOLHISTORY command 354, 355
- DELETE VOLUME command 197, 198
- deletion exit program 382
- DEVCONFIG option 355
- device class
 - amount of space used 173
 - defining for database backup 349
 - deleting 124
 - description of 25
 - DISK 115
 - FILE 115
 - requesting information about 123
 - selecting for import and export 319
 - sequential 116
 - tape 116
- device configuration file
 - backup 355
 - example 357
 - information 355
 - recreating 356
- device type
 - 3590 115, 179
 - 8MM 115, 179
 - CARTRIDGE 115, 179
 - DISK 115, 179
 - FILE 115, 179
 - REEL 115, 179
- DISABLE command 267
- disaster recovery
 - auditing storage pool volumes 364
 - example recovery procedures 365
 - general strategy 341
 - methods 341
 - providing 341
 - when to backup 341, 350
- disk device class, defined 115
- disk storage pool
 - cache, use of 143
 - estimating space 157
 - estimating space for archived files 158
 - estimating space for backed up files 157
 - migration threshold 139
- dismount exit program 378
- documentation, user xx
- drive
 - defining 113
 - deleting 114
 - querying 113
 - updating 114
- DSPTAPCGY command 74
- DSPTAPSTS command 74
- DSPVOLADSM command 356, 361
- dynamic serialization, description of 223, 228

E

- ENABLE command 268
- ENDSVRADSM command 265
- event record
 - deleting 258
 - description of 239
 - removing from the database 257
- event retention period 257
- event, description of 239
- exit program
 - deletion 382
 - dismount 378
 - expiration 385
 - mount 373
- EXPINTERVAL option 211, 231
- expiration date, setting 251
- expiration exit program 385
- expiration processing
 - description 154
 - files eligible 211
 - starting 231
- EXPIRE INVENTORY command 211, 231
- export
 - labelling tapes 320
 - monitoring 320
 - planning for sequential media 319
 - PREVIEW parameter 318
 - querying about a process 320
 - querying the activity log 323
 - using scratch media 320

- export (*continued*)
 - viewing information about a process 320
- EXPORT ADMIN command 326
- EXPORT commands 321
- EXPORT NODE command 327
- EXPORT POLICY command 328
- EXPORT SERVER command 319, 326
- exporting
 - administrator data 326
 - client node data 327
 - data to tape 324
 - description of 26, 317
 - policy data 328
 - server data 326
- EXTEND DB command 285
- EXTEND LOG command 285

F

- file data, importing 317
- FILE device type
 - defining device class 115
 - deleting scratch volumes 355
- file size, determining maximum for storage pool 159
- file space
 - deleting 314
 - description of 26, 313
 - querying 313
 - renaming 337
 - viewing information about 313
- files, server migration of 139
- formatting a volume for a storage pool 181
- frequency, description of 224

G

- GRANT AUTHORITY command 300

H

- HALT command 264
- halting the server 264
- HELP command 275
- hierarchical storage management
 - archive policy, relationship to 215
 - backup policy, relationship to 215
 - description 204
 - files, destination for 222
 - migration of client files
 - description 205
 - eligibility 215

- hierarchical storage management (*continued*)
 - policy for, setting 222
 - premigration 205
 - recall of migrated files 205
 - reconciliation between client and server 205
 - selective migration 205
 - setting policy for 215, 222
 - space-managed file, definition 204
 - stub file 205
- hierarchy, storage
 - defining in reverse order 161
 - establishing 135
- HSM
 - See hierarchical storage management

I

- import
 - monitoring 320
 - PREVIEW parameter 318, 329
 - querying about a process 320
 - querying the activity log 323
 - recovering from an error 337
 - viewing information about a process 320
- IMPORT ADMIN command 328
- IMPORT commands 321
- IMPORT NODE command 328, 335
- IMPORT POLICY command 328
- IMPORT SERVER command 328, 335
- importing
 - data 328
 - date of creation 335
 - description of 26, 317
 - directing messages to an output file 332
 - duplicate file spaces 335
 - file data 334
 - policy definitions 332
 - server control data 333
 - server storage definitions 331, 333
 - subsets of information 336
- include-exclude file
 - description of 26
 - for policy environment 218
- incremental backup
 - file eligibility for 212
 - full 212
 - partial 213
 - specifying frequency 246
- initial start date, description of 250

- initial start time, description of 250
- initialize tape volume 181
- insert category, automated libraries 84
- interface, application programming
 - compression option 315
 - deletion option 316
 - description of 3
 - registering to server 315
- INZTAP command 181

L

- labels
 - checking media 86
 - INZTAP command 181
 - overwriting existing labels 182
 - sequential storage pools 181
- library
 - 349X 84, 86
 - auditing volume inventory 88
 - automated 83
 - configuration example 69, 73
 - defining 110
 - deleting 112
 - managing 107, 110
 - manual 19, 69, 80
 - querying 111
 - type 18
 - updating 111
- license
 - compliance 298
 - example 298
 - features
 - for additional clients 296
 - for clients other than AIX 297
 - for device module support 297, 298
 - monitoring 298
 - using 295
- LOCK ADMIN command 307
- LOCK NODE command 311
- log mode
 - normal 350, 352
 - roll-forward 350, 352
 - setting 350
- logical devices 10, 56
- LOGPOOLSIZE option 291

M

- macro file 64, 68
- magnetic disk devices 9, 55
- management class
 - assigning a default 229
 - associating a file with 209
 - binding a file to 209
 - controlling user access 207
 - copying 217, 222
 - default 208
 - defining 222
 - deleting 235
 - description of 205, 206, 207
 - rebinding a file 211
 - updating 217, 222
- management class configuration 207
- MANUAL libraries 19, 69
- maximum extension 283
- media labels, checking 86
- media loss, recovery from 369
- messages
 - directing import messages to an output file 332
 - for automated libraries 80
 - mount, queue setup 80
 - mount, using the administrative client 80
 - recovery from ANR8263W 127
- migrating a file 204, 215
- migration
 - automatic, for HSM client
 - demand 205
 - threshold 205
 - canceling the server process 170
 - defining threshold for disk storage pool 141
 - defining threshold for tape storage pool 142
 - description, server process 139
 - monitoring thresholds for storage pools 167
 - premigration for HSM client 205
 - providing additional space for server process 170
 - providing users with immediate access to files on disk 141
 - reconciliation 205
 - selective, for HSM client 205
 - stub file on HSM client 205
 - threshold for a storage pool
 - high 139
 - low 139
- mirrored volume
 - description of 348
 - querying 348

- mirrored volume *(continued)*
 - viewing information about 348
- mirroring
 - advantages 347
 - database 347
 - defining volumes 348
 - description of 27
 - recovery log 342, 347, 348
 - recovery procedure 358
- mode
 - description of 224
 - scheduling 246
- modified mode, description of 224
- mount
 - library 121
 - limit, for tapes 116
 - query 104
 - retention period 117
 - wait period 117
- mount exit program 373
- mount mode 80
- mount operations 101
- MOVE DATA command 194
- moving data
 - example 196
 - from offsite volume in a copy storage pool 195
 - monitoring the movement of 197
 - procedures 195
 - requesting processing information 196
 - to another storage pool 194
 - to other volumes in same storage pool 194

O

- occupancy, querying 172
- offsite volumes, moving data in a copy storage pool 195
- one-drive library, manual volume reclamation 59, 153
- open registration
 - description of 309
 - setting 308
- operator privilege class
 - description of 303
 - granting 303
 - revoking 305
- option, server
 - BUFPOOLSIZE 290
 - CSLMSGQ 323
 - DEVCFGFILE 356
 - LOGPOOLSIZE 292

- option, server *(continued)*
 - MAXSESSIONS 280
 - MNTMSGQ 70, 75
 - VOLHSTFILE 354
- options, querying
 - BUFPOOLSIZE 291
 - LOGPOOLSIZE 291
 - VIRTUALMOUNTPOINT 313
- options, server
 - See server option

P

- page, description of 290
- password
 - resetting an administrative 300
 - setting authentication for a client 299
 - setting expiration 299
- performance
 - cache, improved retrievability of files 57
 - concurrent client/server operation considerations 247
 - database or recovery log, optimizing 290
 - database read, increase with mirroring 347
 - volume frequently used, improve with longer mount retention 117
 - workstation, compression option considerations 309
- period, specifying for an incremental backup 246
- policies, managing ADSM 203
- policy definitions, importing 332
- policy domain
 - creating 220
 - deleting 235
 - description of 205, 206
 - updating 217, 219
- policy objects
 - deleting 234
 - description of 205
 - querying 231
- policy operations 204
- policy privilege class
 - changing administrative authority 305
 - description of 302
 - granting 302
- policy set
 - activating 229
 - copying 217, 221
 - defining 221
 - deleting 235
 - description of 205, 206

- policy set (*continued*)
 - updating 221
 - validating 229, 231
- policy, storage management
 - description of 26, 205
 - managing 203
 - tailoring 217
 - using standard 216
- pool, storage
 - amount of space used 173
 - auditing a volume 189
 - backup and recovery 345
 - copy 132
 - creating a hierarchy 135
 - defining 159
 - defining for disk 161
 - defining for tape 161
 - deleting 175
 - description of 132
 - determining access mode 159, 162
 - determining maximum file size 159
 - determining whether to use collocation 144, 160, 163
 - enabling cache for disk 143, 160
 - estimating space for archived files on disk 158
 - estimating space for backed up files on disk 157
 - estimating space for disk 157
 - estimating space for sequential 158
 - estimating space in multiple 135
 - managing 131
 - monitoring 166
 - moving files 194
 - moving files between 194
 - overview 12
 - primary 132
 - querying 166
 - random access 132
 - recovery log, effect on 277
 - restore 346, 367
 - sequential access 132
 - updating 159
 - updating for disk 161
 - using cache on disk 143, 160
 - viewing information about 166
- premigration 205
- PREVIEW parameter 318, 329
- private status of volumes 81, 85
- privilege class, administrator
 - analyst 304
 - description of 300

- privilege class, administrator (*continued*)
 - granting authority 300
 - operator 303
 - policy 301, 302
 - revoking all 305
 - storage 302, 303
 - system 301
- privilege class, policy
 - changing administrative authority 305
 - description of 302
 - granting 302
- process canceling 269
- programming interface notice xv
- protecting your data 341
- publications xx

Q

- QUERY ACTLOG command 273, 323
- QUERY ADMIN command 307
- QUERY ASSOCIATION command 258
- QUERY CONTENT command 187
- QUERY COPYGROUP command 232, 334
- QUERY DB command 287, 290
- QUERY DBBACKUPTRIGGER command 353
- QUERY DBVOLUME command 286, 348
- QUERY DEVCLASS command 319
- QUERY DOMAIN command 233
- QUERY EVENT command 255
- QUERY EXIT command 96
- QUERY FILESPACE command 313
- QUERY LICENSE command 299
- QUERY LOG command 292
- QUERY LOGVOLUME command 286, 348
- QUERY MGMTCLASS command 232
- QUERY NODE command 311
- QUERY OCCUPANCY 173, 174
- QUERY OCCUPANCY command 172
- QUERY OPTION command 271
- QUERY POLICYSET command 233
- QUERY PROCESS command 170, 196, 269, 320
- QUERY SCHEDULE command 243
- QUERY SESSION command 265
- QUERY STATUS 270
- QUERY STGPOOL command 166, 167, 171
- QUERY VOLHISTORY command 355
- QUERY VOLUME command 185, 197
- querying for general information 185
- querying policy objects 231

querying storage volumes 185

R

randomize, description of 247

read-only access mode 180

read/write access mode 180

rebinding

description of 211

file to a management class 211

recalling a file

selective 205

transparent 205

reclamation

affect of collocation on 153

delaying reuse of volumes 153, 154

description of 25

offsite volume 152

setting a threshold for sequential storage pool 149, 160, 163

threshold 25

with single drive 83, 153

reclamation threshold, setting for sequential storage

pool 149, 160, 163

recovering storage pools 345

recovering the database 358

recovery from disaster

See disaster recovery

recovery log

adding space to (CRTVOLADSM) 282

available space 279, 280

buffer pool 293

consistent database image 277

defining a volume 283

defining mirrored volumes 347

deleting a volume 288

deleting space 286

description of 26, 277

determining how much space is allocated 278, 280

estimating the amount of space needed 280

logical volume 278, 280

managing 277

mirroring 342, 347

monitoring space 278, 280

monitoring the buffer pool 293

optimizing performance 290

querying the buffer pool 292

reducing capacity 288

size of 351

storage pool size effect 277

recovery log (*continued*)

viewing information about 292

volume placement 283

when to backup 341, 347, 350

recovery log mode

normal 350, 352

roll-forward 350, 352

setting 350

recovery, disaster

auditing storage pool volumes 364

example recovery procedures 365

general strategy 341

methods 341

providing 341

when to backup 341, 350

REDUCE DB command 288

REDUCE LOG command 288

REEL device type 115

REGISTER ADMIN command 300

REGISTER LICENSE command 298

REGISTER NODE command 309

registering a workstation 315

registration

closed 309

description of 308

managing client node 308

managing for a client node 295

managing for an administrator 295

open 309

setting for a client node 308

REMOVE ADMIN command 306

REMOVE NODE command 315

RENAME ADMIN command 306

RENAME FILESPACE command 337

RENAME NODE command 311

renaming an administrator ID 306

RESET BUFPOOL command 290

RESET DBMAXUTILIZATION command 279, 280

RESET LOGCONSUMPTION command 351

RESET LOGMAXUTILIZATION command 279, 280

restarting the server 265

RESTORE STGPOOL command 346, 367

RESTORE VOLUME command 369

restoring a file 204

restoring the database

point-in-time 358

to its most current state 362

restricted policy privilege

changing administrative authority 304

granting 302

- restricted storage privilege
 - changing administrative authority 304
 - granting 303
 - retain extra versions, description of 227
 - retain only version, description of 227
 - retention grace period
 - description of archive 220
 - description of backup 220
 - using archive 220
 - using backup 220
 - retrieving a file 204
 - reuse of sequential volumes
 - delaying 153, 154
 - storage pool volumes 82
 - REVOKE AUTHORITY command 304
 - roll-forward recovery
 - database backup trigger 363
 - mirroring recovery log 363
 - recovery log 363
- S**
- schedule
 - associating client node 242
 - client options to use 253
 - coordinating 244
 - copying 254
 - day of the week 250
 - defining 250
 - deleting 255
 - description of 239
 - expiration date 251
 - files to process 252
 - frequency of service 251
 - initial start date 250
 - initial time 250
 - managing associations 250
 - missed, querying 244, 256
 - priority 251
 - querying 243
 - results of 255
 - startup window 246, 250
 - type of action 253
 - updating 250
 - viewing information about 243
 - schedule event
 - managing 255
 - querying 255
 - viewing information about 255
 - scheduled operations, setting the maximum 247
 - scheduler workload, controlling 246
 - scheduling mode
 - client-polling 245
 - description of 239
 - selecting 246
 - server-prompted 245
 - setting on a client node 246
 - setting on the server 246
 - scheduling, central
 - controlling the workload 246
 - coordinating 244
 - description of 26, 239
 - scratch status, 349X library 85
 - scratch volume
 - deleting 135, 355
 - description 81
 - FILE volumes 57
 - number allowed in a storage pool 160, 162
 - using in storage pools 135
 - secondary server attachment, registering 298
 - selective backup 204, 214
 - selective recall 205
 - sequential storage pool
 - auditing a single volume in 193
 - auditing multiple volumes in 192
 - collocation 149
 - estimating space 158
 - migration threshold 142
 - reclamation 149
 - server
 - canceling process 269
 - description of 3
 - disabling 267
 - disabling access 267
 - enabling 268
 - enabling access 267
 - halting 264
 - managing operation 263
 - managing processes 268
 - querying about processes 269
 - querying options 271
 - querying status 270
 - restarting 265
 - scheduling mode 245
 - setting the name 271
 - starting 263
 - stopping 264
 - viewing information about 270
 - viewing information about processes 269

- server option
 - BUFPOOLSIZE 290
 - CSLMSGQ 323
 - DEVCFGFILE 356
 - LOGPOOLSIZE 292
 - MAXSESSIONS 280
 - MNTMSGQ 70, 75
 - VOLHSTFILE 354
- server storage
 - client files, process for storing 4
 - concepts overview 4
 - considering user needs for recovering 17
 - deleting files from 197
 - evaluating 16
 - managing 4
 - monitoring 189
 - planning 16
 - tailoring definitions 333
 - using disk devices 55
 - using media management interface 89
 - using tape devices 59
- server-prompted scheduling 245
- session
 - canceling 267
 - setting the maximum percentage for scheduled operations 247
- SET ACCOUNTING command 274
- SET ACTLOGRETENTION command 273
- SET AUTHENTICATION command 299
- SET EVENTRETENTION command 257
- SET LICENSEAUDITPERIOD command 299
- SET LOGMODE command 354
- SET MAXCMDRETRIES command 249
- SET MAXSCHEDESESSIONS command 247
- SET PASSEXP command 299
- SET QUERYSCHEDPERIOD command 249
- SET RANDOMIZE command 247
- SET REGISTRATION command 308
- SET RETRYPERIOD command 249
- SET SCHEDMODES command 246
- SET SERVERNAME command 271
- setting a password 27, 299
- setting compression 309
- shared dynamic serialization, description of 223, 228
- shared static serialization, description of 223, 228
- single drive library, manual volume reclamation 59, 153
- space
 - adding to the database or recovery log 282
 - deleting from the database or recovery log 286
 - estimating database and recovery log requirements 280
- space management
 - See hierarchical storage management
- space-managed file 204
- standard management class, copying 222
- standard storage management policies, using 216
- start time, randomizing for a schedule 247
- starting the server 263
- startup window, description of 247
- static serialization, description of 223, 228
- status codes, volume 85
- stopping the server 264
- storage hierarchy
 - defining in reverse order 161
 - establishing 135
- storage management policy
 - description of 26, 205
 - managing 203
 - tailoring 217
 - using standard 216
- storage occupancy, querying 172
- storage pool
 - amount of space used 173
 - auditing a volume 189
 - backup and recovery 345
 - copy 132
 - creating a hierarchy 135
 - defining 159
 - defining for disk 161
 - defining for tape 161
 - deleting 175
 - description of 132
 - determining access mode 159, 162
 - determining maximum file size 159
 - determining whether to use collocation 144, 160, 163
 - enabling cache for disk 143, 160
 - estimating space for archived files on disk 158
 - estimating space for backed up files on disk 157
 - estimating space for disk 157
 - estimating space for sequential 158
 - estimating space in multiple 135
 - managing 131
 - monitoring 166
 - moving files 194
 - moving files between 194
 - overview 12
 - primary 132
 - querying 166
 - random access 132
 - recovery log, effect on 277

- storage pool (*continued*)
 - restore 346, 367
 - sequential access 132
 - updating 159
 - updating for disk 161
 - using cache on disk 143, 160
 - viewing information about 166
- storage pool backup
 - full 345
 - incremental 345
- storage privilege class
 - changing administrative authority 305
 - description of 303
 - granting 303
- storage volume
 - auditing 189
 - contents 187
 - information about 185
 - labeling sequential access 181
 - managing 179
 - monitoring use 185
 - overview 12
 - preparing random access 181
 - preparing sequential access 181
 - quarter-inch tape cartridge 179
- STRSTADSM command 359, 362
- STRSVRADSM command 264
- stub file 205
- swapping volumes 87
- system privilege class
 - changing administrative authority 305
 - description of 301
 - granting 301

T

- tape
 - exporting data 324
 - label prefix 118
 - planning for exporting data 319
 - recording format 118
 - reuse in storage pools 82
 - scratch, determining use 135, 160, 162
 - setting mount retention period 117
- tape management system, using
 - about the exits 89
 - creating exits 90
 - exit program, defining 92, 96
 - exit program, querying 96
 - exit program, updating 96

- tape management system, using (*continued*)
 - messages 97
 - notes on operations 97
 - setting up 90
- trademarks xvii
- transactions
 - database 277, 278
 - how ADSM processes 278
- transparent recall 205
- type, device
 - 3590 115, 179
 - 8MM 115, 179
 - CARTRIDGE 115, 179
 - DISK 115, 179
 - FILE 115, 179
 - REEL 115, 179

U

- unavailable access mode 180
- UNLOCK ADMIN command 307
- UNLOCK NODE command 311
- unplanned shutdown 264
- unrestricted policy privilege
 - changing administrative privilege 304
 - granting 301
- unrestricted storage privilege
 - changing administrative authority 304
 - granting 302
- unusable space for database and recovery log 279
- UPDATE ADMIN command 300
- UPDATE COPYGROUP command 223, 227
- UPDATE DBBACKUPTRIGGER command 353
- UPDATE DOMAIN command 220
- UPDATE EXIT command 96
- UPDATE MGMTCLASS command 222
- UPDATE NODE command 310
- UPDATE POLICYSET command 221
- UPDATE SCHEDULE command 250
- UPDATE STGPOOL command 72
- UPDATE VOLUME command 183
- usable space 279
- user documentation xx
- USRDFN libraries
 - about 20
 - defining 110
 - updating 112
- utility, device configuration 62
- utilization
 - description of 279

utilization (*continued*)
 monitoring 279, 280

V

VALIDATE POLICYSET command 229
VARY command 270
varying volumes on or off line 270
versions data deleted, description of 226
versions data exists, description of 224
VFYSVRADSM command 264
VIRTUALMOUNTPOINT option 313
VOLHSTFILE option 354
volume
 access, controlling 82
 allocating space for disk 181
 auditing 88, 189
 auditing considerations 190
 capacity, compression effect 125
 defining for database 283
 defining for recovery log 283
 defining to storage pools 183
 deleting 197, 198, 355
 detailed report 189
 determining which are mounted 104
 disk storage 183
 disk storage pool 191
 dismounting 105
 inventory maintenance 81
 managing 83
 monitoring movement of data 197
 monitoring use 185
 mount retention period 117
 moving files between 193
 new 86
 preparing for storage pool 179
 private 85
 querying 185
 querying contents 187
 querying for general information 185
 random access storage pools 132, 135
 recovery using mirroring 358
 removing 87
 returning 88
 reuse delay 153, 154
 scratch category 85
 scratch, using 135
 sequential 183
 sequential storage pools 181
 setting access mode 180

volume (*continued*)
 standard report 188
 status codes 85
 swapping 87
 updating 87
 updating to storage pools 183
 varying 270
volume capacity 121
volume copy
 allocating to separate disks 347
 description of 347
volume history backup
 and point-in-time 354
volume history files, establishing 354

W

workstation, registering 315
WRKDEVADSM command 62

Communicating Your Comments to IBM

ADSTAR Distributed Storage Manager
for AS/400
Administrator's Guide
Version 2
Publication No. SC35-0196-00

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 AS/400
Administrator's Guide
Version 2**

Publication No. SC35-0196-00

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

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
SC35-0196-00



Cut or Fold
Along Line

Fold and Tape

Please do not staple

Fold and Tape



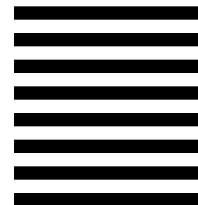
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

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

SC35-0196-00

Cut or Fold
Along Line



Program Number: 5763-SV2



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC35-0196-00



Spine information:



ADSTAR Distributed Storage Manager
for AS/400

Administrator's Guide

Version 2