ADSTAR Distributed Storage Manager
for AIX

# Administrator's Guide

*Version 2*

SH35-0134-01

**Note!**

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

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

| **Second Edition (March 1996)**

This edition applies to Version 2 of the ADSTAR Distributed Storage Manager for AIX (5765-564) 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. The changes made to this manual are indicated by a vertical line (|) in the left margin.

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.

# Contents

**Notices** . . . . . xiii
Programming Interface . . . . . xiii
Trademarks . . . . . xv

**Preface** . . . . . xvii
Who Should Read This Publication . . . . . xvii
What You Should Know Before Reading This Publication . . . . . xvii
ADSTAR Distributed Storage Manager Publications . . . . . xvii
Related AIX System Publications . . . . . xix
| Related Hardware Products Publications . . . . . xix
| IBM ITSC Publications (Redbooks) . . . . . xx
Software Developer's Program . . . . . xx
Do You Have Comments or Suggestions? . . . . . xxi
Translations . . . . . xxi

| **Summary of Changes for ADSTAR Distributed Storage Manager** . . . . . xxiii
| Changes to This Book for Version 2—March 1996 . . . . . xxiii
| Changes for Version 2—December 1995 . . . . . xxiii
Changes for Version 2—July 1995 . . . . . xxiii

**Chapter 1. Introducing ADSTAR Distributed Storage Manager** . . . . . 1
Administrator Tasks . . . . . 2
  Using the ADSM Interfaces . . . . . 3
  Managing Server Operations . . . . . 3
  Managing the Database and Recovery Log . . . . . 3
  Managing Licensing, Privilege Classes, and Registration . . . . . 3
  Managing Policies . . . . . 4
  Scheduling Operations . . . . . 4
  Managing Drives and Libraries . . . . . 4
  Managing Storage Devices . . . . . 4
  Managing Storage Pools . . . . . 5
  Managing Storage Pool Volumes . . . . . 5
  Exporting and Importing Data . . . . . 5
  Recovering Data . . . . . 6
  Using Disaster Recovery Manager . . . . . 6

**Chapter 2. Using ADSTAR Distributed Storage Manager Interfaces** . . . . . 7
Using the Graphical User Interface . . . . . 8
  Getting Started . . . . . 8
  Using Online Help . . . . . 20
Using the Administrative Command-Line Interface . . . . . 27
  Using Console Mode . . . . . 27
  Using Mount Mode . . . . . 28
  Using Batch Mode . . . . . 28
  Using Interactive Mode . . . . . 29

Using Administrative Client Options 29
Using Macros to Issue Commands 29
Using the Server Console Session 30
Application Programming Interface 31
Task, GUI, and Command Cross-Reference 32
Tasks Found in “Using ADSTAR Distributed Storage Manager Interfaces” 33
Tasks Found in “Managing Server Operations” 34
Tasks Found in “Managing the Database and Recovery Log” 36
Tasks Found in “Managing Licensing, Privilege Classes, and Registration” 38
Tasks Found in “Managing Policies” 41
Tasks Found in “Scheduling Operations” 44
Tasks Found in “Managing Drives and Libraries” 46
Tasks Found in “Managing Storage Devices” 48
Tasks Found in “Managing Storage Pools” 49
Tasks Found in “Managing Storage Pool Volumes” 50
Tasks Found in “Exporting and Importing Data” 52
Tasks Found in “Recovering Data” 54
Tasks Found in “Using Disaster Recovery Manager” 56

**Chapter 3. Managing Server Operations** 61
Starting, Halting, and Restarting the Server 61
Starting the Server 61
Halting the Server 62
Restarting the Server 63
Freeing Links for SNA LU6.2 Client Connections 63
Managing Client Sessions 64
Requesting Information about Client Sessions 64
Canceling a Client Session 65
Disabling or Enabling Server Access 66
Managing Server Processes 66
Requesting Information about Server Processes 67
Canceling Server Processes 68
Varying Disk Volumes Online or Offline 68
Requesting Information about Server Status 68
Setting the Server Name 69
Querying Server Options 70
Managing the Activity Log 70
Changing the Size of the Activity Log 70
Setting the Activity Log Retention Period 71
Requesting Information from the Activity Log 71
Monitoring Accounting Records 72
Getting Help on Commands and Error Messages 74

**Chapter 4. Managing the Database and Recovery Log** 75
Database and Recovery Log 75
Database Information 76
Recovery Log Information 76
Relationship between the Database and the Recovery Log 76

Storage Pool Volume Locations . . . . . . . . . . . . . . . . . . . . . . . . 77
Transaction Processing . . . . . . . . . . . . . . . . . . . . . . . . . . . . 77
Allocating Space for the Database and Recovery Log . . . . . . . . . . . . . . . 78
Step 1: Estimating the Amount of the Database Space . . . . . . . . . . . . 78
Step 2: Estimating the Amount of Recovery Log Space . . . . . . . . . . . . 79
Step 3: Understanding How Space is Managed by the Server . . . . . . . . . 79
Step 4: Allocating Space for the Database and Recovery Log . . . . . . . . . 83
Adding Space to the Database or Recovery Log . . . . . . . . . . . . . . . . . 85
Step 1: Defining Disk Volumes . . . . . . . . . . . . . . . . . . . . . . . . 85
Step 2: Determining the Maximum Extension . . . . . . . . . . . . . . . . . 86
Step 3: Extending the Capacity of the Database or Recovery Log . . . . . . . 87
Deleting Space from the Database or Recovery Log . . . . . . . . . . . . . . . 89
Step 1: Determining the Size of Volumes . . . . . . . . . . . . . . . . . . . 89
Step 2: Determining If There Is Sufficient Free Space . . . . . . . . . . . . 91
Step 3: Reducing the Capacity of the Database or Recovery Log . . . . . . . 92
Step 4: Deleting a Volume from the Database or Recovery Log . . . . . . . . 93
Optimizing the Performance of the Database or Recovery Log . . . . . . . . . . 94
Database Buffer Pool . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 94
Recovery Log Buffer Pool . . . . . . . . . . . . . . . . . . . . . . . . . . . 94
Monitoring the Database Buffer Pool . . . . . . . . . . . . . . . . . . . . . 95
Monitoring the Recovery Log Buffer Pool . . . . . . . . . . . . . . . . . . . 96

**Chapter 5. Managing Licensing, Privilege Classes, and Registration . . . . 99**
Managing ADSM Licenses . . . . . . . . . . . . . . . . . . . . . . . . . . . . 99
Licensed Features . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 100
License Compliance . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 103
Licensing Example . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 103
Monitoring Licenses . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 104
Trial License . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 104
Ensuring Client/Server Authentication . . . . . . . . . . . . . . . . . . . . . 104
Setting Client Password Authentication . . . . . . . . . . . . . . . . . . . 104
Setting User Password Expiration . . . . . . . . . . . . . . . . . . . . . . 105
Registering Administrators or Updating Information . . . . . . . . . . . . . . . 105
Granting Administrative Authority . . . . . . . . . . . . . . . . . . . . . . . 106
System Privilege . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 107
Unrestricted Policy Privilege . . . . . . . . . . . . . . . . . . . . . . . . . 108
Restricted Policy Privilege . . . . . . . . . . . . . . . . . . . . . . . . . . 108
Unrestricted Storage Privilege . . . . . . . . . . . . . . . . . . . . . . . . 109
Restricted Storage Privilege . . . . . . . . . . . . . . . . . . . . . . . . . 109
Operator Privilege . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 110
Analyst Privilege . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 110
Revoking or Reducing Administrative Authority . . . . . . . . . . . . . . . . . 110
Revoking All Administrative Privilege Classes . . . . . . . . . . . . . . . . 111
Revoking One or More Administrative Privilege Classes . . . . . . . . . . . 111
Reducing System Authority . . . . . . . . . . . . . . . . . . . . . . . . . . 111
Reducing Unrestricted Policy to Restricted Policy Privilege . . . . . . . . . 112
Reducing Unrestricted Storage to Restricted Storage Privilege . . . . . . . . 112
Reducing Restricted Policy or Storage Privilege . . . . . . . . . . . . . . . 113

Managing Administrator Access . . . . . . . . . . . . . . . . . . . . . . . . . . 113
Renaming an Administrator . . . . . . . . . . . . . . . . . . . . . . . . . . . 113
Removing Administrators . . . . . . . . . . . . . . . . . . . . . . . . . . . . 114
Locking and Unlocking Administrators from the Server . . . . . . . . . . . . . 114
Managing Client Nodes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 115
Setting Client Node Registration . . . . . . . . . . . . . . . . . . . . . . . . 116
Registering Client Nodes . . . . . . . . . . . . . . . . . . . . . . . . . . . . 118
Managing Client Node Access . . . . . . . . . . . . . . . . . . . . . . . . . . 119
Requesting Information about Client Nodes or File Spaces . . . . . . . . . . 120
Deleting File Spaces and Client Nodes . . . . . . . . . . . . . . . . . . . . . 123
Registering an Application Programming Interface to the Server . . . . . . . . . 124
Understanding How the Compression Option is Set . . . . . . . . . . . . . . . 124
Understanding How the File Deletion Option is Set . . . . . . . . . . . . . . . 125

**Chapter 6. Managing Policies** . . . . . . . . . . . . . . . . . . . . . . . . . . 127
Policy Operations . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 128
Backup and Restore . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 128
Archive and Retrieve . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 128
Migration and Recall . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 128
Policy Objects . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 129
Management Classes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 131
Management Class Configuration . . . . . . . . . . . . . . . . . . . . . . . . 131
Default Management Classes . . . . . . . . . . . . . . . . . . . . . . . . . . 132
The Include-Exclude List . . . . . . . . . . . . . . . . . . . . . . . . . . . . 132
How Files Are Associated with a Management Class . . . . . . . . . . . . . . 133
File Eligibility for Policy Operations . . . . . . . . . . . . . . . . . . . . . . . 134
Incremental Backup . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 135
Selective Backup . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 136
Archive . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 136
Automatic Migration from a Client Node . . . . . . . . . . . . . . . . . . . . 136
How Client Migration Works with Backup and Archive . . . . . . . . . . . . . . 137
Using the Standard Storage Management Policies . . . . . . . . . . . . . . . . 137
Creating Your Own Storage Management Policies . . . . . . . . . . . . . . . . 138
Defining and Updating a Policy Domain . . . . . . . . . . . . . . . . . . . . . 142
Defining and Updating a Policy Set . . . . . . . . . . . . . . . . . . . . . . . 143
Defining and Updating a Management Class . . . . . . . . . . . . . . . . . . 144
Defining and Updating a Backup Copy Group . . . . . . . . . . . . . . . . . . 145
Defining and Updating an Archive Copy Group . . . . . . . . . . . . . . . . . 149
Assigning a Default Management Class . . . . . . . . . . . . . . . . . . . . . 150
Validating and Activating Policy Sets . . . . . . . . . . . . . . . . . . . . . . 150
Activating Policy Sets . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 151
Starting Expiration Processing . . . . . . . . . . . . . . . . . . . . . . . . . 152
Querying Policy Objects . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 152
Querying Copy Groups . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 153
Querying Management Classes . . . . . . . . . . . . . . . . . . . . . . . . . 153
Querying Policy Sets . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 154
Querying Policy Domains . . . . . . . . . . . . . . . . . . . . . . . . . . . . 155
Deleting Policy Objects . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 155

Deleting Copy Groups . . . . . . . . . . . . . . . . . . . . . . . . . . . . 155
Deleting Management Classes . . . . . . . . . . . . . . . . . . . . . . . . 156
Deleting Policy Sets . . . . . . . . . . . . . . . . . . . . . . . . . . . . 156
Deleting Policy Domains . . . . . . . . . . . . . . . . . . . . . . . . . . 157

**Chapter 7. Scheduling Operations** . . . . . . . . . . . . . . . . . . . . 159
Central Scheduling . . . . . . . . . . . . . . . . . . . . . . . . . . . . 159
Scheduling Modes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 160
Client-Polling Scheduling Mode . . . . . . . . . . . . . . . . . . . . . 160
Server-Prompted Scheduling Mode . . . . . . . . . . . . . . . . . . . . 161
Setting Scheduling Modes . . . . . . . . . . . . . . . . . . . . . . . . . 162
Setting the Scheduling Mode on the Server . . . . . . . . . . . . . . . . 162
Setting the Scheduling Mode on Client Nodes . . . . . . . . . . . . . . . 163
Coordinating Server and Client Node Schedules . . . . . . . . . . . . . . 163
Specifying the Schedule Period for Incremental Backup Operations . . . . . . 163
Controlling the Scheduler Workload . . . . . . . . . . . . . . . . . . . . 163
Controlling Contact with the Server . . . . . . . . . . . . . . . . . . . . 165
Managing Schedules and Associations . . . . . . . . . . . . . . . . . . . 167
Defining or Updating Schedules . . . . . . . . . . . . . . . . . . . . . . 167
Copying Schedules . . . . . . . . . . . . . . . . . . . . . . . . . . . . 171
Querying Schedules . . . . . . . . . . . . . . . . . . . . . . . . . . . . 172
Deleting Schedules . . . . . . . . . . . . . . . . . . . . . . . . . . . . 172
Associating Client Nodes with Schedules . . . . . . . . . . . . . . . . . 172
Querying Associations . . . . . . . . . . . . . . . . . . . . . . . . . . 173
Deleting Associations . . . . . . . . . . . . . . . . . . . . . . . . . . 173
Managing Scheduled Event Records . . . . . . . . . . . . . . . . . . . . 174
Querying Event Records . . . . . . . . . . . . . . . . . . . . . . . . . 174
Removing Event Records from the Database . . . . . . . . . . . . . . . . 175

**Chapter 8. Key Storage Management Concepts** . . . . . . . . . . . . . . . 177
How ADSM Stores Client Data . . . . . . . . . . . . . . . . . . . . . . . 178
How ADSM Represents Devices . . . . . . . . . . . . . . . . . . . . . . . 180
Disk Devices . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 180
Tape and Optical Devices . . . . . . . . . . . . . . . . . . . . . . . . . 180
Files as Logical Devices . . . . . . . . . . . . . . . . . . . . . . . . . 181
How ADSM Represents Storage Media . . . . . . . . . . . . . . . . . . . 182
Overview of ADSM Storage Objects . . . . . . . . . . . . . . . . . . . . 182
Storage Pool Volumes . . . . . . . . . . . . . . . . . . . . . . . . . . 183
Storage Pools . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 183
Device Class . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 184
Library . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 184
Drive . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 184
What’s in a Device Class? . . . . . . . . . . . . . . . . . . . . . . . . 184
Device Classes for Random Access Devices . . . . . . . . . . . . . . . . 184
Device Classes for Sequential Access Devices . . . . . . . . . . . . . . . 185
Putting It All Together . . . . . . . . . . . . . . . . . . . . . . . . . . 187
Planning to Configure the ADSM Storage Environment . . . . . . . . . . . 189
Evaluating Your Storage Environment . . . . . . . . . . . . . . . . . . . 189

Mapping Devices to Device Classes . . . 189
Mapping Storage Pools to Device Classes . . . 190
Configuring Devices . . . 191
Disk Devices . . . 191
Tape and Optical Disk Devices . . . 192
Scenario: Configuring an 8mm Automated Tape Library . . . 194
Task 1: Define the Library/MediumChanger (Robot) to the Operating System . 195
Task 2: Define the Drive in the Library/MediumChanger to the Operating System . . . 196
Task 3: Define the Automated Library to ADSM . . . 197
Task 4: Define Policy that Links Client Data with Media for the Automated Library . . . 198
Task 5: Register a Node to the Automated Library Policy Domain . . . 199
Task 6: Label Media for the Automated Library . . . 199
Task 7: Add the Automated Library Volumes . . . 200

**Chapter 9. Managing Drives and Libraries** . . . 201
How ADSM Selects Media and Devices for Data Storage and Retrieval . . . 202
ADSM Library Types . . . 202
MANUAL Libraries . . . 202
SCSI Libraries . . . 203
349X Libraries . . . 203
External Libraries . . . 206
Single Drive Libraries . . . 207
Special File Names for Devices . . . 207
Volume Status: Scratch or Private . . . 208
Scratch Volumes in a Library . . . 208
Private Volumes in a Library . . . 208
Defining and Managing Libraries . . . 209
Defining Libraries . . . 209
Querying Libraries . . . 210
Updating Libraries . . . 210
Deleting Libraries . . . 210
Defining and Managing Drives . . . 211
Defining Drives . . . 211
Querying Drives . . . 212
Updating Drives . . . 212
Deleting Drives . . . 213
Managing Storage Volumes in Automated Libraries . . . 213
Informing the Server about New Volumes . . . 213
Removing Volumes from a Library . . . 215
Changing the Status of a Volume in a Library . . . 216
Auditing a Library’s Volume Inventory . . . 216
Special Considerations for SCSI Libraries . . . 217
Managing Mount Operations . . . 217
Querying Pending Operator Requests . . . 218
Replying to Operator Requests . . . 218
Canceling an Operator Request . . . 218

Determining Which Volumes are Mounted . . . . . . . . . . . . . . . . . . . 218
Dismounting an Idle Volume . . . . . . . . . . . . . . . . . . . . . . . . . 218

**Chapter 10. Managing Storage Devices** . . . . . . . . . . . . . . . . . . . . . 219
Defining Device Classes for Sequential Media . . . . . . . . . . . . . . . . . . . 220
Defining Device Classes for Tape . . . . . . . . . . . . . . . . . . . . . . . 221
| Defining Device Classes for Optical Drives . . . . . . . . . . . . . . . . . . . 230
Defining FILE Device Classes . . . . . . . . . . . . . . . . . . . . . . . . 231
Requesting Information about a Device Class . . . . . . . . . . . . . . . . . . . 233
Updating Device Classes . . . . . . . . . . . . . . . . . . . . . . . . . . . 234
Deleting Device Classes . . . . . . . . . . . . . . . . . . . . . . . . . . . 235

**Chapter 11. Managing Storage Pools** . . . . . . . . . . . . . . . . . . . . . . 237
Storage Pools . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 238
Primary Storage Pools . . . . . . . . . . . . . . . . . . . . . . . . . . . 238
Copy Storage Pools . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 239
Assigning Volumes in Storage Pools . . . . . . . . . . . . . . . . . . . . . . . 244
Assigning Random Access Storage Pool Volumes . . . . . . . . . . . . . . . 244
Assigning Sequential Access Storage Pool Volumes . . . . . . . . . . . . . . 244
Storage Pool Hierarchy . . . . . . . . . . . . . . . . . . . . . . . . . . . . 245
Storage Considerations . . . . . . . . . . . . . . . . . . . . . . . . . . . 246
How ADSM Stores Files in a Storage Pool Hierarchy . . . . . . . . . . . . . 246
Server Migration of Files . . . . . . . . . . . . . . . . . . . . . . . . . . . 247
Migration Thresholds for Disk Storage Pools . . . . . . . . . . . . . . . . . 248
Migration Thresholds for Sequential Access Storage Pools . . . . . . . . . . 250
Damaged Files . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 251
The Use of Cache on Disk Storage Pools . . . . . . . . . . . . . . . . . . . . 252
Considerations for Using Cache . . . . . . . . . . . . . . . . . . . . . . . 252
Considerations for Not Using Cache . . . . . . . . . . . . . . . . . . . . . 252
Collocation on Sequential Access Storage Pools . . . . . . . . . . . . . . . . . 253
How the Server Selects Volumes with Collocation Enabled . . . . . . . . . . 255
How the Server Selects Volumes with Collocation Disabled . . . . . . . . . . 256
Space Reclamation for Sequential Access Storage Pools . . . . . . . . . . . . . 257
Reclamation of Offsite Volumes . . . . . . . . . . . . . . . . . . . . . . . 258
Reclamation Threshold . . . . . . . . . . . . . . . . . . . . . . . . . . . 259
How Collocation Affects Reclamation . . . . . . . . . . . . . . . . . . . . 259
Reclamation in a Single Drive Library . . . . . . . . . . . . . . . . . . . . 260
Estimating Space Needs for Storage Pools . . . . . . . . . . . . . . . . . . . . 260
Estimating Space Needs in Random Access Storage Pools . . . . . . . . . . 260
Estimating Space Needs in Sequential Access Storage Pools . . . . . . . . . 262
Defining or Updating Storage Pools . . . . . . . . . . . . . . . . . . . . . . . 262
Defining a Primary Storage Pool . . . . . . . . . . . . . . . . . . . . . . . 262
Defining a Copy Storage Pool . . . . . . . . . . . . . . . . . . . . . . . . 266
Using Copy Storage Pools to Improve Data Availability . . . . . . . . . . . . . 267
Recreating Damaged Files . . . . . . . . . . . . . . . . . . . . . . . . . . 268
Backing Up Storage Pools . . . . . . . . . . . . . . . . . . . . . . . . . . . 269
Full Storage Pool Backups . . . . . . . . . . . . . . . . . . . . . . . . . . 270
Monitoring the Use of Storage Pools . . . . . . . . . . . . . . . . . . . . . . 271

Monitoring the Use of Storage Pool Space . . . . . . . . . . . . . . . . . . 271
Monitoring Migration Thresholds . . . . . . . . . . . . . . . . . . . . . . . . 273
Monitoring the Use of Cache Space on Disk Storage . . . . . . . . . . . . . 276
Requesting Information on Storage Occupancy . . . . . . . . . . . . . . . . 277
Deleting a Storage Pool . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 280
Restoring Storage Pools . . . . . . . . . . . . . . . . . . . . . . . . . . . . 281

**Chapter 12. Managing Storage Pool Volumes** . . . . . . . . . . . . . . . . 285
Storage Pool Volumes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 285
Access Mode for Storage Pool Volumes . . . . . . . . . . . . . . . . . . . . 286
Allocating Space for Random Access Storage Pool Volumes . . . . . . . . . . 287
Labeling Sequential Storage Pool Volumes . . . . . . . . . . . . . . . . . . 288
Identifying Drives for Labeling Activities . . . . . . . . . . . . . . . . . . 288
Overwriting Existing Volume Labels . . . . . . . . . . . . . . . . . . . . . 288
Labeling Volumes by Using a Library Device . . . . . . . . . . . . . . . . . 288
Volume Labeling Examples . . . . . . . . . . . . . . . . . . . . . . . . . . 289
Defining Storage Pool Volumes . . . . . . . . . . . . . . . . . . . . . . . . 290
Updating Storage Pool Volumes . . . . . . . . . . . . . . . . . . . . . . . . 291
Monitoring the Use of Storage Pool Volumes . . . . . . . . . . . . . . . . . 292
Requesting General Information about Storage Pool Volumes . . . . . . . . . 292
Requesting Detailed Information about Storage Pool Volumes . . . . . . . . . 293
Managing Data Storage . . . . . . . . . . . . . . . . . . . . . . . . . . . . 294
Auditing a Storage Pool Volume . . . . . . . . . . . . . . . . . . . . . . . . 294
Requesting Information about Storage Pool Volume Contents . . . . . . . . . 298
Moving Files from One Volume to Another Volume . . . . . . . . . . . . . . 300
Moving Data to Other Volumes in the Same Storage Pool . . . . . . . . . . . 301
Moving Data to Another Storage Pool . . . . . . . . . . . . . . . . . . . . . 301
Moving Data from an Offsite Volume in a Copy Storage Pool . . . . . . . . . 302
Preparing to Move Data . . . . . . . . . . . . . . . . . . . . . . . . . . . . 302
Requesting Information about the Data Movement Process . . . . . . . . . . 303
Monitoring the Movement of Data between Volumes . . . . . . . . . . . . . . 303
Deleting Data Storage Pool Volumes . . . . . . . . . . . . . . . . . . . . . 304
Deleting a Storage Pool Volume with Data . . . . . . . . . . . . . . . . . . 304
Deleting an Empty Storage Pool Volume . . . . . . . . . . . . . . . . . . . 305
Deleting a Storage Pool Volume That Contains Residual Data . . . . . . . . 305
Restoring Storage Pool Volumes . . . . . . . . . . . . . . . . . . . . . . . . 306

**Chapter 13. Exporting and Importing Data** . . . . . . . . . . . . . . . . . 309
Data That Can Be Exported and Imported . . . . . . . . . . . . . . . . . . . 309
Preparing to Export or Import Data . . . . . . . . . . . . . . . . . . . . . . 310
Using Preview before Exporting or Importing Data . . . . . . . . . . . . . . 310
Planning for Sequential Media Used to Export Data . . . . . . . . . . . . . . 311
Monitoring Export and Import Processes . . . . . . . . . . . . . . . . . . . . 312
Requesting Information about an Export or Import Process . . . . . . . . . . 312
Viewing Information from the Server Console . . . . . . . . . . . . . . . . . 313
Viewing Information from an Administrative Client . . . . . . . . . . . . . . 314
Querying the Activity Log for Export or Import Information . . . . . . . . . . 314
Exporting Data to Sequential Media Volumes . . . . . . . . . . . . . . . . . 315

Deciding When to Export Data . . . . . . . . . . . . . . . . . . . . . . . . 316
Exporting Server Data . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 317
Exporting Administrator Information . . . . . . . . . . . . . . . . . . . . . . . 317
Exporting Client Node Information . . . . . . . . . . . . . . . . . . . . . . . 318
Exporting Policy Information . . . . . . . . . . . . . . . . . . . . . . . . . . 319
Importing Data from Sequential Media Volumes . . . . . . . . . . . . . . . . . . 319
Step 1: Previewing Information before You Import Data . . . . . . . . . . . . 320
Step 2: Importing Definitions . . . . . . . . . . . . . . . . . . . . . . . . . . 322
Step 3: Tailoring Data Storage Definitions on the Target Server . . . . . . . . 324
Step 4: Importing File Data Information . . . . . . . . . . . . . . . . . . . . 325
Considerations When Importing Data . . . . . . . . . . . . . . . . . . . . . . 327
Recovering from Errors during the Import Process . . . . . . . . . . . . . . . 328

**Chapter 14. Recovering Data** . . . . . . . . . . . . . . . . . . . . . . . . . 329
Database and Recovery Log Mirroring . . . . . . . . . . . . . . . . . . . . . . . 330
Using Mirroring . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 331
Recovering by Using Mirrored Copies of the Database . . . . . . . . . . . . . . 333
Database Backup and Recovery . . . . . . . . . . . . . . . . . . . . . . . . . . 334
Roll-Forward Recovery . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 334
Point-in-Time Recovery . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 335
Determining Which Recovery Mode to Run . . . . . . . . . . . . . . . . . . . 335
Determining Which Type of Backup to Run . . . . . . . . . . . . . . . . . . . 336
When to Perform Backups . . . . . . . . . . . . . . . . . . . . . . . . . . . . 336
Database Backup Trigger . . . . . . . . . . . . . . . . . . . . . . . . . . . . 336
Volume History Backup Files . . . . . . . . . . . . . . . . . . . . . . . . . . 337
Device Configuration Backup Files . . . . . . . . . . . . . . . . . . . . . . . 337
Using Database Backup Features . . . . . . . . . . . . . . . . . . . . . . . . . 338
Defining Device Classes for Backup . . . . . . . . . . . . . . . . . . . . . . 338
Establishing Volume History Backup Files . . . . . . . . . . . . . . . . . . . 339
Establishing Device Configuration Backup Files . . . . . . . . . . . . . . . . 340
Performing the Database Backups . . . . . . . . . . . . . . . . . . . . . . . . 342
Setting the Database Backup Trigger . . . . . . . . . . . . . . . . . . . . . . 343
Setting the Recovery Log Mode . . . . . . . . . . . . . . . . . . . . . . . . . 345
Adjusting the Size of the Recovery Log . . . . . . . . . . . . . . . . . . . . 345
Recovering by Using Backed Up Copies of the Database . . . . . . . . . . . . . 346
Restoring a Database by Using Point-in-Time Recovery . . . . . . . . . . . . 346
Restoring a Database to its Most Current State . . . . . . . . . . . . . . . . 348
Storage Pool Backup and Recovery . . . . . . . . . . . . . . . . . . . . . . . . 349
Using Storage Pool Backup Features . . . . . . . . . . . . . . . . . . . . . . . 350
Recovering by Using Backed Up Copies of Storage Pools . . . . . . . . . . . . 351
ADSM Backup and Recovery Scenarios . . . . . . . . . . . . . . . . . . . . . . 351
Backup . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 351
Recovery from a Disaster . . . . . . . . . . . . . . . . . . . . . . . . . . . . 353
Recovery from Media Loss . . . . . . . . . . . . . . . . . . . . . . . . . . . 354
Database Salvage Utilities . . . . . . . . . . . . . . . . . . . . . . . . . . . . 355
Examples of Recovering a Corrupted Database Using DSMSERV Salvage Utilities . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 356

**Chapter 15. Using Disaster Recovery Manager** . . . . . . . . . . . . . . . 359
Comparing Availability Management to Disaster Recovery Management . . . . . 360
Features of Disaster Recovery Manager . . . . . . . . . . . . . . . . . . . . . 360
Automated Generation of a Server Disaster Recovery Plan . . . . . . . . . . 360
Offsite Recovery Media Management . . . . . . . . . . . . . . . . . . . . . 361
Storage of Client Recovery Information . . . . . . . . . . . . . . . . . . . 361
Overview of Set Up for Disaster Recovery Manager . . . . . . . . . . . . . . . 362
Enabling Disaster Recovery Manager . . . . . . . . . . . . . . . . . . . . . . 363
Creating a Backup Copy of Server Primary Storage Pools and Database . . . . 363
Offsite Recovery Media Management . . . . . . . . . . . . . . . . . . . . . . 364
Sending Server Backup Volumes Offsite . . . . . . . . . . . . . . . . . . . 364
Moving Reclaimed or Expired Volumes Back Onsite . . . . . . . . . . . . . 366
Creating the ADSM Server Disaster Recovery Plan File . . . . . . . . . . . . . 368
About the Disaster Recovery Plan File . . . . . . . . . . . . . . . . . . . . 369
About the Recovery Plan File Stanzas . . . . . . . . . . . . . . . . . . . . 369
Example of a Disaster Recovery Plan File . . . . . . . . . . . . . . . . . . 379
Example: Routine Operations Using Disaster Recovery Manager . . . . . . . . 388
Storage of Client Recovery Information . . . . . . . . . . . . . . . . . . . . . 391
Defining Machine Information . . . . . . . . . . . . . . . . . . . . . . . . 391
Defining and Tracking Recovery Media . . . . . . . . . . . . . . . . . . . . 394
Recovering the Server . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 396
Example: Recovering the ADSM Server . . . . . . . . . . . . . . . . . . . 396
Recovering the Clients . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 399
Example: Recovering ADSM Clients . . . . . . . . . . . . . . . . . . . . . 399
Customizing Disaster Recovery Manager . . . . . . . . . . . . . . . . . . . . 402
Customizing the Management of Storage Pools, Path Name Prefixes, and Replacement Volume Identifiers . . . . . . . . . . . . . . . . . . . . . . . . 402
Customizing the Management of Offsite Recovery Media . . . . . . . . . . . 405
Querying the Disaster Recovery Manager System Parameters . . . . . . . . 407
Customizing the Site Specific RECOVERY.INSTRUCTIONS . . . . . . . . . . 407
Using an Awk Script to Break Out a Disaster Recovery Plan File . . . . . . . . 409
Summarized Example of Disaster Recovery Manager Usage . . . . . . . . . . . 410

**Appendix A. External Media Management Interface Description** . . . . . . . 413
Processing during ADSM Server Initialization . . . . . . . . . . . . . . . . . . 413
Processing for Volume Mount, Dismount, and Release Requests . . . . . . . . . 413
Initialization Requests . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 414
Volume Mount Requests . . . . . . . . . . . . . . . . . . . . . . . . . . . . 415
Volume Dismount Requests . . . . . . . . . . . . . . . . . . . . . . . . . . . 417
Volume Release Requests . . . . . . . . . . . . . . . . . . . . . . . . . . . . 417

**Glossary** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 419

**Index** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 437

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

The following terms are trademarks of other companies:

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

C-bus is a trademark of Corollary, Inc.

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.

Windows is a trademark of Microsoft Corporation.

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 to customers in a multivendor computer environment. ADSM provides automated, centrally-scheduled, policy-managed backup, archive, and space management services for file-servers and workstations. The Hierarchical Storage Management (HSM) client is the portion of ADSM that provides space management services.

## Who Should Read This Publication

This reference is intended for anyone who has been assigned an administrative privilege class. While ADSM can be managed by a single administrator, administrative responsibilities can be divided among a number of people as an installation requires.

All the administrator commands and interfaces that 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, refer to *ADSM Licensed Program Specifications*.

For information on installing ADSM, refer to *ADSM Installing the Server and Administrative Client*.

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.

## ADSTAR Distributed Storage Manager Publications

The following table lists ADSM publications.

The ADSM library is also available in softcopy on the IBM Online Library *ADSTAR Distributed Storage Manager Collection* CD-ROM (order number SK2T-1878).

| Short Title | Publication Title | Order Number |
|---|---|---|
| ADSM General Information | *ADSTAR Distributed Storage Manager: General Information* | GH35-0131 |
| ADSM Messages | *ADSTAR Distributed Storage Manager: Messages* | SH35-0133 |

| Short Title | Publication Title | Order Number |
|---|---|---|
| ADSM Licensed Program Specifications | *ADSTAR Distributed Storage Manager for AIX: Licensed Program Specifications* | GH35-0132 |
| ADSM Installing the Server and Administrative Client | *ADSTAR Distributed Storage Manager for AIX: Installing the Server and Administrative Client* | SH35-0136 |
| ADSM Administrator’s Reference | *ADSTAR Distributed Storage Manager for AIX: Administrator’s Reference* | SH35-0135 |
| ADSM Using the UNIX HSM Clients | *ADSTAR Distributed Storage Manager: Using the UNIX Hierarchical Storage Management Clients* | SH26-4030 |
| ADSM V2 Using the Apple Macintosh Backup-Archive Client | *ADSTAR Distributed Storage Manager Version 2: Using the Apple Macintosh Backup-Archive Client* | SH26-4051 |
| ADSM Using the UNIX Backup-Archive Clients | *ADSTAR Distributed Storage Manager Version 2: Using the UNIX Backup-Archive Clients* | SH26-4052 |
| ADSM V2 Using the OS/2 Backup-Archive Client | *ADSTAR Distributed Storage Manager Version 2: Using the OS/2 Backup-Archive Client* | SH26-4053 |
| ADSM V2 Using the DOS Backup-Archive Client | *ADSTAR Distributed Storage Manager Version 2: Using the DOS Backup-Archive Client* | SH26-4054 |
| ADSM V2 Using the Novell NetWare Backup-Archive Client | *ADSTAR Distributed Storage Manager Version 2: Using the Novell NetWare Backup-Archive Client* | SH26-4055 |
| ADSM V2 Using the Microsoft Windows Backup-Archive Clients | *ADSTAR Distributed Storage Manager Version 2: Using the Microsoft Windows Backup-Archive Clients* | SH26-4056 |
| ADSM Using the UNIX Backup-Archive Clients | *ADSTAR Distributed Storage Manager Version 2: Using the UNIX Backup-Archive Clients* | SH26-4052 |
| ADSM Using the Lotus Notes Backup Agent | *ADSTAR Distributed Storage Manager: Using the Lotus Notes Backup Agent* | SH26-4047 |
| ADSM Installing the Clients | *ADSTAR Distributed Storage Manager: Installing the Clients* | SH26-4049 |

## Related AIX System Publications

The following table lists titles and order numbers for related AIX publications.

| Short Title | Publication Title | Order Number |
|---|---|---|
| AIX Installation Instructions | *Installation Instructions for AIX for RISC System/6000* | SC23-2341 |
| AIX General Concepts | *AIX for RISC System/6000 General Concepts and Procedures* | GC23-2202 |
| AIX System User's Guide | *AIX for RISC System/6000 System User's Guide* | GC23-2377 |
| AIX SNA/6000 User's Guide | *AIX SNA Server/6000 User's Guide* | SC31-7002 |
| AIX SNA/6000 Transaction Program Reference | *AIX SNA Server/6000 Transaction Program Reference* | SC31-7003 |
| AIX SNA/6000 Configuration Reference | *AIX SNA Server/6000 Configuration Reference* | SC31-7014 |
| AIX SNA/6000 Command Reference | *AIX SNA Server/6000 Command Reference* | SC31-7100 |
| AIX SNA/6000 Diagnosis Guide | *AIX SNA Server/6000 Diagnosis Guide and Messages* | SC31-7101 |

## Related Hardware Products Publications

The following table lists titles and order numbers for related publications:

| Short Title | Publication Title | Order Number |
|---|---|---|
| IBM 3490 Tape Subsystem User's Guide | *IBM 3490 Magnetic Tape Subsystem Enhanced Capability Models E01 and E11 User's Guide* | GA32-0298 |
| IBM 3494 Operator's Guide | *IBM 3494 Tape Library Dataserver Operator's Guide* | GA32-0280 |
| IBM 3590 Tape Subsystem User's Guide | *IBM 3590 High Performance Tape Subsystem User's Guide* | GA32-0330 |
| IBM 3495 Operator's Guide | *IBM 3495 Tape Library Dataserver Models L20, L30, L40, and L50 Operator's Guide* | GA32-0235 |
| IBM AIX Parallel Channel Tape Attachment/6000 Installation and User's Guide | *IBM AIX Parallel Channel Tape Attachment/6000 Installation and User's Guide* | GA32-0311 |
| IBM SCSI Tape Drive, Medium Changer, and Library Device Drivers | *IBM SCSI Tape Drive, Medium Changer, and Library Device Drivers Installation and User's Guide* | GC35-0154 |

## IBM ITSC Publications (Redbooks)

| Publication Title | Order Number |
|---|---|
| ADSM Version 2 Presentation Guide | SG24-4532 |
| ADSM Implementation Examples | GG24-4034 |
| ADSM Advanced Implementation Examples | GG24-4221 |
| Getting Started with ADSM/2 | GG24-4321 |
| Getting Started with ADSM/6000 | GG24-4421 |
| Getting Started with the NetWare Client | GG24-4242 |
| Getting Started with the AIX/6000 Client | GG24-4243 |
| ADSM API Examples for OS/2 and Windows | SG24-2588 |
| Using ADSM to Back Up Databases | GG24-4335 |
| AIX Storage Management Products Comparison | GG24-4495 |
| Easy Access to Host Data with Distributed File Manager | GG24-4427 |
| ADSM/VSE Implementation | GG24-4266 |
| Setting Up and Implementing ADSM/400 | GG24-4460 |
| AIX Storage Management Products Comparison | GG24-4495 |
| ADSM/6000 on 9076 SP2 | 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 readers' 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 addressess:

- 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 and previous editions of this book.

## Changes to This Book for Version 2—March 1996

The following information has been included in this publication:

**External media management**

ADSM provides an interface that allows external media management systems to operate in conjunction with the ADSM server. To use the interface for one or more devices, you must define a library with library type EXTERNAL. See “External Libraries” on page 206 and Appendix A, “External Media Management Interface Description” on page 413 for details.

## Changes for Version 2—December 1995

The following summarizes changes made for ADSM Version 2 in December 1995. This book is available only in softcopy format.

**Disaster Recovery Manager**

The Disaster Recovery Manager (DRM) feature allows you to prepare for and helps you to recover from disasters that destroy the ADSM server and clients. See Chapter 15, “Using Disaster Recovery Manager” on page 359 for details.

**Configuration and administration enhancements**

You can use the ADSM utilities interface to make ADSM configuration and administration tasks easier. See Chapter 2, “Using ADSTAR Distributed Storage Manager Interfaces” on page 7 for details.

**Device support enhancements**

Device class types now include support for digital linear tape (DLT) drives and libraries, and write-once read-many (WORM) optical drives. See Chapter 10, “Managing Storage Devices” on page 219 for details.

## Changes for Version 2—July 1995

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 14, “Recovering Data” on page 329 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 11, “Managing Storage Pools” on page 237 for details.

**Administrative command scheduling**

You can define schedules for automatically issuing administrative commands once or periodically.

See Chapter 7, “Scheduling Operations” on page 159 for details.

**Hierarchical storage management**

Hierarchical storage management (HSM) provides space management services to HSM clients. HSM clients can automatically migrate user files to storage pools to free up client storage space. A user can access a migrated file as if it were on local storage.

See Chapter 6, “Managing Policies” on page 127 for details.

**Device support enhancements**

Device class types now include support for 4mm tape drives, quarter-inch cartridge (QIC) tape drives, and IBM 3590 tape drives.

Library device support now allows the following:

- On an IBM 3490, 3590E, or 3590 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.

ADSM can operate in conjunction with external media management systems.

See Chapter 10, “Managing Storage Devices” on page 219 for details.

# Chapter 1. Introducing ADSTAR Distributed Storage Manager

ADSTAR Distributed Storage Manager (ADSM) is an enterprise-wide storage management application for the network. It provides automatic 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 data storage, or storage pools. These are groups of random and sequential access media that store files backed up, archived, and migrated from client nodes.

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

**Backup-archive client**
Allows users to register their workstations with a server as client nodes. ADSM users can maintain backup versions of their files, which they can restore if the original files are lost or damaged. Users can also archive files that they do not currently need on their workstations and retrieve the archived files when necessary.

**Hierarchical storage management client**
Provides space management services for workstations. ADSM users can free up workstation storage by migrating less frequently used files to server storage pools. 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 2 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 workstation files on disk, optical, or tape volumes in data storage, which can be grouped into storage pools.

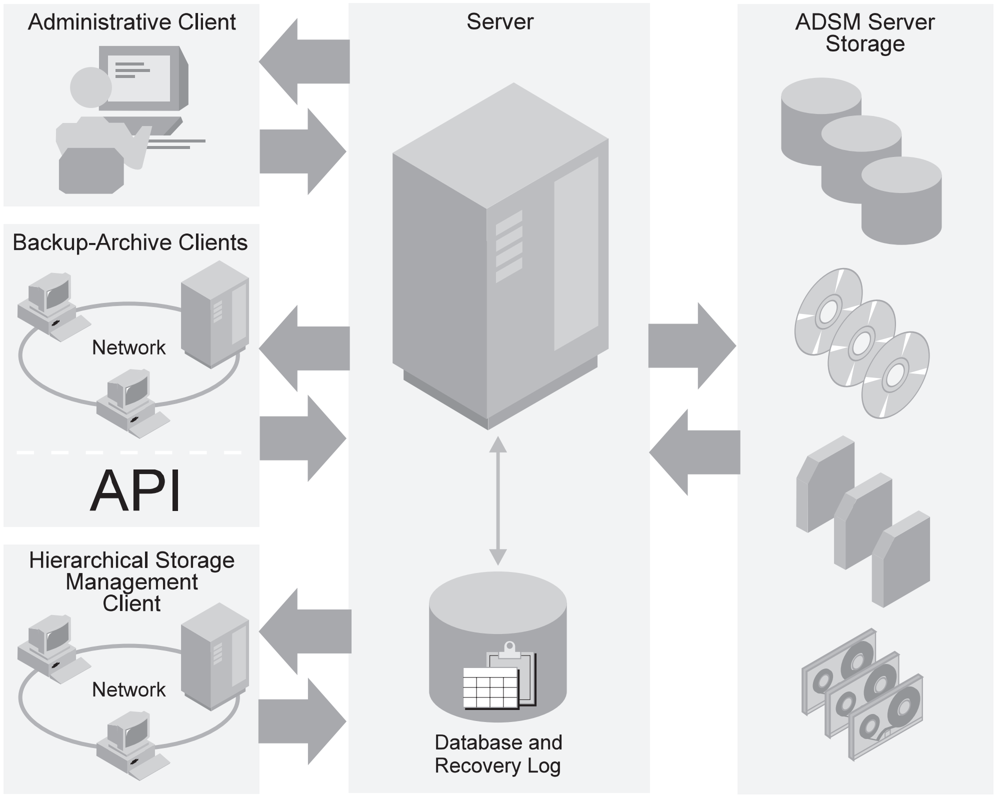


*Figure 1. Sample Client/Server Environment*

## Administrator Tasks

This section 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. This section presents the tasks in the order in which they appear in the chapters of this book:

- Using the ADSM interfaces
- Managing server operations
- Managing the database and recovery log
- Managing licensing, privilege classes, and registration
- Managing ADSM policies
- Scheduling ADSM operations
- Managing drives and libraries
- Managing storage devices
- Managing storage pools
- Managing storage volumes

- Exporting and importing data
- Recovering data
- Using Disaster Recovery Manager

## Using the ADSM Interfaces

There are three interfaces to ADSM:

- Graphical user interface (GUI)
- Command-line interface
- Application programming interface

For information about these interfaces and for tables that relate GUI locations and administrative commands with specific tasks, see Chapter 2, “Using ADSTAR Distributed Storage Manager Interfaces” on page 7.

## 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 3, “Managing Server Operations” on page 61.

## Managing the Database and Recovery Log

The ADSM database contains information about the client data in storage pools, registered client nodes, and ADSM policies. 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 4, “Managing the Database and Recovery Log” on page 75.

## 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 5, “Managing Licensing, Privilege Classes, and Registration” on page 99.

## Managing Policies

From a client node, users can back up or archive files to the server. This process ensures that current data can be restored or retrieved if it is accidentally deleted or corrupted on their workstations. Users can also migrate files to server storage. This process frees up space on their workstations, but the users can recall the files when they are needed.

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 6, “Managing Policies” on page 127.

## Scheduling 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 7, “Scheduling Operations” on page 159.

## Managing Drives and Libraries

A drive is a device used to read and write data on a medium, such as disk or tape. In ADSM a library is a collection of drives that use the same method of mounting a medium, for example, manual or automatic. You are responsible for defining drives and libraries, managing the volumes in automated libraries, and managing mount operations for drives.

For more information about these tasks, see Chapter 9, “Managing Drives and Libraries” on page 201.

## Managing Storage Devices

A device class represents a set of storage devices with similar availability, performance, and storage characteristics. You must define device classes for the drives available to an ADSM server. You specify a device class each time 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 10, “Managing Storage Devices” on page 219.

## Managing Storage Pools

Backed up, archived, or space-managed files are stored in groups of volumes called storage pools. The data on these primary storage pools can be backed to copy storage pools. 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 user 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 user's files on a minimal number of volumes within a storage pool. Because user 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 understanding and defining storage pools and taking advantage of storage pool features, see Chapter 11, "Managing Storage Pools" on page 237.

## Managing Storage Pool Volumes

You manage storage volumes not only by defining, updating, and deleting volumes, but also 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 12, "Managing Storage Pool Volumes" on page 285.

## Exporting and Importing Data

As your storage needs increase, you can move data from one server to another. This process is accomplished by exporting 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 importing data between servers, see Chapter 13, "Exporting and Importing Data" on page 309.

## Recovering Data

ADSM provides a number of ways to recover from media failure or from the a loss of 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
- Offline dump of the database

For more information about preparing for a disaster and for details about recovering from a disaster, see Chapter 14, “Recovering Data” on page 329.

## Using Disaster Recovery Manager

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

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

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

# Chapter 2. Using ADSTAR Distributed Storage Manager Interfaces

Users have three interfaces to ADSM:

- Graphical user interface (GUI)
- Command-line interface
- Application programming interface (API)

This chapter describes those interfaces. It also describes how to use the GUI and the command-line interface.

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

| Section | Page |
|---|---|
| **Tasks:** | |
| Using the graphical user interface (GUI) | 8 |
| Using online help | 20 |
| Using the command-line interface in console mode | 27 |
| Using the command-line interface in mount mode | 28 |
| Using the command-line interface in batch mode | 28 |
| Using the command-line interface in interactive mode | 29 |
| Using administrative client options | 29 |
| Using macros to issue commands | 29 |
| Using the server console session | 30 |

See “Task, GUI, and Command Cross-Reference” on page 32 for task, GUI, and command cross-referencing.

## Using the Graphical User Interface

You can manage the server from a graphical user interface on a workstation. ADSM provides a GUI that can be used to run the ADSM utilities, the ADSM server, the ADSM administrative client, the ADSM backup-archive client, and the HSM client. The ADSM GUI also provides access to the ADSM online books by selecting **Help** from the main window, see Figure 2 on page 9. You may then select **View Books** from the Help menu pulldown for a list of online books. This section contains information about:

- Getting started
- Using online help

## Getting Started

This section contains information about:

- Making selections by using the mouse or the keyboard
- Starting ADSM
- Using the ADSM main window
- Using the utilities
- Starting the administrative client
- Using menu choices
- Completing a task by using the interface
- Closing an administrative client session

### Making Selections by Using the Mouse or the Keyboard

You can make a selection on the GUI by using your mouse or your keyboard:

**Mouse**

Hold down the left mouse button and drag the cursor over the area to be selected.

**Keyboard**

Press the control key and select an object with the space bar.

### Starting ADSM

If the basic installation and configuration was performed using the Motif version of SMIT, the ADSM main window, shown in Figure 2 on page 9, appears when ADSM is started.

There are three ways to start the server:

- To start the server from the ADSM main window:
  1. Double-click on the **ADSM Server** icon.
  2. Select **Begin Session**.
- To start the server from any aixterm window:
  1. Enter the following in any aixterm window:

     ```
     adsm
     ```

  2. From the main window, double-click on the **ADSM Server** icon.

3. Select **Begin Session**.

- To start the server from an AIX command line:

1. Ensure that you are in the directory where your ADSM server is installed. For example, to run the default ADSM server, enter:

```
cd /usr/lpp/adsmserv/bin
```

2. Start the server by entering:

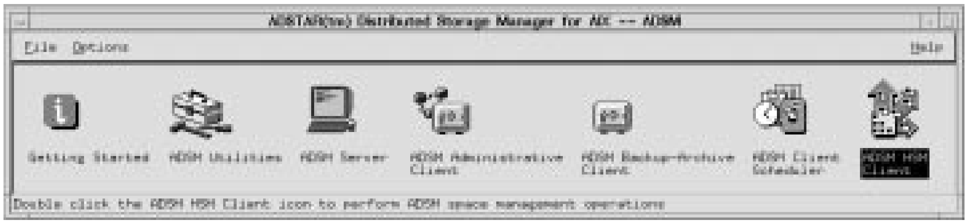
```
dsmserv
```

## Using the Main Window on the GUI

If the basic installation and configuration was performed using the Motif version of SMIT, the ADSM main window shown in Figure 2 appears when ADSM is started.

**OR**

If the basic installation and configuration was performed using a different version of SMIT, the ADSM main window may not be available. If the main window does not appear, proceed to “Starting the Administrative Client” on page 13.


ADSTAR(tm) Distributed Storage Manager for AIX -- ADSM
File
Options
Help
Getting Started
ADSM Utilities
ADSM Server
ADSM Administrative Client
ADSM Backup-Archive Client
ADSM Client Scheduler
ADSM HSM Client

Double click the ADSM HSM Client icon to perform ADSM space management operations

*Figure 2. ADSM Main Window*

The ADSM online books may be accessed by selecting **Help** from the main window. You may then select **View Books** from the Help menu pulldown for a list of online books.

The icons in the ADSM main window are as follows:

**Getting Started**
Offers general information about ADSM and how to use the objects in this window to extend installation, configure devices, configure media, and add additional backup-archive clients.

**ADSM Utilities**
Accesses configuration and administration utilities for the ADSM administrator. See “Using the Utilities” on page 10 for more information.

**ADSM Server**
Accesses the ADSM console. The console in this interface includes a scrollable server message history and the capability to retrieve commands issued at the command line.

**ADSM Administrative Client**

Accesses the ADSM administrative interface. Use the administrative interface to communicate directly with the server to:

- Define ADSM policy
- Define and manage ADSM storage
- Register and manage ADSM clients
- Define and manage ADSM database and recovery log
- Schedule ADSM automatic operations

See "Starting the Administrative Client" on page 13 for more information.

**ADSM Backup-Archive Client**

Accesses the backup-archive interface to:

- Back up and restore files
- Archive and retrieve files
- View backup policies

As part of the basic installation, an ADSM client and ADSM server are installed on the same computer.

**ADSM Client Scheduler**

Allows you to start or stop the automatic scheduler daemon that starts the automatic scheduler.

**ADSM HSM Client**

Enables HSM (hierarchical storage management) clients to migrate to an AIX server when local storage is full. This process frees workstation or file server storage. If a user accesses a migrated file, it is automatically returned from ADSM data storage and placed in the user's local storage.

The ADSM GUI main window also provides access to the ADSM online books from its *Help/View Books* menu item.

## Using the Utilities

Access the utilities by double-clicking on the **ADSM Utilities** icon as shown in Figure 2 on page 9. Figure 3 on page 11 shows the selections available within the utilities.

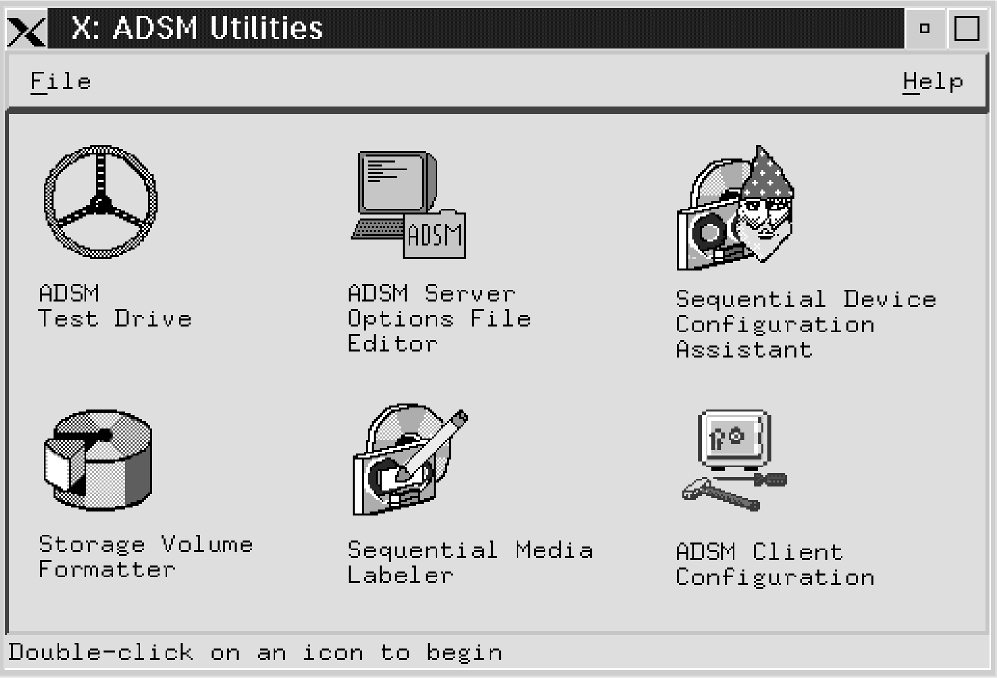

X: ADSM Utilities
File
Help
ADSM Test Drive
ADSM Server Options File Editor
Sequential Device Configuration Assistant
Storage Volume Formatter
Sequential Media Labeler
ADSM Client Configuration
Double-click on an icon to begin

*Figure 3. ADSM Utilities Window*

The icons in the ADSM Utilities are as follows:

**ADSM Test Drive**
Allows you to use the ADSM administrative client and the ADSM backup-archive client to explore using the following ADSM services:

- Backup-archive services
  - Backing up files
  - Restoring files
  - Displaying client policies
- Administrative services
  - Viewing client nodes
  - Viewing administrators
  - Viewing schedules
  - Viewing information on storage volumes
  - Defining and executing a backup schedule

**ADSM Server Options**
Accesses the server options file editor. Use this editor to define server options for the ADSM server.

**ADSM Sequential Device Configuration Assistant**
Configures tape devices and optical devices and defines them to ADSM.

**ADSM Storage Volume Formatter**

Formats volumes for use as ADSM database volumes, recovery log volumes, and storage pool volumes.

**ADSM Sequential Media Labeler**

Labels tape and optical volumes for ADSM use.

**ADSM Client Configuration**

Assists with configuring additional ADSM clients. When ADSM is initially installed, an ADSM server component and an ADSM client component are installed on a single machine.

The administrator can set up a data file and make the utilities available to client machines (for example, by placing it on a file server). Information on this capability is available in the online help for the client configuration utility.

| **Starting the Administrative Client:**

| **1**

| Double-click on the Administrative
| Client icon as shown in the ADSM
| main window.

| **OR**

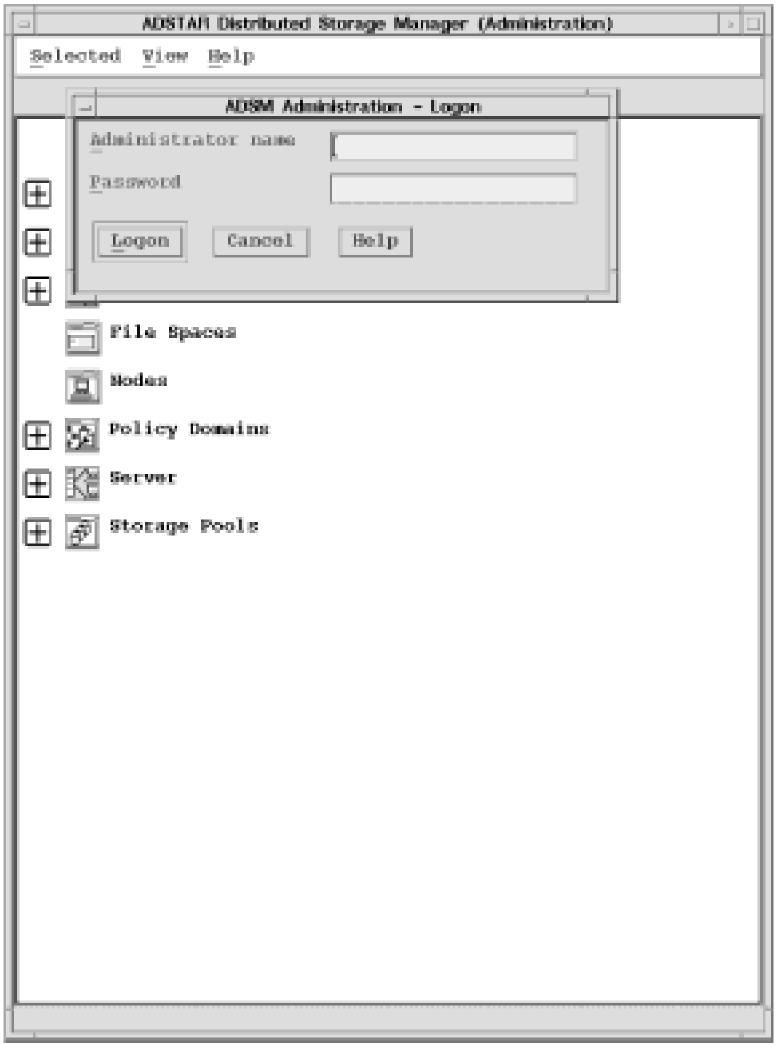
| Enter **dsmadm** in any aixterm
| window.

**2**

Type the administrator name and password in the Logon window and then select **Logon** or press the Enter key.

**Note:** If server authentication is off, you do not need to type the password.

The ADSM administrative icons appear.



Your administrative privilege determines which ADSM tasks you can perform and which objects you will work with most frequently.

| **Administrative Privilege** | **Objects** |
|---|---|
| **Analyst** | Server |
| **Operator** | Server |
| **Policy** | Policy, Nodes, Central Scheduler, and File Spaces |
| **Storage** | Data Storage, Database, and Recovery Log |
| **System** | All |

You can run multiple instances of the interface. Each instance of the interface is a new administrative session. Therefore, you can log on to more than one server from the same administrative client by invoking more than one instance of the graphical user interface.

## Using Menu Choices

In the graphical user interface, each window contains menu bars. Each choice in the menu bar leads to an associated pull-down menu. Choices in the pull-down menus are actions or routing choices that relate to the contents of the window. A routing choice with an ellipsis opens another window. A routing choice with an arrow displays a cascaded menu. Objects in the icon pull-down view of the graphical user interface contain some of the following menu choices:

| **Menu** | **Contents** |
|---|---|
| **Selected** | Contains a number of options, for example:<br>Open as:<br>• Icon<br>• Details<br>• Properties |
| **Edit** | Contains choices that are standard across many different types of objects such as choices that allow you to add new objects or delete existing ones. |
| **View** | Contains choices that affect the way an object is displayed. |
| **Help** | Contains choices that provide access to help information. You can select a help index, general help, instructions about how to use help, task help, or product information. |

**Completing a Task by Using the Interface:** The following example describes how to use the graphical user interface to view and change information about a node. An ADSM administrator with any privilege class can view information about nodes registered with ADSM. In this scenario, you will be displaying information about the **Nodes** object in a domain.

To open the **Icons** view, for example, of the **Nodes** object:

**1**

Select the **Nodes** icon.

**2**

Select **Selected** from the menu bar. Then select **Open as**.

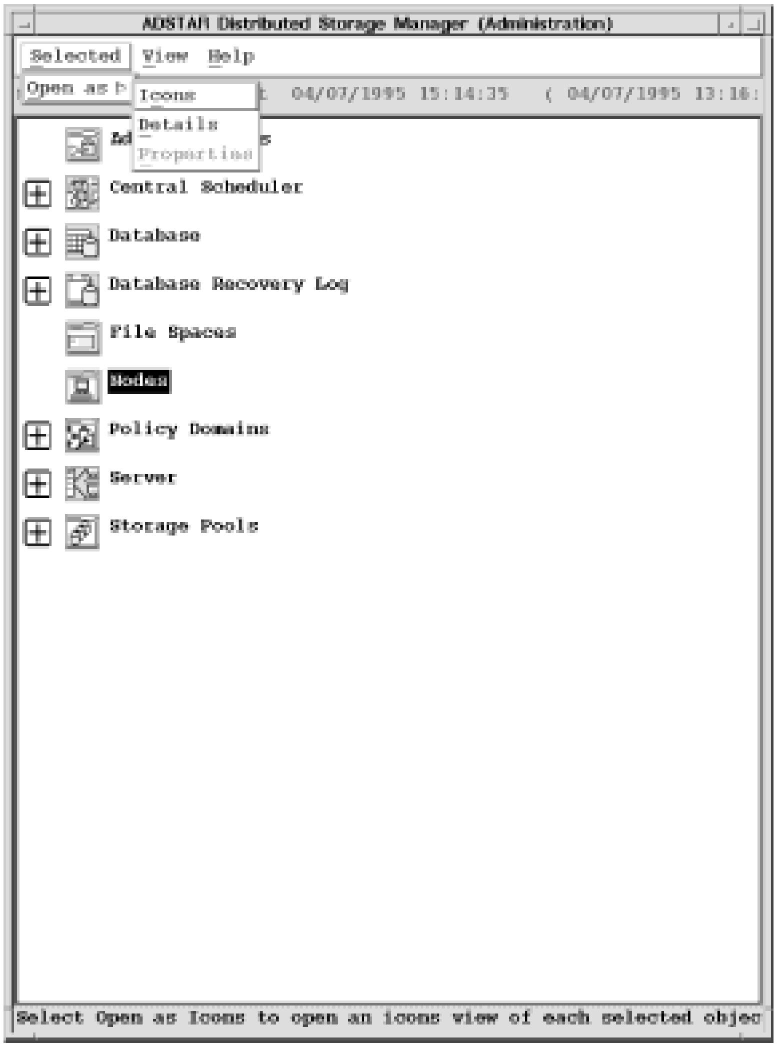
**3**

Select **Icons** from the pull-down menu.

The **Nodes - Icon** screen appears.

**4**

To close the session, see "Closing an Administrative Client Session" on page 19.



ADSTAR Distributed Storage Manager (Administration)

t 04/07/1995 15:14:35 ( 04/07/1995 13:16:

Select Open as Icons to open an icons view of each selected objec

The Details view displays information about the nodes. This information includes the name of the node, its operating system (which the node uses to access the server), the policy domain to which the node is assigned, the number of days that have elapsed since the node last accessed the server, and the number of days that have elapsed since a new password was set for the node. In addition, you can see if the node is locked from accessing the server. Move the horizontal and vertical scroll bars to view all of this information.

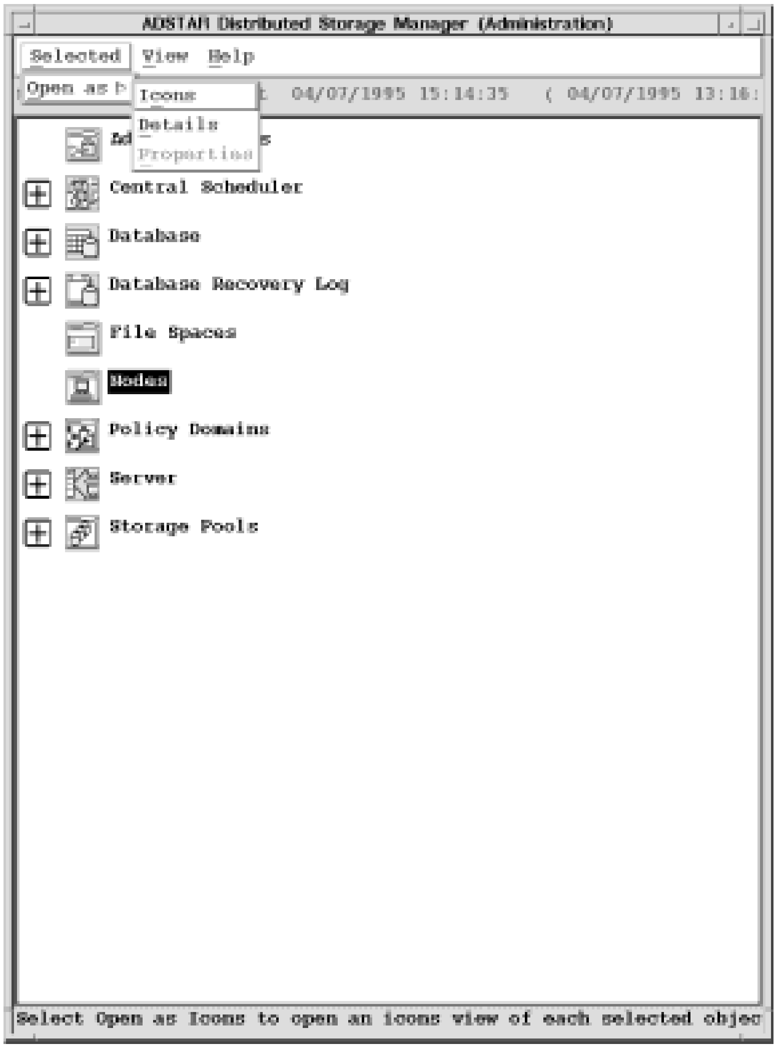
For example, to open the **Details** view window of the **Nodes** object:

## 1

Select the **Nodes** icon.

## 2

Select **Selected** from the menu bar and then select **Open as**.

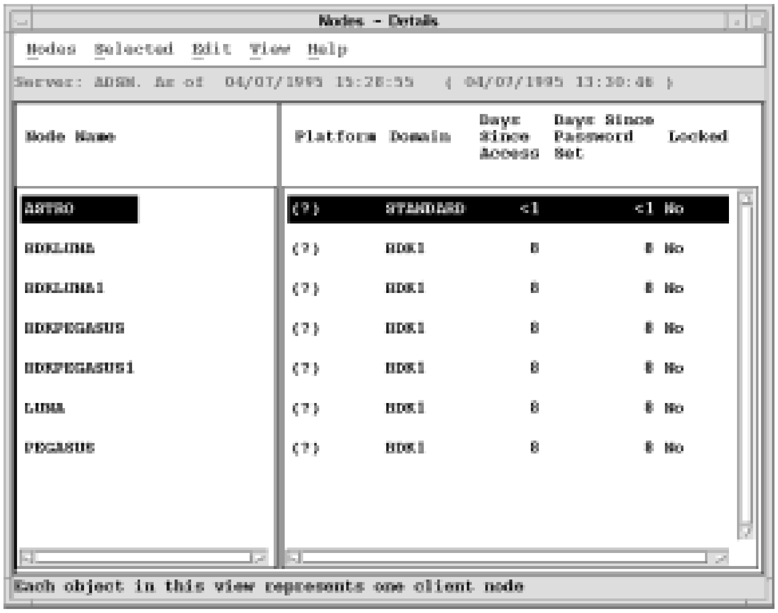


Select Open as Icons to open an icons view of each selected objec

## 3

Select **Details** from the pull-down menu.

The **Nodes - Details** screen appears.



Server: ADSM. As of 04/07/1995 15:28:55 ( 04/07/1995 13:30:48 )

| Node Name | Platform | Domain | Days Since Access | Days Since Password Set | Locked |
|---|---|---|---|---|---|
| ASTRO | (?) | STANDARD | <1 | <1 | No |
| HDKLUNA | (?) | HDK1 | 8 | 8 | No |
| HDKLUNA1 | (?) | HDK1 | 8 | 8 | No |
| HDKPEGASUS | (?) | HDK1 | 8 | 8 | No |
| HDKPEGASUS1 | (?) | HDK1 | 8 | 8 | No |
| LUNA | (?) | HDK1 | 8 | 8 | No |
| PEGASUS | (?) | HDK1 | 8 | 8 | No |

Each object in this view represents one client node

## 4

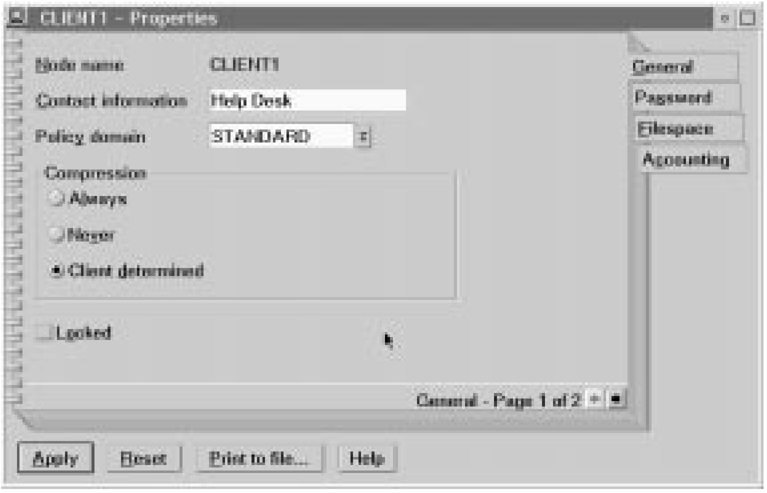
To close the session, see “Closing an Administrative Client Session” on page 19.

The Properties view window of a Node uses the following notebook tabs to display information about the objects/icon:

- The **General** notebook tab displays the node name, contact information, policy domain, file compression status, and locked status on the first page. The second page of the **General** notebook section shows platform access and registration information.
- The **Password** notebook tab allows you to change the password.
- The **Filespace** notebook tab displays whether the node owner can delete the node's backed up or archived files.
- The **Accounting** notebook tab displays valuable accounting information.

If you have policy or system privilege, you can change information in the Properties view.

For example, to open the **Properties** view window of the **Nodes** object, and to see additional information or change information:



Node name CLIENT1

Contact information Help Desk

Policy domain STANDARD

Compression

○ Always

○ Never

● Client determined

□ Locked

1

Select the **Nodes** icon.

2

Select **Selected** from the menu bar and then select **Open as**.

3

Select **Icons** from the pull-down menu.

4

Select a node from the Nodes-Icons screen.

5

Select **Selected** from the menu.

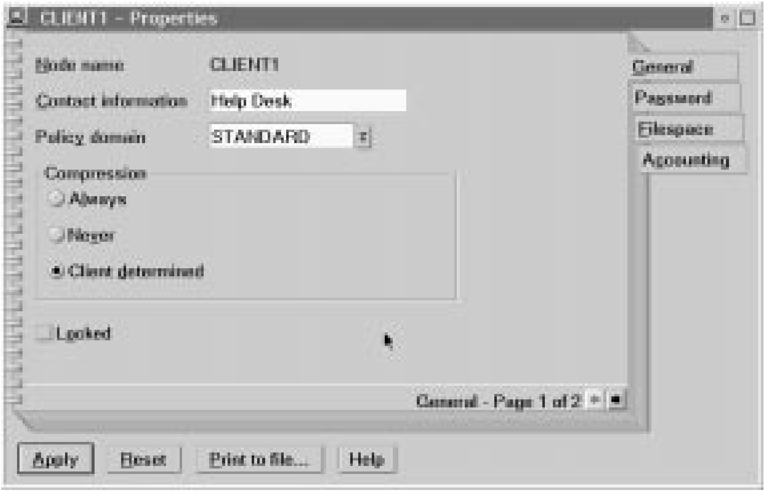
6

Select **Open as properties**.

The **Nodes - Properties** screen appears.

Use the tabs and arrow buttons at the bottom to navigate through the pages of the notebook.

To change information in an entry field of the **Properties** view window:



CLIENT1 - Properties

Node name CLIENT1

Contact information Help Desk

Policy domain STANDARD

Compression

○ Always

○ Never

● Client determined

□ Locked

General - Page 1 of 2

**1**

Select the field and enter the new information.

**2**

Press the Enter key to accept the information.

**3**

By using the mouse or cursor keys: change the status of the radio buttons, check button or select from the items in a drop down list.

When you have finished editing the data, select the **Apply** push button.

**4**

To close the session, see “Closing an Administrative Client Session” on page 19.

## Closing an Administrative Client Session

You can close an administrative client session in any one of the following ways:

- Double-click on the system menu symbol.
- Select the system menu. Then select **Close** from the pull-down menu that appears.
- Press Alt+F4 to close the most recent application window.

## Using Online Help

This section describes how to use online help to assist you with the following information:

- Understanding the types of help available
- Accessing help from windows
- Multiple ways to access task help
- Accessing contextual help
- Finding information in the help facility

### Understanding the Types of Help Available

You can request help to receive information about tasks, windows, and the selectable objects, menu choices, notebook tabs, fields, controls or push buttons. All of the following, except contextual help, are available from the Help menu.

The ADSM administrative graphical user interface provides the following types of online help:

**Task help** Provides a list of tasks that can be completed with the selected object. When you select a task, the help provides step-by-step information on how to complete the task.

**General help** Provides an overview of the function of the window. For example, general help for the Nodes—Icons window explains how to use the window, lists the tasks you can complete with the Nodes object, and displays links (routing choices) where you can obtain more details or related information about the Node object.

**Contextual help** Provides specific information for each selectable object, menu choice, notebook tab, field, and control or push button in a window. For example, you can access help for which information to enter in the password field.

**Help index** Provides an alphabetic listing of all the help topics. For example, within the help index, you can select **Keys Help** for information on how to use function keys or short cut keys in the ADSM administrative graphical user interface.

**Using help** Provides information on how to access the different types of help.

**Product information** Provides ADSM product information, including the copyright notice and release information.

**Accessing Help from Windows:** To access help in an ADSM administrative graphical user interface:

- Select **Help** from the menu bar
- Select an object/icon and press F1
- Select the Help button, if available, to display contextual help on a field or control that has cursor focus
- Press F2 from any contextual or task help window to display **General help**

***Accessing General Help:*** You can access general help for a window from the help index, from the help table of contents, from the help menu within the graphical user interface, or from a contextual or task help window of an application window:

**From the help index:**

In the Help **index**, select the name of the general help window that you want to view.

**From the help table of contents:**

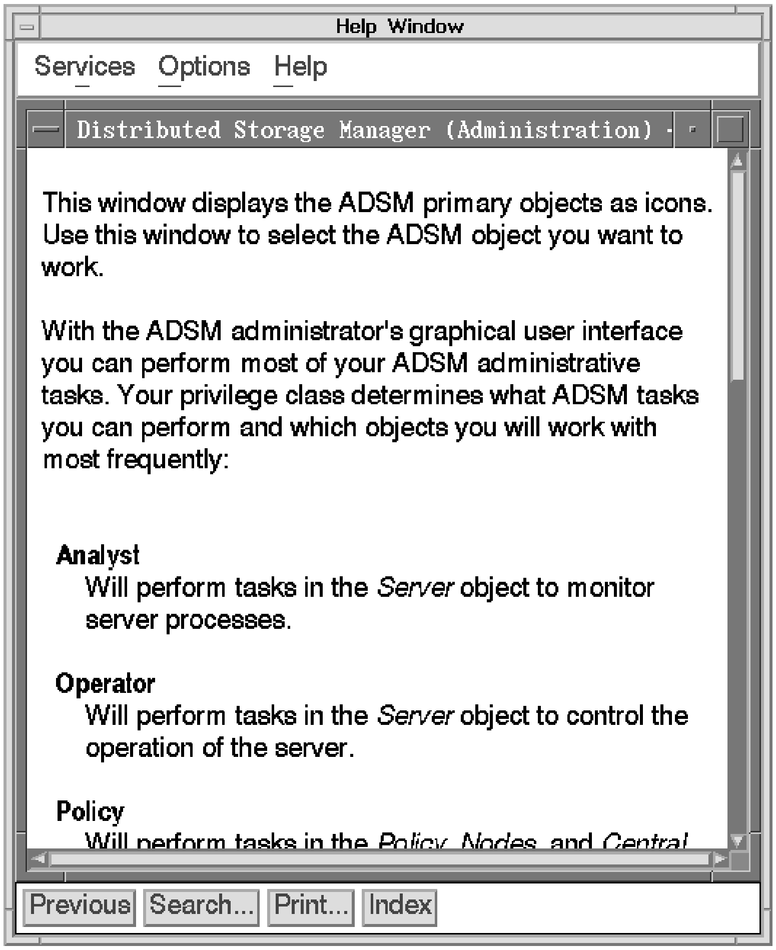
In the Help system, select **Contents** from the **Options** menu and select the help title from the table of contents list.

**From the help menu:**

Select **General help** from the pull-down menu under **Help** on the menu bar.

**From a contextual or task help window:**

Press F2.



This window displays the ADSM primary objects as icons. Use this window to select the ADSM object you want to work.

With the ADSM administrator's graphical user interface you can perform most of your ADSM administrative tasks. Your privilege class determines what ADSM tasks you can perform and which objects you will work with most frequently:

**Analyst**
Will perform tasks in the *Server* object to monitor server processes.

**Operator**
Will perform tasks in the *Server* object to control the operation of the server.

**Policy**
Will perform tasks in the *Policy*, *Nodes*, and *Central*

**Multiple Ways to Access Task Help:** To view **Task help** about the **Nodes** object/icon, for example, select one of the following:

- **Task help** from the help menu
- The task you need help with from the **Help index**
- Related Information from the **General help** window

To access **Task help**:

**1**

Select **Nodes**, for example.

**2**

Select **Help** from the menu bar on the Nodes Window.

**3**

Select **Task help** from the pull-down menu.

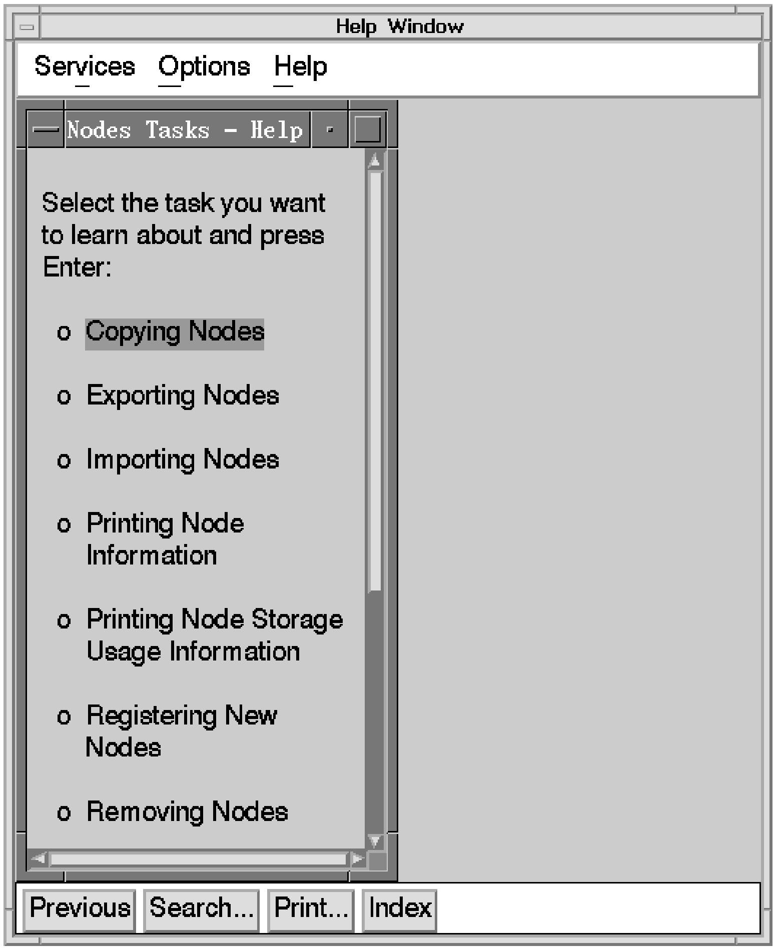
**4**

Follow the instructions in the windows for procedural information for a particular task.

A split window appears. Select a task from the task list on the left. The steps for the task display on the right.

**5**

Close the help window.



Select the task you want to learn about and press Enter:

o Copying Nodes

o Exporting Nodes

o Importing Nodes

o Printing Node Information

o Printing Node Storage Usage Information

o Registering New Nodes

o Removing Nodes

**Accessing Contextual Help:** You can access contextual help from a window by moving the cursor to a selectable object, menu choice, notebook tab, field, control, or push button.

To select contextual help:

**1**

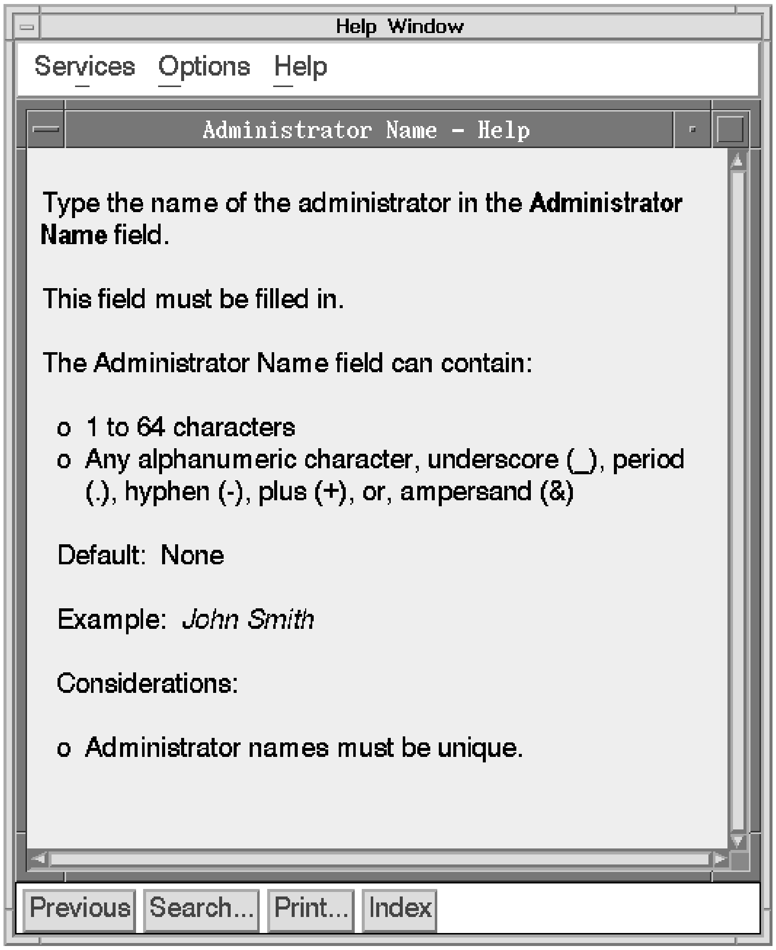
From the Logon window select, for example, the **Password** field.

**2**

Press F1.

**OR**

Select the **Help** push button, if it is displayed on the window.



Type the name of the administrator in the **Administrator Name** field.

This field must be filled in.

The Administrator Name field can contain:

o 1 to 64 characters
o Any alphanumeric character, underscore (_), period (.), hyphen (-), plus (+), or, ampersand (&)

Default: None

Example: *John Smith*

Considerations:

o Administrator names must be unique.

**Finding Information in the Help Facility:** After you have accessed the help facility, you can use the following help features:

- Hypertext links
- Search facility

**Using Hypertext Links:** Certain words or phrases, known as hypertext links, appear in a different color from the rest of the help text. Select the hypertext links to display related help:

**1**

Select an icon from the ADSM main window. For example, the **Administrators** icon.

**2**

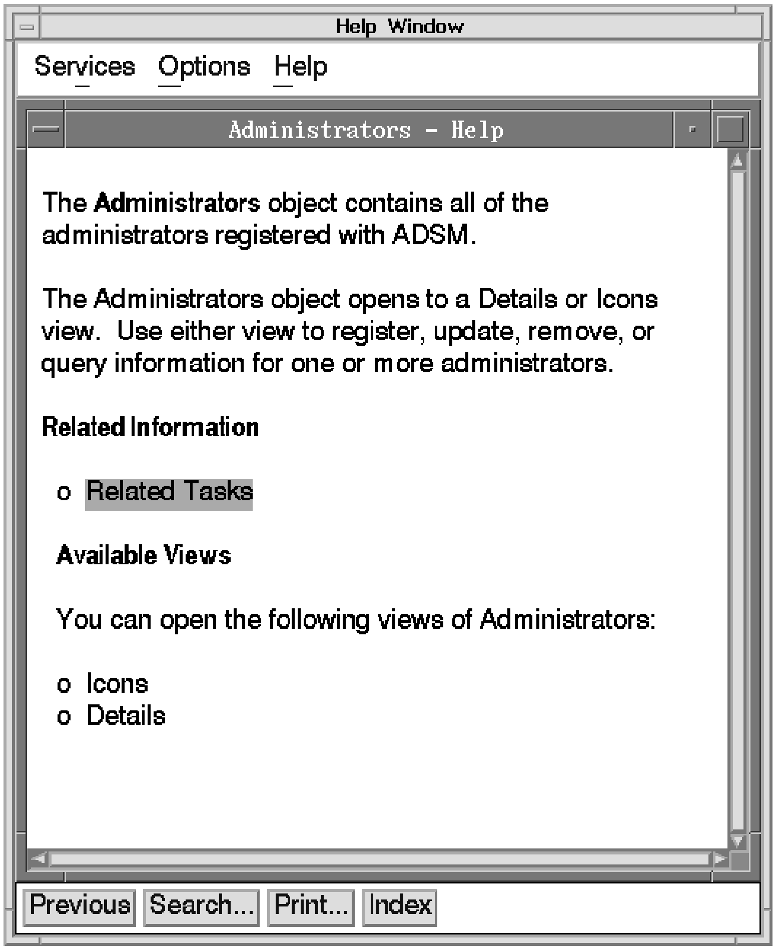
Press F1. Help for that icon is displayed.

**3**

For additional help information: select **Related Tasks**, **Icons**, or **Details**.

**4**

To close the window, use the Escape key or select the **Previous** push button.



The **Administrators** object contains all of the administrators registered with ADSM.

The Administrators object opens to a Details or Icons view. Use either view to register, update, remove, or query information for one or more administrators.

**Related Information**

o Related Tasks

**Available Views**

You can open the following views of Administrators:

o Icons
o Details

**Using the Search Facility:** You can use the search facility in the ADSM administrative graphical user interface help to find occurrences of words or phrases.

To access the search facility on a help panel:

**1**

Select **Administrators** object, for example.

**2**

Select **General help** from the help menu.

**3**

Select **Search** from the Services menu, or select the Search push button.

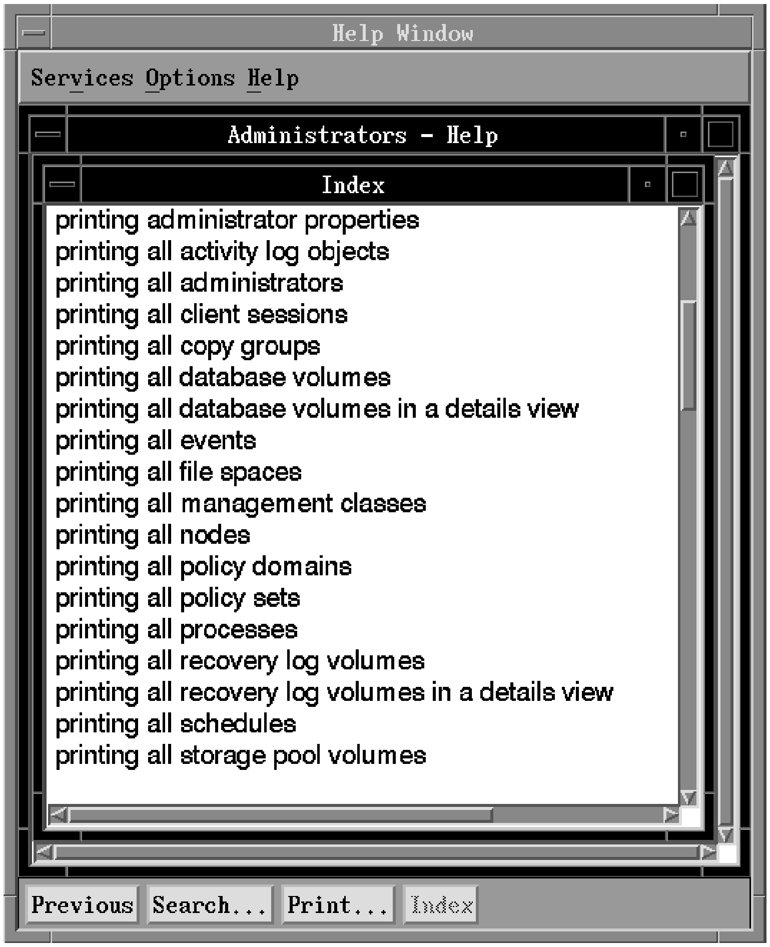
**4**

Type in the search word or words (for example, **print**).

**5**

Select the **Search** push button.

The search results appear in a help window.



printing administrator properties
printing all activity log objects
printing all administrators
printing all client sessions
printing all copy groups
printing all database volumes
printing all database volumes in a details view
printing all events
printing all file spaces
printing all management classes
printing all nodes
printing all policy domains
printing all policy sets
printing all processes
printing all recovery log volumes
printing all recovery log volumes in a details view
printing all schedules
printing all storage pool volumes

You also can search for every occurrence of a word or phrase in one or more help topics by using the radio buttons in the Search window.

For example, to search the help index for information about printing:

**1**

Type **print** in the Search string field.

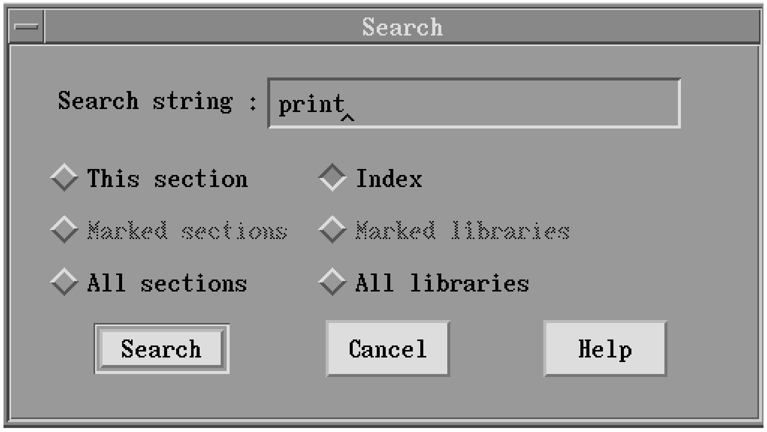
**2**

Select the **Index** radio button.

**3**

Select the **Search** push button.

The search results appear in a new help window.



Search string : print

To indicate to ADSM where to look for a word or phrase:

- Choose the **This section** radio button for the help topic that appears in the active help window.
- Choose the **Marked sections** radio button for selected help topics in the administrative graphical user interface.
- Choose the **All sections** radio button for all help topics in the administrative graphical user interface.
- Choose the **Index** radio button for the text of the index entries in the graphical user interface help.
- Choose the **Marked libraries** radio button for help topics in selected installed programs.
- Choose the **All libraries** radio button for help topics in all installed programs.

## Using the Administrative Command-Line Interface

The administrative command-line client lets administrators control and monitor the server through administrative commands. After you have installed the administrative client and modified the options file, registered administrators can access the server from any administrative client in the network. You can start an administrative client session in one of four modes:

- Console mode
- Mount mode
- Batch mode
- Interactive mode

You can also issue administrative commands from a server console session.

All these ways of accessing the server are discussed in this section. In addition, this section includes information about using administrative client options and macros.

For more information on using the command-line interface, see *ADSM Administrator's Reference*.

### Using Console Mode

Use console mode to monitor server activities as they occur. For example, you can monitor clients logging on to ADSM. This information is displayed on your terminal and, optionally, can be written to a file.

**Note:** You cannot enter administrator commands in console mode.

To start the administrative client in console mode, enter:

```
dsmadmc -consolemode
```

You are then prompted to enter your user ID and password. If you do not want to be prompted for that information, you can also enter your user ID and password in the DSMADMC command by using the ID and PASSWORD options. For example, enter:

```
dsmadmc -id=yourid -password=secret -consolemode
```

To end an administrative client session, use one of the following keyboard break sequences:

| Environment | Break Sequence |
|---|---|
| UNIX | Ctrl+C |
| CMS | HX |
| DOS | Ctrl+C, Ctrl+Break |
| OS/2 | Ctrl+C, Ctrl+Break |
| TSO | ATTN |
| Windows and Windows NT | Ctrl+C, Ctrl+Break |

## Using Mount Mode

Use mount mode to monitor removable media mount messages. This information is displayed on your terminal and, optionally, can be written to a file.

**Note:** You cannot enter administrator commands in mount mode.

To start the administrative client in mount mode, enter:

```
dsmadmc -mountmode
```

You are then prompted to enter your user ID and password. If you do not want to be prompted for that information, you can also enter your user ID and password in the DSMADMC command by using the ID and PASSWORD options. For example, enter:

```
dsmadmc -id=yourid -password=secret -mountmode
```

To end an administrative client session in mount mode, use the Ctrl+C keyboard break sequence.

## Using Batch Mode

Use batch mode to enter a single command, which can be a MACRO command. Your administrative client session automatically ends when the command or macro has executed.

For example, to start the administrative client in batch mode and issue the ENABLE command, enter:

```
dsmadmc -id=smith -password=secret enable
```

## Using Interactive Mode

Use interactive mode to enter a series of administrative commands. To start an administrative client session in interactive mode, a server session must be available.

To start the administrative client in interactive mode, enter:

```
dsmadmc
```

You are then prompted to enter your user ID and password. If you do not want to be prompted for that information, you can also enter your user ID and password in the DSMADMC command by using the ID and PASSWORD options. For example, enter:

```
dsmadmc -id=yourid -password=secret
```

To end an administrative client session in interactive mode, use the QUIT command:

```
quit
```

## Using Administrative Client Options

In all administrative client modes, the administrative client options modify your administrative client session responses. For example, the `-quiet` option specifies that you do not want ADSM to write any standard output messages to your terminal. For details about other options, see *ADSM Administrator's Reference*.

## Using Macros to Issue Commands

A macro is a file that contains one or more ADSM administrative commands. You can use macros when you want to issue commands repeatedly. You create a macro and issue the MACRO command with the name of the macro and, optionally, values for any substitution variables in the macro. For example, to issue the commands in the macro file named REGENG.MAC. enter:

```
macro regeng.mac
```

You can only issue macros from the administrative client in batch or interactive mode.

You can control whether the changes made to the database are permanent or not by using the COMMIT and ROLLBACK commands.

**COMMIT** Use this command to commit changes to the database made by the commands in a macro. For example:

```
/* macro to register policy administrators & grant authority*/
register admin jones boat
grant authority jones classes=policy
commit
register admin brown plane
grant authority brown classes=policy
commit
```

**ROLLBACK** Use this command to undo any changes to the database made by the commands in a macro, but not yet committed. By using the ROLLBACK command, you can test a macro without making the changes called for in the macro. For example:

```
/* macro to register policy administrators & grant authority*/
register admin jones boat
grant authority jones classes=policy
register admin brown plane
grant authority brown classes=policy
rollback
```

**Note:** If an administrative client session is running with the ITEMCOMMIT administrative client option, the ROLLBACK command has no effect.

For details about the MACRO command and the ITEMCOMMIT option, see *ADSM Administrator's Reference*.

## Using the Server Console Session

The *server console* is the terminal from which an administrator activates the server after ADSM is installed. ADSM has set up a special administrative user ID named SERVER_CONSOLE that allows you to activate the server from the server console after ADSM is installed. SERVER_CONSOLE is automatically registered as an administrator and is given system authority. Use the SERVER_CONSOLE administrator user ID to register and grant system privileges to any other administrator as soon as ADSM is installed. (Authentication must be turned off to use SERVER_CONSOLE as an administrative client ID.)

After you register other system administrators, reduce the authority of SERVER_CONSOLE to operator privilege to restrict access to administrative functions. Run the server from one of your newly registered administrative clients, rather than continuing to use SERVER_CONSOLE. For information on using the

SERVER_CONSOLE administrator user ID, see *ADSM Installing the Server and Administrative Client*.

You cannot modify SERVER_CONSOLE attributes and therefore cannot make the following changes to this user ID:

- Register or update SERVER_CONSOLE
- Lock or unlock SERVER_CONSOLE from ADSM
- Rename SERVER_CONSOLE
- Remove SERVER_CONSOLE

The SERVER_CONSOLE administrative ID does *not* receive a confirmation message when issuing commands that affect the availability of the server or data managed by the server. For more information on how to issue commands from the SERVER_CONSOLE administrative ID, refer to *ADSM Administrator's Reference*.

You should change the privilege class of the SERVER_CONSOLE administrative ID to operator privilege in order to prevent your tape operators from having system access to the server.

## Application Programming Interface

ADSM provides an application programming interface (API) that can be used by software vendors to integrate a storage management solution with existing applications to ensure that critical data is protected and easily recoverable.

When an application uses the ADSM API, it becomes an ADSM application client that can communicate with an ADSM server to backup, archive, or recover objects from ADSM storage.

An example of an application client is IBM Database 2 AIX (DB2/6000) for the RISC System/6000, which provides online backup services through its own interface.

After an application client is installed, a user must modify the client options file to identify the node name of the workstation and the communication method used to communicate with the server.

Applications residing on AIX, HP-UX, OS/2, SunOS/Solaris, or Windows can be developed or modified to use the ADSM Storage Management API.

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.

## Task, GUI, and Command Cross-Reference

The following tables list administrator tasks and show where they can be performed on the graphical user interface and command-line interface. The sequence of windows is shown under “Location in the GUI.” You can find detailed help for using the graphical user interface in its online help facility. For details about the commands, refer to the *ADSM Administrator's Reference*, and the command-line interface online help (accessed through the HELP command).

## Tasks Found in “Using ADSTAR Distributed Storage Manager Interfaces”

Table 1 shows a listing of tasks and commands referenced in Chapter 2, “Using ADSTAR Distributed Storage Manager Interfaces” on page 7.

*Table 1. Tasks Found in “Using ADSTAR Distributed Storage Manager Interfaces”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Start the administrative client. (See “Using the Administrative Command-Line Interface” on page 27) | Not available | DSMADMC |
| End the interactive mode of the administrative command-line client. (See “Using Interactive Mode” on page 29.) | Not available | QUIT |
| Invoke a macro file of one or more ADSM commands. (See “Using Macros to Issue Commands” on page 29.) | Not available | MACRO |
| Commit changes made by commands. (See “Using Macros to Issue Commands” on page 29.) | Automatic | COMMIT |
| Rollback uncommitted changes made by commands. (See “Using Macros to Issue Commands” on page 29.) | Not available | ROLLBACK |

## Tasks Found in “Managing Server Operations”

Table 2 shows a listing of tasks and commands referenced in Chapter 3, “Managing Server Operations” on page 61.

*Table 2 (Page 1 of 2). Tasks Found in “Managing Server Operations”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Start or restart the ADSM server. (See “Starting the Server” on page 61) | Not available | DSMSERV |
| Shut down the server. (See “Halting the Server” on page 62.) | Server | HALT |
| Query one or more client sessions. (See “Requesting Information about Client Sessions” on page 64.) | 1. Server<br>2. Sessions | QUERY SESSION |
| Cancel one or more client sessions. (See “Canceling a Client Session” on page 65.) | 1. Server<br>2. Sessions | CANCEL SESSION |
| Temporarily prevent client node access to the server. (See “Disabling or Enabling Server Access” on page 66.) | Server | DISABLE |
| Resume user activity on the server. (See “Disabling or Enabling Server Access” on page 66.) | Server | ENABLE |
| Request information about server background processes. (See “Requesting Information about Server Processes” on page 67.) | 1. Server<br>2. Processes | QUERY PROCESS |
| Cancel a server background processes. (See “Canceling Server Processes” on page 68.) | 1. Server<br>2. Processes | CANCEL PROCESS |

*Table 2 (Page 2 of 2). Tasks Found in “Managing Server Operations”*

| **Task** | **Location in the GUI** | **Command Used** |
|---|---|---|
| Bring a random access volume online or offline. (See “Varying Disk Volumes Online or Offline” on page 68.) | *Database volumes:*<br>1. Database<br>2. Database Volumes<br>*Recovery log volumes:*<br>1. Database Recovery Log<br>2. Recovery Log Volumes<br>*Storage pool volumes:*<br>1. Storage Pools<br>2. Storage Pool Volumes | VARY |
| Query system parameters. (See “Requesting Information about Server Status” on page 68.) | Server | QUERY STATUS |
| Specify the server name. (See “Setting the Server Name” on page 69) | Server | SET SERVERNAME |
| Query one or more server options. (See “Querying Server Options” on page 70.) | Not available | QUERY OPTION |
| Set the retention period for the activity log. (See “Setting the Activity Log Retention Period” on page 71.) | Server | SET ACTLOGRETENTION |
| Search activity log for messages. (See “Requesting Information from the Activity Log” on page 71.) | 1. Server<br>2. Activity Log | QUERY ACTLOG |
| Set accounting records on or off. (See “Monitoring Accounting Records” on page 72.) | Server | SET ACCOUNTING |
| Get help on commands and error messages. (See “Getting Help on Commands and Error Messages” on page 74.) | Not available | HELP |

## Tasks Found in “Managing the Database and Recovery Log”

Table 3 shows a listing of tasks and commands referenced in Chapter 4, “Managing the Database and Recovery Log” on page 75.

*Table 3 (Page 1 of 2). Tasks Found in “Managing the Database and Recovery Log”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Reset the utilization statistics for the database. (See “Monitoring Utilization” on page 82.) | Database | RESET DBMAXUTILIZATION |
| Reset the utilization statistics for the log. (See “Monitoring Utilization” on page 82.) | Database Recovery Log | RESET LOGMAXUTILIZATION |
| Allocate space for the database and recovery log. (See “Allocating Database or Recovery Log Space on Disk Storage” on page 84.) | Not available | DSMFMT |
| Define a database volume. (See “Step 1: Defining Disk Volumes” on page 85.) | 1. Database<br>2. Database Volumes | DEFINE DBVOLUME |
| Define a recovery log volume. (See “Step 1: Defining Disk Volumes” on page 85.) | 1. Database Recovery Log<br>2. Recovery Log Volumes | DEFINE LOGVOLUME |
| Increase the assigned capacity of the database. (See “Step 3: Extending the Capacity of the Database or Recovery Log” on page 87.) | Database | EXTEND DB |
| Increase the assigned capacity of the recovery log. (See “Step 3: Extending the Capacity of the Database or Recovery Log” on page 87.) | Database Recovery Log | EXTEND LOG |
| Display information about volumes defined to the database. (See “Step 1: Determining the Size of Volumes” on page 89.) | 1. Database<br>2. Database Volumes | QUERY DBVOLUME |

*Table 3 (Page 2 of 2). Tasks Found in “Managing the Database and Recovery Log”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Display information about volumes defined to the recovery log. (See “Step 1: Determining the Size of Volumes” on page 89.) | 1. Database Recovery Log<br>2. Recovery Log Volumes | QUERY LOGVOLUME |
| Decrease the assigned capacity of the database. (See “Step 3: Reducing the Capacity of the Database or Recovery Log” on page 92.) | Database | REDUCE DB |
| Decrease the assigned capacity of the recovery log. (See “Step 3: Reducing the Capacity of the Database or Recovery Log” on page 92.) | Database Recovery Log | REDUCE LOG |
| Delete a database volume. (See “Step 4: Deleting a Volume from the Database or Recovery Log” on page 93.) | 1. Database<br>2. Database Volumes | DELETE DBVOLUME |
| Delete a recovery log volume. (See “Step 4: Deleting a Volume from the Database or Recovery Log” on page 93.) | 1. Database Recovery Log<br>2. Recovery Log Volumes | DELETE LOGVOLUME |
| Display information on the database. (See “Requesting Information about the Database Buffer Pool” on page 95.) | Database | QUERY DB |
| Reset the buffer pool statistics for the database. (See “Resetting Database Buffer Pool Statistics” on page 96.) | Not available | RESET BUFPOOL |
| Display information on the recovery log. (See “Requesting Information about the Recovery Log Buffer Pool” on page 97.) | Database Recovery Log | QUERY LOG |

## Tasks Found in “Managing Licensing, Privilege Classes, and Registration”

Table 4 shows a listing of tasks and commands referenced in Chapter 5, “Managing Licensing, Privilege Classes, and Registration” on page 99.

*Table 4 (Page 1 of 3). Tasks Found in “Managing Licensing, Privilege Classes, and Registration”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Register a new license. (See “Managing ADSM Licenses” on page 99.) | Server | REGISTER LICENSE |
| Audit the current server configuration and licenses. (See “Monitoring Licenses” on page 104.) | Server | AUDIT LICENSES |
| Display license information. (See “Monitoring Licenses” on page 104.) | Server | QUERY LICENSE |
| Set license audit period. (See “Monitoring Licenses” on page 104.) | Server | SET LICENSEAUDITPERIOD |
| Set password expiration date. (See “Setting User Password Expiration” on page 105.) | Server | SET PASSEXP |
| Set password authentication. (See “Setting Client Password Authentication” on page 104.) | Server | SET AUTHENTICATION |
| Register an administrator. (See “Registering Administrators or Updating Information” on page 105.) | Administrators | REGISTER ADMIN |
| Update an administrator. (See “Registering Administrators or Updating Information” on page 105.) | Administrators | UPDATE ADMIN |
| Add administrator authority. (See “Granting Administrative Authority” on page 106.) | Administrators | GRANT AUTHORITY |
| Revoke or reduce administrator authority. (See “Revoking or Reducing Administrative Authority” on page 110.) | Administrators | REVOKE AUTHORITY |

*Table 4 (Page 2 of 3). Tasks Found in "Managing Licensing, Privilege Classes, and Registration"*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Rename an administrator. (See "Renaming an Administrator" on page 113.) | Administrators | RENAME ADMIN |
| Delete an administrator. (See "Removing Administrators" on page 114.) | Administrators | REMOVE ADMIN |
| Lock out an administrator. (See "Locking and Unlocking Administrators from the Server" on page 114.) | Administrators | LOCK ADMIN |
| Unlock an administrator. (See "Locking and Unlocking Administrators from the Server" on page 114.) | Administrators | UNLOCK ADMIN |
| Display information on one or more administrators. (See "Requesting Information about Administrators" on page 115.) | Administrators | QUERY ADMIN |
| Set open or closed registration. (See "Setting Client Node Registration" on page 116.) | Central Scheduler | SET REGISTRATION |
| Register a client node. (See "Registering Client Nodes" on page 118.) | Nodes | REGISTER NODE |
| Update a client node. (See "Updating Client Node Information" on page 119.) | Nodes | UPDATE NODE |
| Rename a client node. (See "Renaming Client Nodes" on page 119.) | Nodes | RENAME NODE |
| Lock out a client node. (See "Locking and Unlocking Client Nodes" on page 120.) | Nodes | LOCK NODE |
| Unlock a client node. (See "Locking and Unlocking Client Nodes" on page 120.) | Nodes | UNLOCK NODE |

*Table 4 (Page 3 of 3). Tasks Found in "Managing Licensing, Privilege Classes, and Registration"*

| **Task** | **Location in the GUI** | **Command Used** |
|---|---|---|
| Query one or more client nodes. (See "Requesting Information about Client Nodes" on page 120.) | Nodes | QUERY NODE |
| Query one or more file spaces. (See "Requesting File Space Information" on page 122.) | File Spaces | QUERY FILESPACE |
| Delete client node data from the server. (See "Deleting a File Space" on page 123.) | File Spaces | DELETE FILESPACE |
| Delete a client node. (See "Removing Client Nodes" on page 124.) | Nodes | REMOVE NODE |
| Register an application programming interface. (See "Registering an Application Programming Interface to the Server" on page 124.) | Nodes | REGISTER NODE |

## Tasks Found in “Managing Policies”

Table 5 shows a listing of tasks and commands referenced in Chapter 6, “Managing Policies” on page 127.

*Table 5 (Page 1 of 3). Tasks Found in “Managing Policies”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Define a new policy domain. (See “Example: Defining a Policy Domain” on page 142.) | Policy Domains | DEFINE DOMAIN |
| Copy a policy domain. (See “Example: Defining a Policy Domain” on page 142.) | Policy Domains | COPY DOMAIN |
| Update a policy domain. (See “Example: Defining a Policy Domain” on page 142.) | Policy Domains | UPDATE DOMAIN |
| Copy a policy set. (See “Defining and Updating a Policy Set” on page 143.) | 1. Policy Domains<br>2. Policy Sets | COPY POLICYSET |
| Define a policy set. (See “Defining and Updating a Policy Set” on page 143.) | 1. Policy Domains<br>2. Policy Sets | DEFINE POLICYSET |
| Update a policy set. (See “Defining and Updating a Policy Set” on page 143.) | 1. Policy Domains<br>2. Policy Sets | UPDATE POLICYSET |
| Define a management class. (See “Defining and Updating a Management Class” on page 144.) | 1. Policy Domains<br>2. Management Class | DEFINE MGMTCLASS |
| Copy a management class. (See “Example: Define a New Management Class” on page 144.) | 1. Policy Domains<br>2. Management Class | COPY MGMTCLASS |
| Update a management class (See “Example: Define a New Management Class” on page 144.) | 1. Policy Domains<br>2. Management Class | UPDATE MGMTCLASS |
| Define a backup copy group. (See “Defining and Updating a Backup Copy Group” on page 145.) | 1. Policy Domains<br>2. Backup Copy Groups | DEFINE COPYGROUP |

*Table 5 (Page 2 of 3). Tasks Found in "Managing Policies"*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Update a backup copy group. (See "Defining and Updating a Backup Copy Group" on page 145.) | 1. Policy Domains<br>2. Backup Copy Groups | UPDATE COPYGROUP |
| Define an archive copy group. (See "Defining and Updating an Archive Copy Group" on page 149.) | 1. Policy Domains<br>2. Archive Copy Groups | DEFINE COPYGROUP |
| Update an archive copy group. (See "Defining and Updating an Archive Copy Group" on page 149.) | 1. Policy Domains<br>2. Archive Copy Groups | UPDATE COPYGROUP |
| Assign a default management class. (See "Assigning a Default Management Class" on page 150.) | 1. Policy Domains<br>2. Management Class | ASSIGN DEFMGMTCLASS |
| Activate policy set. (See "Validating and Activating Policy Sets" on page 150.) | 1. Policy Domains<br>2. Policy Sets | ACTIVATE POLICYSET |
| Verifies a policy set. (See "Validating and Activating Policy Sets" on page 150.) | 1. Policy Domains<br>2. Policy Sets | VALIDATE POLICYSET |
| Start inventory expiration processing. (See "Starting Expiration Processing" on page 152.) | Not available | EXPIRE INVENTORY |
| Display information about a copy group. (See "Querying Copy Groups" on page 153.) | 1. Policy Domains<br>2. Backup Copy Groups or Archive Copy Groups | QUERY COPYGROUP |
| Query a management class. (See "Querying Management Classes" on page 153.) | 1. Policy Domains<br>2. Management Class | QUERY MGMTCLASS |
| Query a policy set. (See "Querying Policy Sets" on page 154.) | 1. Policy Domains<br>2. Policy Sets | QUERY POLICYSET |
| Query one or more policy domains. (See "Querying Policy Domains" on page 155.) | Policy Domains | QUERY DOMAIN |

*Table 5 (Page 3 of 3). Tasks Found in "Managing Policies"*

| **Task** | **Location in the GUI** | **Command Used** |
|---|---|---|
| Delete a copy group. (See "Deleting Copy Groups" on page 155.) | 1. Policy Domains<br>2. Backup Copy Groups or Archive Copy Groups | DELETE COPYGROUP |
| Delete a management class. (See "Deleting Management Classes" on page 156.) | 1. Policy Domains<br>2. Management Class | DELETE MGMTCLASS |
| Delete a policy set. (See "Deleting Policy Sets" on page 156.) | 1. Policy Domains<br>2. Policy Sets | DELETE POLICYSET |
| Delete a policy domain. (See "Deleting Policy Domains" on page 157.) | Policy Domains | DELETE DOMAIN |

## Tasks Found in “Scheduling Operations”

Table 6 shows a listing of tasks and commands referenced in Chapter 7, “Scheduling Operations” on page 159.

*Table 6 (Page 1 of 2). Tasks Found in “Scheduling Operations”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Select a central scheduling mode. (See “Setting the Scheduling Mode on the Server” on page 162.) | Central Scheduler | SET SCHEDMODES |
| Set maximum scheduled sessions. (See “Setting the Maximum Percentage of Sessions for Scheduled Operations” on page 163.) | Central Scheduler | SET MAXSCHEDSESSIONS |
| Randomly distribute scheduled start times. (See “Randomizing Schedule Start Times” on page 164.) | Central Scheduler | SET RANDOMIZE |
| Control how often client nodes contact the server to perform scheduled operations. (See “Setting How Often Clients Query the Server” on page 165.) | Central Scheduler | SET QUERYSCHEDPERIOD |
| Set number of times scheduler retries commands. (See “Setting the Number of Command Retry Attempts” on page 166.) | Central Scheduler | SET MAXCMDRETRIES |
| Set time between retry attempts. (See “Setting the Amount of Time between Retry Attempts” on page 166.) | Central Scheduler | SET RETRYPERIOD |
| Define a backup or archive schedule. (See “Defining or Updating Schedules” on page 167.) | 1. Central Scheduler<br>2. Backup/Archive Schedules | DEFINE SCHEDULE |
| Update a schedule. (See “Defining or Updating Schedules” on page 167.) | 1. Central Scheduler<br>2. Administrative Command Schedules or Backup/Archive Schedules | UPDATE SCHEDULE |

*Table 6 (Page 2 of 2). Tasks Found in “Scheduling Operations”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Copy a schedule. (See “Copying Schedules” on page 171.) | 1. Central Scheduler<br>2. Administrative Command Schedules or Backup/Archive Schedules | COPY SCHEDULE |
| Query one or more schedules. (See “Querying Schedules” on page 172.) | 1. Central Scheduler<br>2. Administrative Command Schedules or Backup/Archive Schedules | QUERY SCHEDULE |
| Delete one or more schedules. (See “Deleting Schedules” on page 172.) | 1. Central Scheduler<br>2. Administrative Command Schedules or Backup/Archive Schedules | DELETE SCHEDULE |
| Associate client nodes with a schedule. (See “Associating Client Nodes with Schedules” on page 172.) | 1. Central Scheduler<br>2. Administrative Events or Backup/Archive Events | DEFINE ASSOCIATION |
| Query client node associations with a schedule. (See “Querying Associations” on page 173.) | 1. Central Scheduler<br>2. Administrative Events or Backup/Archive Events | QUERY ASSOCIATION |
| Delete node associations with a schedule. (See “Deleting Associations” on page 173.) | 1. Central Scheduler<br>2. Administrative Events or Backup/Archive Events | DELETE ASSOCIATION |
| Query scheduled and completed events. (See “Querying Event Records” on page 174.) | 1. Central Scheduler.<br>2. Administrative Events or Backup/Archive Events | QUERY EVENT |
| Set the retention period for event records. (See “Setting the Event Record Retention Period” on page 175.) | 1. Central Scheduler<br>2. Administrative Events or Backup/Archive Events | SET EVENTRETENTION |
| Delete event records. (See “Deleting Event Records” on page 175.) | 1. Central Scheduler<br>2. Administrative Events or Backup/Archive Events | DELETE EVENT |

## Tasks Found in “Managing Drives and Libraries”

Table 7 shows a listing of tasks and commands referenced in Chapter 9, “Managing Drives and Libraries” on page 201.

*Table 7 (Page 1 of 2). Tasks Found in “Managing Drives and Libraries”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Change the status of a storage volume. (See “Private Volumes in a Library” on page 208.) | Not available | UPDATE LIBVOLUME |
| Define a library. (See “Defining and Managing Libraries” on page 209.) | Not available | DEFINE LIBRARY |
| Query a library. (See “Querying Libraries” on page 210.) | Not available | QUERY LIBRARY |
| Delete a library. (See “Deleting Libraries” on page 210.) | Not available | DELETE LIBRARY |
| Update a library. (See “Updating Libraries” on page 210.) | Not available | UPDATE LIBRARY |
| Define a drive to a library. (See “Defining and Managing Drives” on page 211.) | Not available | DEFINE DRIVE |
| Display information about a drive. (See “Querying Drives” on page 212.) | Not available | QUERY DRIVE |
| Delete a drive from a library. (See “Deleting Drives” on page 213.) | Not available | DELETE DRIVE |
| Update a drive. (See “Updating Drives” on page 212.) | Not available | UPDATE DRIVE |
| Check a storage volume into a library. (See “Informing the Server about New Volumes” on page 213.) | Not available | CHECKIN LIBVOLUME |
| Check a storage volume out of a library. (See “Removing Volumes from a Library” on page 215.) | Not available | CHECKOUT LIBVOLUME |

*Table 7 (Page 2 of 2). Tasks Found in “Managing Drives and Libraries”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Verify an automated library’s inventory. (See “Auditing a Library’s Volume Inventory” on page 216.) | Not available | AUDIT LIBRARY |
| Query one or more pending mount requests. (See “Querying Pending Operator Requests” on page 218.) | Not available | QUERY REQUEST |
| Allow a request to continue processing. (See “Replying to Operator Requests” on page 218.) | Not available | REPLY |
| Cancel one or more mount requests. (See “Canceling an Operator Request” on page 218.) | Not available | CANCEL REQUEST |
| Display information on mounted sequential access volumes. (See “Determining Which Volumes are Mounted” on page 218.) | Not available | QUERY MOUNT |
| Dismount a volume by volume name. (See “Dismounting an Idle Volume” on page 218.) | Not available | DISMOUNT VOLUME |

## Tasks Found in “Managing Storage Devices”

Table 8 shows a listing of tasks and commands referenced in Chapter 10, “Managing Storage Devices” on page 219.

*Table 8. Tasks Found in “Managing Storage Devices“*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Define a device class for tape. (See “Defining Device Classes for Tape” on page 221.) | Not available | DEFINE DEVCLASS |
| Define a device class for optical. (See “Defining Device Classes for Optical Drives” on page 230.) | Not available | DEFINE DEVCLASS |
| Define a device class for FILE. (See “Defining FILE Device Classes” on page 231.) | Not available | DEFINE DEVCLASS |
| Display information on one or more device classes. (See “Requesting Information about a Device Class” on page 233.) | Not available | QUERY DEVCLASS |
| Change the attributes of a device class. (See “Updating Device Classes” on page 234.) | Not available | UPDATE DEVCLASS |
| Delete a device class. (See “Deleting Device Classes” on page 235.) | Not available | DELETE DEVCLASS |

## Tasks Found in “Managing Storage Pools”

Table 9 shows a listing of tasks and commands referenced in Chapter 11, “Managing Storage Pools” on page 237.

*Table 9. Tasks Found in “Managing Storage Pools”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Define a storage pool. (See “Defining or Updating Storage Pools” on page 262.) | Storage Pools | DEFINE STGPOOL |
| Change the attributes of a storage pool. (See “Defining or Updating Storage Pools” on page 262.) | Storage Pools | UPDATE STGPOOL |
| Back up a primary storage pool. (See “ Backing Up Storage Pools” on page 269.) | Storage Pools | BACKUP STGPOOL |
| Query one or more storage pools. (See “Monitoring the Use of Storage Pool Space” on page 271.) | Storage Pools | QUERY STGPOOL |
| Cancel a migration process. (See “Canceling the Migration Process” on page 275.) | 1. Server<br>2. Processes | CANCEL PROCESS |
| Display file space information by storage pool. (See “Amount of Space Used by Client Node” on page 278.) | Not available | QUERY OCCUPANCY |
| Delete a storage pool. (See “Deleting a Storage Pool” on page 280.) | Storage Pools | DELETE STGPOOL |
| Recreate files in a primary storage pool. (See “Restoring Storage Pools” on page 281.) | Storage Pools | RESTORE STGPOOL |

## Tasks Found in “Managing Storage Pool Volumes”

Table 10 shows a listing of tasks and commands referenced in Chapter 12, “Managing Storage Pool Volumes” on page 285.

*Table 10 (Page 1 of 2). Tasks Found in “Managing Storage Pool Volumes”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Define a volume in a storage pool. (See “Storage Pool Volumes” on page 285.) | 1. Storage Pools<br>2. Storage Pool Volumes | DEFINE VOLUME |
| Write a label to sequential storage volumes. (See “Labeling Sequential Storage Pool Volumes” on page 288.) | Not available | dsmlabel |
| Change user access to a storage pool volume. (See “Defining Storage Pool Volumes” on page 290.) | 1. Storage Pools<br>2. Storage Pool Volumes | UPDATE VOLUME |
| Query one or more storage pool volumes. (See “Requesting General Information about Storage Pool Volumes” on page 292.) | 1. Storage Pools<br>2. Storage Pool Volumes | QUERY VOLUME |
| Verify database information for a storage pool volume. (See “Auditing a Storage Pool Volume” on page 294.) | 1. Storage Pools<br>2. Storage Pool Volumes | AUDIT VOLUME |
| Query the contents of a storage pool volume. (See “Viewing a Standard Report on the Contents of a Volume” on page 299.) | 1. Storage Pools<br>2. Storage Pool Volumes | QUERY CONTENT |
| Move files on a storage pool volume. (See “Moving Files from One Volume to Another Volume” on page 300.) | 1. Storage Pools<br>2. Storage Pool Volumes | MOVE DATA |
| Query one or more server processes. (See “Requesting Information about the Data Movement Process” on page 303.) | 1. Server<br>2. Processes | QUERY PROCESS |

*Table 10 (Page 2 of 2). Tasks Found in “Managing Storage Pool Volumes”*

| **Task** | **Location in the GUI** | **Command Used** |
|---|---|---|
| Delete a storage pool volume. (See “Deleting an Empty Storage Pool Volume” on page 305.) | 1. Storage Pools<br>2. Storage Pool Volumes | DELETE VOLUME |
| Recreate files in a primary storage pool volume. (See “Restoring Storage Pool Volumes” on page 306.) | 1. Storage Pools<br>2. Storage Pool Volumes | RESTORE VOLUME |

## Tasks Found in “Exporting and Importing Data”

Table 11 shows a listing of tasks and commands referenced in Chapter 13, “Exporting and Importing Data” on page 309.

*Table 11 (Page 1 of 2). Tasks Found in “Exporting and Importing Data”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Copy server information to sequential media. (See “Preparing to Export or Import Data” on page 310.) | Server | EXPORT SERVER |
| Request information about an export or import process (See “Requesting Information about an Export or Import Process” on page 312.) | 1. Server<br>2. Processes | QUERY PROCESS |
| Copy administrator information to sequential media. (See “Exporting Administrator Information” on page 317.) | Administrators | EXPORT ADMIN |
| Copy client node information to sequential media. (See “Exporting Client Node Information” on page 318.) | Nodes | EXPORT NODE |
| Copy policy information to sequential media. (See “Exporting Policy Information” on page 319.) | Policy Domains | EXPORT POLICY |
| Import administrator information. (See “Importing Data from Sequential Media Volumes” on page 319.) | Administrators | IMPORT ADMIN |
| Import client node information. (See “Importing Data from Sequential Media Volumes” on page 319.) | Nodes | IMPORT NODE |
| Import policy information. (See “Importing Data from Sequential Media Volumes” on page 319.) | Policy Domains | IMPORT POLICY |
| Import the server. (See “Importing Data from Sequential Media Volumes” on page 319.) | Server | IMPORT SERVER |

*Table 11 (Page 2 of 2). Tasks Found in "Exporting and Importing Data"*

| **Task** | **Location in the GUI** | **Command Used** |
|---|---|---|
| Rename a client filespace on the server. (See "Renaming a File Space" on page 328.) | File spaces | RENAME FILESPACE |

## Tasks Found in “Recovering Data”

Table 12 shows a listing of tasks and commands referenced in Chapter 14, “Recovering Data” on page 329.

*Table 12 (Page 1 of 3). Tasks Found in “Recovering Data”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Define a volume copy of a database volume. (See “Defining Database or Recovery Log Mirrored Volumes” on page 331.) | 1. Database<br>2. Database Volumes | DEFINE DBCOPY |
| Define a volume copy of a recovery log volume. (See “Defining Database or Recovery Log Mirrored Volumes” on page 331.) | 1. Database Recovery Log<br>2. Recovery Log Volumes | DEFINE LOGCOPY |
| Display information on one or more database volumes. (See “Requesting Information about Mirrored Volumes” on page 332.) | 1. Database.<br>2. Database Volumes | QUERY DBVOLUME |
| Display information on one or more log volumes. (See “Requesting Information about Mirrored Volumes” on page 332.) | 1. Database Recovery Log<br>2. Recovery Log Volumes | QUERY LOGVOLUME |
| Save sequential volume history information to one or more files. (See “Establishing Volume History Backup Files” on page 339.) | 1. Server<br>2. Sequential Volume History | BACKUP VOLHISTORY |
| Delete sequential volume history information collected by the server. (See “Establishing Volume History Backup Files” on page 339.) | 1. Server<br>2. Sequential Volume History | DELETE VOLHISTORY |
| Display sequential volume history information. (See “Establishing Volume History Backup Files” on page 339.) | 1. Server<br>2. Sequential Volume History | QUERY VOLHISTORY |

*Table 12 (Page 2 of 3). Tasks Found in "Recovering Data"*

| **Task** | **Location in the GUI** | **Command Used** |
|---|---|---|
| Backup device configuration information. (See "Establishing Device Configuration Backup Files" on page 340.) | Server | BACKUP DEVCONFIG |
| Backup the database. (See "Performing the Database Backups" on page 342.) | Database | BACKUP DB |
| Define automatic database backup. (See "Setting the Database Backup Trigger" on page 343.) | Database | DEFINE DBBACKUPTRIGGER |
| Delete automatic database backup. (See "Setting the Database Backup Trigger" on page 343.) | Database | DELETE DBBACKUPTRIGGER |
| Update the automatic database backup. (See "Setting the Database Backup Trigger" on page 343.) | Database | UPDATE DBBACKUPTRIGGER |
| Query the setting for the back up trigger. (See "Setting the Database Backup Trigger" on page 343.) | Database | QUERY DBBACKUPTRIGGER |
| Set the recovery log mode. (See "Setting the Recovery Log Mode" on page 345.) | Database Recovery Log | SET LOGMODE |
| Restore the database to a point in time. (See "Restoring a Database by Using Point-in-Time Recovery" on page 346.) | Not available | DSMSERV RESTORE DB |
| Backup a primary storage pool. (See "Using Storage Pool Backup Features" on page 350.) | Storage Pools | BACKUP STGPOOL |

*Table 12 (Page 3 of 3). Tasks Found in "Recovering Data"*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Restore a primary storage pool from a copy storage pool. (See "Recovering by Using Backed Up Copies of Storage Pools" on page 351.) | Storage Pools | RESTORE STGPOOL |
| Restore a volume from a copy storage pool. (See "Recovery from Media Loss" on page 354.) | 1. Storage Pools<br>2. Storage Pool Volumes | RESTORE VOLUME |
| Do an offline dump of the database (See "Database Salvage Utilities" on page 355.) | Not available | DSMSERV DUMPDB |
| Initialize the database and recovery log (See "Database Salvage Utilities" on page 355.) | Not available | DSMSERV INSTALL |
| Reload a dumped database (See "Database Salvage Utilities" on page 355.) | Not available | DSMSERV LOADDB |
| Verify that a reloaded database is returned to a consistent state (See "Database Salvage Utilities" on page 355.) | Not available | DSMSERV AUDITDB |

## Tasks Found in "Using Disaster Recovery Manager"

Table 13 shows a listing of tasks and commands referenced in Chapter 15, "Using Disaster Recovery Manager" on page 359.

*Table 13 (Page 1 of 5). Tasks Found in "Using Disaster Recovery Manager"*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Register Disaster Recovery Manager. (See "Enabling Disaster Recovery Manager" on page 363.) | Not available | REGISTER LICENSE |

*Table 13 (Page 2 of 5). Tasks Found in "Using Disaster Recovery Manager"*

| **Task** | **Location in the GUI** | **Command Used** |
|---|---|---|
| Back up your storage pools (See "Creating a Backup Copy of Server Primary Storage Pools and Database" on page 363.) | Storage Pools | BACKUP STGPOOL |
| Back up your database. (See "Creating a Backup Copy of Server Primary Storage Pools and Database" on page 363.) | Database | BACKUP DB |
| Eject the volumes from the library (if required) and mark them unavailable. (See "Sending Server Backup Volumes Offsite" on page 364.) | Not available | MOVE DRMEDIA |
| Mark the backed up volumes as offsite. (See "Sending Server Backup Volumes Offsite" on page 364.) | Not available | MOVE DRMEDIA |
| Define boot recovery media. (See "Defining and Tracking Recovery Media" on page 394.) | Not available | DEFINE RECOVERYMEDIA |
| Associate one or more machines with a recovery media. (See "Defining and Tracking Recovery Media" on page 394.) | Not available | DEFINE RECMEDMACHASSOCIATION |

*Table 13 (Page 3 of 5). Tasks Found in "Using Disaster Recovery Manager"*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Update the location when boot media is moved offsite. (See "Defining and Tracking Recovery Media" on page 394.) | Not available | UPDATE RECOVERYMEDIA |
| Create a disaster recovery plan file. (See "Creating the ADSM Server Disaster Recovery Plan File" on page 368.) | Not available | PREPARE |
| Define client machine location. (See "Defining Machine Information" on page 391.) | Not available | DEFINE MACHINE |
| Associate one or more client nodes with a machine. (See "Defining Machine Information" on page 391.) | Not available | DEFINE MACHNODEASSOCIATION |
| Move reclaimed volumes back onsite. (See "Moving Reclaimed or Expired Volumes Back Onsite" on page 366.) | Not available | MOVE DRMEDIA |
| Specify copy storage pools to be managed. (See "Customizing the Management of Storage Pools, Path Name Prefixes, and Replacement Volume Identifiers" on page 402.) | Not available | SET DRMCOPYSTGPOOL |

*Table 13 (Page 4 of 5). Tasks Found in “Using Disaster Recovery Manager”*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Specify primary storage pools to be managed. (See “Customizing the Management of Storage Pools, Path Name Prefixes, and Replacement Volume Identifiers” on page 402.) | Not available | SET DRMPRIMSTGPOOL |
| Specify character identification for replacement volume names. (See “Customizing the Management of Storage Pools, Path Name Prefixes, and Replacement Volume Identifiers” on page 402.) | Not available | SET DRMPLANPOSTFIX |
| Specify the directory where recovery plan instruction stanzas reside. (See “Customizing the Management of Storage Pools, Path Name Prefixes, and Replacement Volume Identifiers” on page 402.) | Not available | SET DRMINSTRPREFIX |
| Specify the directory where the recovery plan file resides. (See “Customizing the Management of Storage Pools, Path Name Prefixes, and Replacement Volume Identifiers” on page 402.) | Not available | SET DRMPLANPREFIX |

*Table 13 (Page 5 of 5). Tasks Found in "Using Disaster Recovery Manager"*

| Task | Location in the GUI | Command Used |
|---|---|---|
| Specify the name of the courier that transports recovery media volumes. (See "Customizing the Management of Offsite Recovery Media" on page 405.) | Not available | SET DRMCOURIERNAME |
| Specify whether ADSM should read sequential media labels of volumes checked out with the MOVE DRMEDIA command. (See "Customizing the Management of Offsite Recovery Media" on page 405.) | Not available | SET DRMCHECKLABEL |
| Specify the minimum number of days before a database series is expired. (See "Customizing the Management of Offsite Recovery Media" on page 405.) | Not available | SET DRMDBBACKUPEXPIREDAYS |
| Specify the name of the vault location where recovery media volumes are stored. (See "Customizing the Management of Offsite Recovery Media" on page 405.) | Not available | SET DRMVAULTNAME |

# Chapter 3. 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 begins at the indicated pages.

| Section | Page |
|---|---|
| **Tasks:** | |
| Starting, halting, or restarting the server | 61 |
| Freeing links for client connections | 63 |
| Managing client sessions | 64 |
| Disabling or enabling server access | 66 |
| Managing server processes | 66 |
| Varying disk volumes online or offline | 68 |
| Requesting information about server status | 68 |
| Setting the server name | 69 |
| Querying server options | 70 |
| Managing the activity log | 70 |
| Monitoring accounting records | 72 |
| Getting help on commands and error messages | 74 |

Most tasks presented in this chapter can be performed by using either the graphical user interface or the command line interface. Table 2 on page 34 shows whether a task can be performed on the graphical user interface, the command line interface, or both.

For information about issuing ADSM commands, refer to *ADSM Administrator's Reference*. For help performing a task on the graphical user interface, see the procedure described in "Using Online Help" on page 20.

## Starting, Halting, and Restarting the Server

| Task | Required Privilege Class |
|---|---|
| Start, halt, and restart the server | System or operator |

### Starting the Server

To start the server, complete the following steps:

1. Change to the /usr/lpp/adsmserv/bin directory from an AIX session.

   Enter:

   ```
   cd /usr/lpp/adsmserv/bin
   ```

2. Start the server.

   Enter:

   ```
   dsmserv
   ```

ADSM displays the following information when the server is started:

- Product licensing and copyright information
- Processing information about the server options file
- Communication protocol information
- Database and recovery log information
- Storage pool volume information
- Server generation date
- Progress messages and any errors encountered during server initialization

The following events occur when the server is started:

- The server invokes the communication methods specified in the server options file.
- The server uses the volumes specified in the dsmserv.dsk file for the database and recovery log to record activity. It also identifies storage pool volumes to be used.
- The server starts an ADSM server console session that is used to operate and administer the server until administrative clients are registered to the server.

## 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 66.
2. Query for session information to identify any existing administrative and client node sessions, as described in “Requesting Information about Client Sessions” on page 64.
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 65.
5. Halt the server to shut down all server operations by using the HALT command.

## Restarting the Server

To start the server after it has been halted, follow the instructions in "Starting the Server" on page 61.

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.

## Freeing Links for SNA LU6.2 Client Connections

When a client node initially logs on to an ADSM server by using SNA LU6.2, a SNASVCMG session link is established between the client and the server. This link remains in session even after the user logs off from ADSM. If enough sessions are left connected, new clients can be prevented from connecting to the server.

Because only SNA LU6.2 links must be recycled only after the first time a client logs on and off the system, administrators must deactivate the SNASVCMG link once for each new user. Initially, you may want to recycle links daily until most users have registered with ADSM. After most users have been registered with ADSM, you may want to recycle SNA LU6.2 links less frequently; monthly, for example.

To free unused SNA LU6.2 links, an administrator with root authority must recycle the links as described below.

The server cannot stop the SNASVCMG mode sessions because it does not create them. It is the task of the administrator with root authority to manually deactivate SNASVCMG mode sessions between the server and clients. Because only one SNASVCMG mode session is created for each client, you only need to deactivate the client once. You should deactivate the SNASVCMG sessions on a regular basis to reduce the number of active sessions to zero so that link stations can be recycled.

The administrator with root authority can remove the SNASVCMG sessions by doing the following:

1. On the command line, type **smit sna**.
2. On the first window, click on **Manage SNA Resources**.
3. On the next window, click on **Stop SNA Resources**.
4. On the next window, click on **Stop an SNA Session**.
5. On the next window, select from the list of the conversation group ID, the entry with SNASVCMG mode and the partner LU name.
6. Click on **Do**.

   Repeat the last 2 steps as necessary to deactivate the SNASVCMG mode sessions with other clients.

## 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 114. For information on locking or unlocking client nodes from the server, see “Locking and Unlocking Client Nodes” on page 120.

## 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
```

The following figure shows a sample client session report.

```
  Sess Comm.  Sess     Wait   Bytes   Bytes Sess  Platform Client Name
Number Method State    Time    Sent   Recvd Type
------ ------ ------ ------ ------- ------- ----- -------- --------------------
     3 Tcp/Ip IdleW    9 S    7.8 K     706 Admin OS/2     TOMC
     5 Tcp/Ip IdleW    0 S    1.2 K     222 Admin OS/2     GUEST
     6 Tcp/Ip Run      0 S      117     130 Admin OS/2     MARIE
```

*Figure 4. 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
```

The following figure shows a server background process report after a DELETE FILESPACE command was issued. The report displays a process ID number, a description and a completion status for each background process.

```
 Process Process Description      Status
  Number
-------- ------------------------ ---------------------------------------------
       2 DELETE FILESPACE         Deleting filespace DRIVE_D for node CLIENT1:
                                   172 files deleted.
```

*Figure 5. 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” on page 67 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 adsm/storage/pool001 offline, enter:

```
vary offline adsm/storage/pool001
```

If ADSM encounters a problem with a disk volume, the server automatically varies the volume offline.

After you have replaced the disk volume, you can make it available to the server by varying the volume online. For example, to make the disk volume named adsm/storage/pool001 available to the server, enter:

```
vary online adsm/storage/pool001
```

## Requesting Information about Server Status

Any administrator can request information about the general server parameters defined by SET commands. To query the status of the server, enter:

```
query status
```

ADSM displays information about the server, such as:

- When the server was installed
- Whether the server is enabled or disabled
- Whether client registration is open or closed
- Whether passwords are required for client/server authentication
- How long passwords are valid
- Whether accounting records are being generated
- How long messages remain in the activity log before being deleted
- How many client sessions can concurrently communicate with the server
- How many client node sessions are available for scheduled work
- What percentage of the scheduling startup window is randomized
- What scheduling mode is being used
- How frequently client nodes can poll for scheduled work
- How many times and how frequently a client node can retry a failed attempt to perform a scheduled operation
- How long event records are retained in the database

## Setting the Server Name

| Task | Required Privilege Class |
|---|---|
| Specify the server name | System |

At installation, the server name is set to ADSM. After installation, you can use the SET SERVERNAME command to change the server name. You can use the QUERY STATUS command to see the name of the server.

To specify the server name as WELLS_DESIGN_DEPT., for example, enter the following:

```
set servername wells_design_dept.
```

## Querying Server Options

Any administrator can issue the QUERY OPTION command. Use the QUERY OPTION command to display information about one or more general 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 options name or pattern-matching expression to display information on one or more options in the server options file.

To display general information about all defined server options, enter:

```
query option
```

## Managing the Activity Log

| Task | Required Privilege Class |
|---|---|
| Change the size of the activity log | System or unrestricted storage |
| Set the activity log retention period | System |
| Monitor the activity log | Any administrator |

The activity log contains all messages normally sent to the server console during server operation. The only exceptions are responses to commands entered at the console, such as responses to QUERY commands. Examples of messages sent to the activity log include:

- When client sessions start or end
- When migration starts and ends
- When backup versions are expired
- What data is exported to tape
- When expiration processing is performed
- What export or import processing is performed

Any error messages sent to the server console are also stored in the activity log.

Use the following sections to adjust the size of the activity log, set an activity log retention period, and request information about the activity log.

### Changing the Size of the Activity Log

Because the activity log is stored in the database, the size of the activity log should be factored into the amount of space allocated for the database, 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 server console. If you increase the size of the database or recovery log, ADSM starts activity log recording again. For information about increasing the size of the database or recovery log, see “Step 3: Extending the Capacity of the Database or Recovery Log” on page 87.

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 data 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'
```

General-use programming interface

## 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 AIX accounting file, *dsmaccnt.log*, in the directory from which the server is started. The file contains text records that can be viewed directly or can be read into a spreadsheet such as Lotus 123.

The file remains opened while the server is running and accounting is set on. The file continues to grow until you delete it or prune old records from it. To close the file for pruning, either temporarily set accounting off or bring the server down.

There are 24 fields, which are delimited by commas (,), and each record ends with a new-line character. Each record contains the following information:

| **Field** | **Contents** |
|---|---|
| **1** | Product level |
| **2** | Product sublevel |
| **3** | Product name, ‘ADSM’ |
| **4** | Date of accounting (mm/dd/yyyy) |
| **5** | Time of accounting (hh:mm:ss) |
| **6** | Node name of ADSM client |
| **7** | Client owner name (UNIX) |
| **8** | Client Platform |
| **9** | Authentication method used |
| **10** | Communication method used for the session |
| **11** | Normal server termination indicator (Normal=X'01', Abnormal=X'00') |
| **12** | Number of archive database objects inserted during the session |
| **13** | Amount of archived files, in kilobytes, sent by the client to the server |
| **14** | Number of archived database objects retrieved during the session |
| **15** | Amount of space, in kilobytes, retrieved by archived objects |
| **16** | Number of backup database objects inserted during the session |
| **17** | Amount of backup files, in kilobytes, sent by the client to the server |
| **18** | Number of backup database objects retrieved during the session |
| **19** | Amount of space, in kilobytes, retrieved by backed up objects |
| **20** | Amount of data, in kilobytes, communicated between the client node and the server during the session |
| **21** | Duration of the session, in seconds |
| **22** | Amount of idle wait time during the session, in seconds |
| **23** | Amount of communications wait time during the session, in seconds |
| **24** | Amount of media wait time during the session, in seconds |

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 server and client messages.

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 error 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 error message, such as ANR0992I for example, enter:

```
help 0992
```

# Chapter 4. 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 |

Administrators can manage disk volumes used by the database and recovery log. The sections listed in the following table begin at the indicated pages.

| Section | Page |
|---|---|
| **Concepts:** | |
| Database and recovery log | 75 |
| **Tasks:** | |
| Allocating space for the database and recovery log | 78 |
| Adding space to the database or recovery log | 85 |
| Deleting space from the database or recovery log | 89 |
| Optimizing the performance of the database or recovery log | 94 |

Most tasks presented in this chapter can be performed by using either the graphical user interface or the command-line interface. Table 3 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, refer to *ADSM Administrator's Reference*. For help performing a task on the graphical user interface, see the procedure described in "Using Online Help" on page 20.

## Database and Recovery Log

The database manages information about the location of client backup and archive files residing in storage pools, and it records information necessary for ongoing server operations.

The recovery log is used to maintain a consistent database image by recording changes made to the database as a transaction proceeds. A transaction is an exchange between a client and the server that accomplishes a particular action. For example, defining a management class or copy group is a transaction.

If a transaction completes successfully, database changes are committed, and permanent changes are made to the database. If a transaction does not complete successfully, database changes are undone so that the database changes made by the failing transaction are removed. Because the database and recovery log contain references to transactions, it is important that they are considered together as a single unit.

This section describes the functions of the database and recovery log in more detail, as well as how they interact with each other.

## Database Information

The ADSM server database is used to record information about the location of client backup and archive files. It contains information on:

- Client nodes
- Administrators
- Policies
- Schedules
- Server settings
- Storage volume location for backed up, archived, and client-migrated files

The database also records the information needed for the following server operations:

- Access control information for administrative clients
- Activity logs, which contain the messages generated by the server
- Data storage inventory, used to locate the files that reside in storage pools
- Event records that are generated by the processing of scheduled commands
- Information about registered client nodes
- Information on ADSM volumes
- Policies assigned to the client nodes
- Schedules and their associations with client nodes

## Recovery Log Information

The recovery log is a buffer file within the server that:

- Manages updates that are a result of transactions from clients
- Keeps track of all the updates that have not yet been written to the database
- Plays a key role in restoring the database to a consistent state when a system failure occurs
- Knows which transactions have been completed so that they are copied into the recovered database
- Ensures that incomplete, uncommitted transactions are deleted from the database during the recovery process

## Relationship between the Database and the Recovery Log

Information stored in the server database, recovery log, and storage pool volumes is tightly related. The database and recovery log contain cross-references to units of work called *transactions*. The result of most transactions is that information, files, or instructions are sent to storage pool volumes. It is, therefore, important that all log references to this information be consistent so that the information can be located when it is needed. Examples of transactions include:

- Archiving a client file
- Associating a schedule with a client node
- Backing up a client file
- Defining a management class or copy group
- Deleting a volume from a disk storage pool
- Registering an administrator or a client node

The following is an overview of how the database and recovery log work together when transactions occur:

- A client requests a transaction, for example, a backup for a file.
- A transaction log record is written to the recovery log that describes the action.
- The recovery log associates, or cross-references, the log record with the actual transaction. This process allows the transaction to be rolled back, if necessary, during a recovery procedure.
- The transaction is eventually committed to the database; this process may not happen immediately.
- The transaction record within the recovery log is deleted; it does not necessarily occur the instant the transaction is committed.

“Optimizing the Performance of the Database or Recovery Log” on page 94 has more information on the steps that occur when transactions are processed.

## Storage Pool Volume Locations

The database contains reference information that points to the location of each client file held in storage pool volumes, making it possible to access all client files. As client files are moved, deleted, or migrated across storage volumes, the database is automatically updated to reflect the new location of the files. If the database or recovery log is lost, the data within the storage pools is no longer accessible because the pointer to its location are lost.

## Transaction Processing

When a client sends a transaction to the server, reference information is sent to the recovery log. However, the changes are not immediately made to the database. They are collected and held in a buffer pool and later updated to the database. Because of this, the database and recovery log are not always consistent. For example, when migration takes place, the information is in the recovery log but not immediately written to the database.

This difference between the database and recovery log is a concern only if you have hardware problems or a corruption of the database or storage pool volumes. The recovery process, which is described in Chapter 14, “Recovering Data” on page 329, restores the database to a state in which the database and recovery log are consistent. This state usually coincides with the point of the last database backup. During the recovery process, the server scans the database, ignoring the information in the recovery log and in cache memory. Therefore, any transactions held in the recovery log, but not yet written to the database, are lost.

## Allocating Space for the Database and Recovery Log

During installation, a system programmer allocates space by defining disk volumes to be used by the database and recovery log. The *ADSM Installing the Server and Administrative Client* describes how to allocate the minimum storage space required for the database and recovery log. You can use this minimum storage space to bring up the server and test your client/server environment.

As you register client nodes to the server, consider increasing the size of the database to meet the needs of your installation. With ADSM, you can add or delete storage space to the database or recovery log while the server is running.

Use this section to help you determine how much space you need for the database and recovery log. If you decide to add storage space, use this information to help you prepare disk volumes for database and recovery log storage by completing the following tasks:

1. Estimate the amount of space required for the database
2. Estimate the amount of space required for the recovery log
3. Understand how space is managed by the server
4. Allocate disk space to the database and recovery log

After you have completed these planning tasks, add storage space to the database and recovery log as described in “Adding Space to the Database or Recovery Log” on page 85.

### Step 1: Estimating the Amount of the Database Space

The server uses an internal *database* that stores all information related to its operation. Information stored in the database includes:

- Administrator and client node registration
- Storage management policy definitions
- Backup and archive schedules and associations
- Device class, storage pool, and volume definitions
- Information about backed up, archived, or client-migrated files in storage pools

Remember that the database tracks information *about* backup versions and archive copies. The actual data associated with backed up or archived files is stored in data storage; for example, on disk or tape volumes.

The capacity required for the database is largely affected by the number of files that are backed up or archived, because information about each file is recorded in the database. To estimate the amount of space required for the database, consider how many client nodes are registered to the server and how many files each client node might back up or archive. For information on estimating the amount of data backed up or archived by workstations, see “Estimating Space Needs for Storage Pools” on page 260.

As a general guideline, the initial amount of space allocated to the database should be between 1% to 5% the total amount of space required for data storage. For example, if your installation requires 10GB of data storage, then the size of your database should be between 100MB to 500MB.

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

## Step 2: Estimating the Amount of Recovery Log Space

The server uses the *recovery log* to keep a record of all changes made to the database. When changes occur, the server updates the recovery log before it updates the database. If a system failure occurs, the server uses the information in the recovery log to roll back uncommitted transactions from the database, ensuring that information in the database is returned to a consistent state.

Because the server supports concurrent client sessions, numerous updates can occur at the same time. For example, whenever a file is backed up or archived, a number of records are added to the recovery log. Consequently, the size of the recovery log is dependent on the number of concurrent client sessions communicating with the server, rather than the number of files in data storage. The number of concurrent sessions and the number of background processes executing on the server determine the number of transactions.

The number of concurrent client sessions cannot be greater than the number of maximum client sessions, as defined by the MAXSESSIONS option in the server options file, as described in the *ADSM Installing the Server and Administrative Client*.

Initially, begin with at least 12MB for the recovery log. Then monitor the utilization of the recovery log to determine whether you should increase or decrease the size of the recovery log.

**Note:** The recovery log must be large enough to store additional recovery log records. Operating in roll-forward mode significantly increases recovery log storage requirements. The extent of the increase is determined by the number of database transactions since the last database backup. In roll-forward mode, the recovery log tracks all transactions since the last database backup. See "Database Backup and Recovery" on page 334 for details.

## Step 3: Understanding How Space is Managed by the Server

ADSM tracks all volumes defined to either the database or the recovery log as one *logical volume*. To determine how much space is available in each logical volume, query the database or recovery log.

To request information about the database, enter:

```
query db
```

The server displays a report, similar to Figure 6 on page 80.

```
Available Assigned   Maximum   Maximum    Page     Total      Used %Util  Max.
    Space Capacity Extension Reduction    Size     Pages     Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)
--------- -------- --------- --------- ------- --------- --------- ----- -----
      196       96       100        92   4,096    24,576        86   0.3   0.3
```

*Figure 6. Information about the Database*

To request information about the recovery log, enter:

```
query log
```

The server displays a report, similar to Figure 7.

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

*Figure 7. Information about the Recovery Log*

When you query the database or query the recovery log, ADSM displays the following space utilization information:

- Available space
- Assigned capacity
- Utilization
- Maximum percent utilized

Use this information to determine whether you need to allocate more disk space and define additional volumes to the database and recovery log.

## Available Space

*Available space* is the amount of space, in megabytes, that is available to the database and recovery log. You can use this available space to:

- Extend the capacity of the database or recovery log
- Provide sufficient free space before you try to delete a volume from the database or recovery log

The server determines available space by adding the amount of usable space from all volumes defined to the database or recovery log. ADSM calculates the amount of usable space by:

- Subtracting 1MB from each volume for overhead processing
- Dividing the remaining space into 4MB partitions. Any remaining megabytes result in unused space on the volume

The following examples illustrate how you can efficiently allocate disk space so that most of the space is available to the database and recovery log.

***An Example of Poor Use of Disk Space:*** If you allocate four 20MB volumes for the recovery log, the server:

- Subtracts 1MB from each volume for overhead processing so that each volume has 19MB of available space
- Divides the remaining space (19MB) into 4MB partitions so that each volume contains four 4MB partitions

  In this case, the server uses 16MB of the remaining 19MB from each volume and leaves 3MB of unused space on every volume.

In this example, only 64MB of the initial 80MB are available to the recovery log.

***An Example of Better Use of Disk Space:*** If you allocate four 25MB volumes for the database, the server:

- Subtracts 1MB from each volume for overhead processing so that each volume has 24MB of available space
- Divides the remaining space (24MB) into 4MB partitions so that each volume contains six 4MB partitions

  In this case, the server uses all 24MB of the remaining 24MB in each volume.

In this example, 96MB of the initial 100MB are available for use by the database.

Figure 8 shows that the available space for the database logical volume is 96MB.

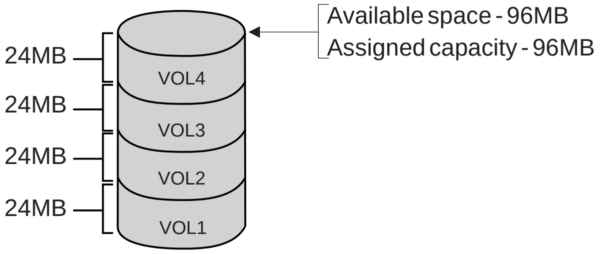


*Figure 8. An Example of Available Space and Assigned Capacity*

## Assigned Capacity

*Assigned capacity* is the portion of available space that can be used to store database or recovery log information. During installation, the server automatically extends the database and recovery log to match the available space, as shown in Figure 8 on page 81.

After installation, if you decide to add additional volumes, then you must extend the database or recovery log to increase the assigned capacity. For information on extending the capacity of the database or recovery log, see “Step 3: Extending the Capacity of the Database or Recovery Log” on page 87.

## Utilization

As data is added to the database or recovery log, the server tracks the percentage of utilization. *Utilization* is the percent of assigned capacity used by the database or recovery log at a specific point in time.

During the day, the utilization percentage varies. The *maximum percent utilized* is the highest percent of assigned capacity used by the database or recovery log.

For example, Figure 9 shows that 80% of a recovery log is utilized during the most active period of the day.

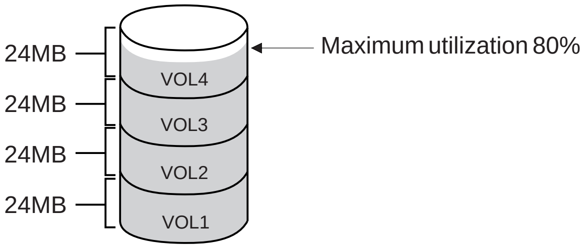


*Figure 9. An Example of Utilization*

## Monitoring Utilization

The maximum amount of space used by the recovery log can vary significantly throughout the day because it is proportional to the transaction load on the system. For example, if many backups occur in the morning, the amount of space required by the recovery log increases during this period.

The maximum amount of space used by the database is more consistent with the utilization percentage because the amount of database space consumed is proportional to the number of objects inserted into or deleted from the database. The utilization percentage and maximum percent utilized tend to be the same, unless a large number of objects are deleted from the database.

Database and recovery log utilization percentages are reset automatically whenever the server is restarted. You can also adjust the size of your database or recovery log by using the RESET DBMAXUTILIZATION and RESET LOGMAXUTILIZATION commands to meet your storage needs.

To monitor the utilization of the database or recovery log on a continuous basis, you can reset the maximum utilization counters. In this way, you can monitor how much space is being utilized every day.

For example, you might initially want to reset database and recovery log utilization statistics on a daily basis to set the maximum utilization percentage equal to the current utilization.

To reset the maximum utilization statistic for the database, enter:

```
reset dbmaxutilization
```

To reset the maximum utilization statistic for the recovery log, enter:

```
reset logmaxutilization
```

## Step 4: Allocating Space for the Database and Recovery Log

Before you allocate disk space for the database or recovery log, consider:

- How the placement of volumes affects the availability of the database and recovery log
- How the placement of volumes affects the performance of the database and recovery log
- How the size of volumes affects space utilization

Then allocate space on disk volumes for use by the database and recovery log.

### Ensuring the Availability of the Database and Recovery Log

To protect database and recovery log volumes from media failure, you must use the mirroring feature to protect the server from media failure. In addition, you want to place database and recovery log mirror volumes on separate physical disks. If you cannot assign each volume to its own HDA, assign database and recovery log volumes to separate physical disks. See “Database and Recovery Log Mirroring” on page 330 for information on the mirroring feature.

### Improving the Performance of the Database and Recovery Log

To improve performance, define more than one volume for the database and recovery log, and place these volumes on separate disks to allow simultaneous access to different parts of the database or recovery log.

When possible, assign each database and recovery log volume to separate DASD strings, preferably on DASD with different control units and channels.

## Using Space Efficiently

To use disk space efficiently, allocate a few large disk volumes rather than many small disk volumes. You want to allocate a few large disk volumes to avoid losing space to ADSM overhead processing. See “Available Space” on page 80 for information on allocating space more efficiently.

## Allocating Database or Recovery Log Space on Disk Storage

To allocate space on disk storage for the database and recovery log, complete the following operating system specific tasks:

1. Run the DSMFMT utility to allocate disk storage space for the database and recovery log.

   **Note:** DSMFMT is not a server command. It runs in a separate AIX session.

   In the following example, 4MB of space is allocated for the *db.2* database volume:

```
% dsmfmt -db db.2 4
```

   *Output*:

```
ADSTAR Distributed Storage Manager
AIX ADSM Server DSMFMT Extent/Volume Formatting Program

Licensed Materials - Property of IBM

5765-564 (C) Copyright IBM Corporation 1990, 1995.  All rights reserved.
U.S. Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corporation.

Actual allocation for db.2 will be 5 MB
Allocated space for db.2: 5242880 bytes
```

2. After the space has been allocated, define the database volume by issuing the DEFINE DBVOLUME command from the server as shown in the following example:

```
adsm>
define dbvolume db.2
```

   *Output*:

```
ANR2240I Database volume /usr/lpp/adsmserv/bin/db.2 defined.
```

3. After defining the database volume, increase the assigned capacity of the database by issuing the EXTEND DB command from the server as shown in the following example:

```
adsm>
extend db 4
```

*Output*:

```
ANR2248I Database assigned capacity has been extended.
```

After you have allocated space to disk volumes, you can add them to the database or recovery log as described in “Adding Space to the Database or Recovery Log.”

## Adding Space to the Database or Recovery Log

**Attention:** The size of an allocated database, recovery log, or storage pool volume cannot be changed after it has been defined to the ADSM server. ADSM uses the initial size allocation of the volume at the time it is defined to the server to calculate data placement for later retrieval. If you change the size of ADSM volumes by extending raw logical volumes through smit or otherwise altering the file sizes of ADSM volumes, ADSM may not initialize correctly and data may be lost.

To add space to the database or recovery log, you must first define a volume and then extend the capacity of the database or recovery log.

You can add space to the database or the recovery log while the server is running. Use this section to help you:

1. Define disk volumes to the database or recovery log
2. Determine the maximum extension of the database or recovery log
3. Extend the capacity of the database or recovery log

### Step 1: Defining Disk Volumes

During installation, you initially defined volumes for your database and recovery log. At any time you can increase the size of the database and recovery log by adding more volumes. Each time you define a volume to the database or recovery log, you increase the available space without interrupting server operations. You can use this space to:

- Extend the capacity of the database or recovery log to store additional database or recovery log records, as described in “Step 3: Extending the Capacity of the Database or Recovery Log” on page 87
- Provide sufficient space to delete a volume from the database or recovery log, as described in “Step 4: Deleting a Volume from the Database or Recovery Log” on page 93

For example, define a 101MB disk volume name VOL5 to the database by entering:

```
define dbvolume vol5
```

When VOL5 is defined, the server tracks the volume as part of the database logical volume, as shown in Figure 10. 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.

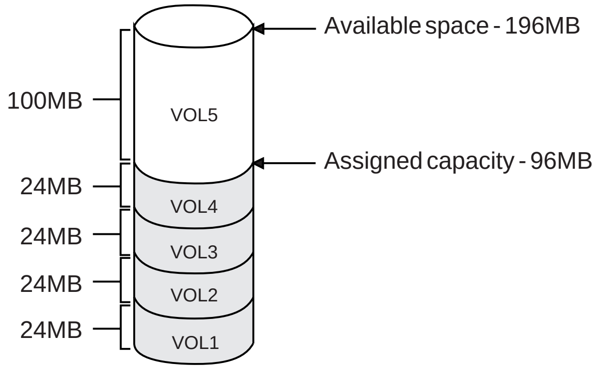


*Figure 10. Adding Volumes Increases Available Space*

To define a recovery log volume named *log2*, enter:

```
define logvolume log2
```

## Step 2: Determining the Maximum Extension

After you define volumes to the database or recovery log, determine how much the database or recovery log can be extended.

The *maximum extension* is determined by subtracting the assigned capacity from the available space. For example, Figure 11 on page 87 shows that the maximum extension of the database is 100MB.

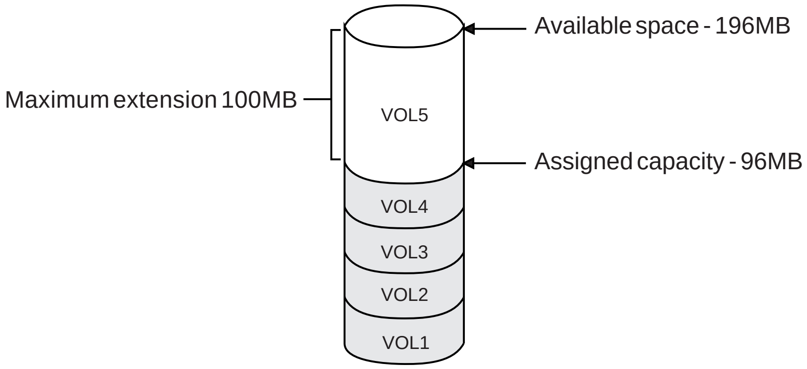
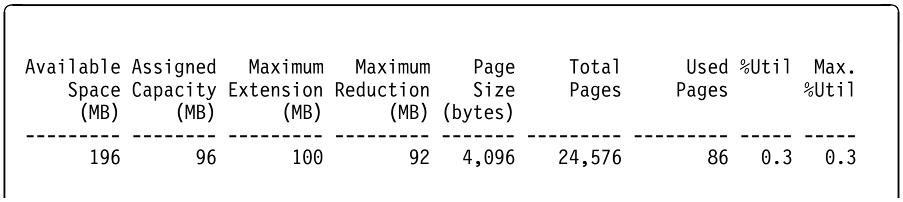


*Figure 11. An Example of Maximum Extension*

To determine the maximum extension of the database, enter:

```
query db
```

Figure 12 displays a standard database report, which shows that the available space is 196MB, the assigned capacity is 96MB, and the maximum extension is 100MB.



```
Available Assigned   Maximum   Maximum     Page     Total      Used %Util  Max.
    Space Capacity Extension Reduction     Size     Pages     Pages       %Util
     (MB)     (MB)      (MB)      (MB)  (bytes)
--------- -------- --------- --------- ------- --------- --------- ----- -----
      196       96       100        92   4,096    24,576        86   0.3   0.3
```

*Figure 12. A Database Report to Determine Maximum Extension*

## Step 3: Extending the Capacity of the Database or Recovery Log

To increase the capacity of 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, then ADSM rounds the number to the next higher 4MB partition. For example, if you extend the recovery log by 1MB, ADSM extends the capacity of the recovery log by 4MB.

For example, to increase the capacity of the database by 100MB, enter:

```
extend db 100
```

To increase the capacity of the recovery log by 4MB, enter:

```
extend log 4
```

When you extend the database or recovery log, ADSM starts a background process to format the new space. When the background process completes, the capacity of the database is increased by 100MB, as shown in Figure 13.

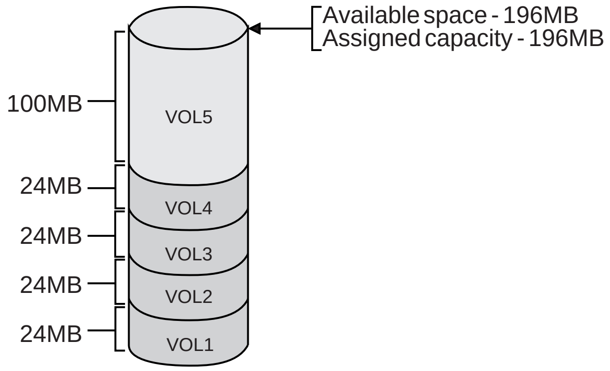


*Figure 13. Extending the Capacity of the Database*

You can query the server to see the assigned capacity of the database or recovery log. For example, to view the assigned capacity of the database, enter:

```
query db
```

After you extend the database, the available space is 196MB, the assigned capacity is 196MB, and the maximum extension is 0MB, as shown in Figure 14.

```
Available Assigned   Maximum   Maximum    Page     Total      Used %Util  Max.
    Space Capacity Extension Reduction    Size     Pages     Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)
--------- -------- --------- --------- ------- --------- --------- ----- -----
      196      196         0       192   4,096    50,176       111   0.2   0.2
```

*Figure 14. A Database Report to Determine Assigned Capacity*

After the capacity of the database or recovery log is extended, additional data can be added to the database or recovery log.

## Deleting Space from the Database or Recovery Log

You should delete volumes from the database or recovery log only if there is sufficient space on other volumes within the database or recovery log. To determine whether there is sufficient space to delete a volume, complete the following steps:

1. Determine the size of the volumes you want to delete.

   Request a detailed database or recovery log volume report to view the available space, allocated space, and free space on each database or recovery log volume, as described in “Step 1: Determining the Size of Volumes.”

2. Determine whether the database or recovery log has sufficient *free space* on which to store information from the volume you want to delete, as described in “Step 2: Determining If There Is Sufficient Free Space” on page 91.

3. If you do not have sufficient free space, do one of the following:

   - Add more space to the database or recovery log by defining additional volumes, as described in “Step 1: Defining Disk Volumes” on page 85
   - Reduce the capacity of the database to free up existing space in the database or recovery log, as described in “Step 3: Reducing the Capacity of the Database or Recovery Log” on page 92

4. Delete volumes from the database or recovery log, when the amount of available space is equal or greater to the size of the volumes you want to delete, as described in “Step 4: Deleting a Volume from the Database or Recovery Log” on page 93.

### Step 1: Determining the Size of Volumes

Any administrator can request detailed information about volumes defined to the database or recovery log. Detailed reports are useful for determining how much space has been allocated and used by an individual volume in the database or recovery log.

For example, to display a detailed volume report about volumes defined to the database, enter:

```
query dbvolume format=detailed
```

Figure 15 on page 90 displays a detailed report about volumes defined to the recovery log. This example shows that VOL4, VOLD, and VOL300 are a group of mirrored volumes, and VOL5, VOLE, and VOL200 are a group of mirrored volumes.

This example also shows that there is no free space available on any of these volumes because all available space has been allocated to the database.

```
Volume Name (Copy 1): VOL4
         Copy Status: Sync'd
Volume Name (Copy 2): VOLD
         Copy Status: Sync'd
Volume Name (Copy 3): 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
```

*Figure 15. Information about Database Volumes*

To display a detailed volume report about the log volume defined to the database, enter:

```
query logvolume format=detailed
```

Figure 16 displays a detailed report about the recovery log volume.

```
Volume Name (Copy 1): /home/bill/dsmserv/build/log.1
         Copy Status: Sync'd
Volume Name (Copy 2):
         Copy Status: Undefined
Volume Name (Copy 3):
         Copy Status: Undefined
Available Space (MB): 8
Allocated Space (MB): 8
     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
```

*Figure 16. Information about Database Volumes*

## Step 2: Determining If There Is Sufficient Free Space

To determine if there is sufficient free space to delete a volume from the database or recovery log, request information about the database or recovery log.

For example, to determine how much free space is available in the database, enter:

```
query db
```

Figure 17 displays a standard database report.

```
Available Assigned   Maximum   Maximum    Page     Total      Used %Util  Max.
    Space Capacity Extension Reduction    Size     Pages     Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)
--------- -------- --------- --------- ------- --------- --------- ----- -----
      196      196         0       176   4,096    50,176     4,755   9.5   9.5
```

*Figure 17. Information about the Database*

From this report, you can determine if there is sufficient free space in one of two ways:

- Subtract the assigned capacity from the available space.

  In this example, 196MB – 196MB = 0.

- Determine how much assigned capacity is not being used to store data as shown by the maximum reduction column.

The server automatically calculates the space actually used and reports the amount of free space in the *Maximum Reduction* column.

In this example, the maximum reduction is 176MB, as shown in Figure 18 on page 92.

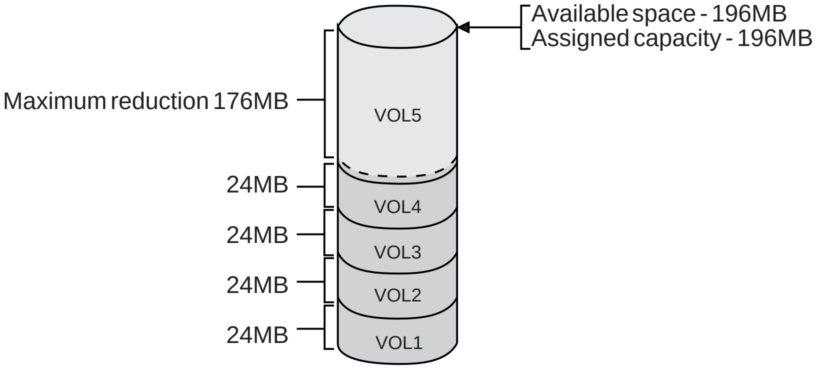


*Figure 18. Understanding Maximum Reduction*

## Step 3: 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 allow you to delete a volume from the database or recovery log.

You can reduce the capacity of the database or recovery log in 4MB increments. If you do not reduce by increments of 4MB, then ADSM rounds up to the next higher 4MB partition. For example, if you enter `reduce db 5`, ADSM reduces the recovery log by 8MB.

For example, to reduce the database by 100MB, enter:

```
reduce db 100
```

To reduce the recovery log by 16MB, enter:

```
reduce log 16
```

When you issue a reduce command, the server begins reclaiming space from the last volume to which data was added. If necessary, it relocates data within the defined space. Depending on how much data is relocated, this process can take a long time. When this happens, the reduce operation is run as a background process.

You can query the server for information about the database or recovery log to determine how much free space is available after reduction. For example, after reducing the database by 100MB, the assigned capacity is 96MB, the maximum extension is 100MB, and the maximum reduction is 92MB, as shown in Figure 19 on page 93.

```
Available Assigned   Maximum   Maximum    Page     Total      Used %Util  Max.
    Space Capacity Extension Reduction    Size     Pages     Pages       %Util
     (MB)     (MB)      (MB)      (MB) (bytes)
--------- -------- --------- --------- ------- --------- --------- ----- -----
      196       96       100        92   4,096    24,576        86   0.3   0.3
```

*Figure 19. Information about the Database*

## Step 4: Deleting a Volume from the Database or Recovery Log

After you reduce the database or recovery log, try using the smaller size for a few days. If the maximum utilization percentage does not go over 70%, then you can delete extra volumes from the database or recovery log.

ADSM does not allow you to delete volumes if you do not have sufficient space to store the existing data in the database or recovery log. In addition, ADSM does not allow you to delete the last remaining volume of the database or recovery log.

However, when there is sufficient free space on other volumes, you can delete volumes from the database or recovery log. This process allows you to consolidate data from partially used volumes so that you can use disk space efficiently.

For example, after you reduce the database by 100MB, you determine that you can delete four 24MB volumes from the database.

To delete volumes 1 through 4 from the database, enter the following commands:

```
delete dbvolume vol1
delete dbvolume vol2
delete dbvolume vol3
delete dbvolume vol4
```

To delete the log.1 recovery log volume, enter:

```
delete logvolume log.1
```

When you delete volumes from the database or recovery log, the server moves existing data to available space on other volumes. In this example, as shown in Figure 20 on page 94 , data is moved from volumes 1 through 4 to available space on VOL5. When all data is moved, these volumes are deleted from the server.

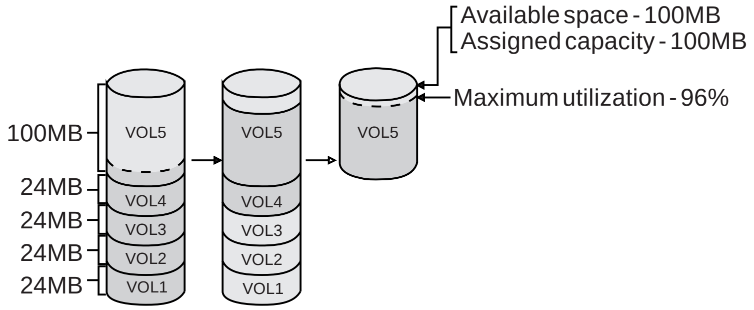


*Figure 20. Deleting a Database Volume*

## Optimizing the Performance of the Database or Recovery Log

When a transaction occurs, the server reads or writes pages to database and recovery log buffer pools. By monitoring the database and recovery log buffer pools, you can optimize the performance of the database and recovery log.

### Database Buffer Pool

When a transaction occurs, one or more database pages may require updating. Each *page* is a 4096-byte block that remains in the buffer pool until space is needed by another page.

An administrator can update the size of the database buffer pool to optimize the I/O performance of the database by updating the BUFPOOLSIZE in the server options file. “Monitoring the Database Buffer Pool” on page 95 provides information about increasing the size of the database buffer pool.

### Recovery Log Buffer Pool

Because the server supports concurrent client sessions, numerous transactions can occur at the same time. Consequently, the recovery log can be updated continuously. To support concurrent transactions, the recovery log holds transaction log records in a log buffer.

You can update the size of the log buffer to optimize the I/O performance of the recovery log by updating the LOGPOOLSIZE in the server options file. “Monitoring the Recovery Log Buffer Pool” on page 96 provides information about increasing the size of the recovery log buffer pool.

Transaction log records remain in the log buffer until the active buffer becomes full or ADSM forces log records to the recovery log.

When all log records for a transaction are written to the recovery log, the updates are committed to the database. Then the recovery log can be used to roll back the transaction's updates if there is a system failure or if the transaction does not complete successfully.

## Monitoring the Database Buffer Pool

The *database buffer pool* provides cache storage, which allows database pages to remain in memory for longer periods of time. When database pages remain in cache, the server can make continuous updates to the pages without requiring I/O operations to external storage. While a large database buffer pool can improve server performance, it also requires more memory.

A system administrator can control the size of the database buffer pool by updating the BUFPOOLSIZE option in the server options file. At installation, the database buffer pool is set to 512KB, which equals 128 database pages. For information about setting server options, see *ADSM Installing the Server and Administrative Client*.

### Requesting Information about the Database Buffer Pool

To evaluate whether the current size of the database buffer pool is adequate for database performance, request a detailed database report by entering:

```
query db format=detailed
```

Figure 21 displays a detailed database report.

```
  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
```

*Figure 21. Detailed Information about the Database*

Use the following fields to evaluate your current use of the database buffer pool:

**Buffer Pool Pages**

Specifies the number of pages in the database buffer pool. This value is determined by the value set for the BUFPOOLSIZE option in the server options file.

**Total Buffer Requests**

Specifies the cumulative number of requests for database pages since the server was last started or since the last reset of the buffer pool.

**Cache Hit Pct**

Specifies, as a percentage, the number 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 drops below 90%, consider increasing the size of the database buffer pool.

**Cache Wait Pct**

Specifies, as a percentage, the number 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 above 0, increase the size of the database buffer pool.

### Resetting Database Buffer Pool 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 on a weekly basis.

For example, to reset the database buffer pool, enter:

```
reset bufpool
```

## Monitoring the Recovery Log Buffer Pool

The *recovery log buffer pool* is used to hold new transaction records until they can be written to the recovery log. The size of the recovery log buffer pool can affect how frequently the server forces records to the recovery log.

A system administrator can control the size of the recovery log buffer pool by updating the LOGPOOLSIZE option in the server options file. At installation, the default setting is 128KB, which equals 32 recovery log pages. For information about setting server options, see *ADSM Installing the Server and Administrative Client*.

## Requesting Information about the Recovery Log Buffer Pool

To determine how the buffer pool size affects recovery log performance, query the recovery log for a detailed report by entering:

```
query log format=detailed
```

Figure 22 displays a detailed report.

```
  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
```

*Figure 22. Detailed Information about the Recovery Log*

Use the following fields to optimize the log buffer pool size for your installation:

**Log Pool Pages**
Specifies the number of pages in the recovery log buffer pool. This value is determined by the value set for the LOGPOOLSIZE option in the server options file.

**Log Pool Pct. Util**
Specifies, as a percentage of all recovery log buffer pool pages, the number of pages used to write changes to the recovery log after a transaction is committed.

A low *log pool percent utilization* indicates that the size of your recovery log buffer pool is adequate. As this number grows larger, consider increasing the size of the recovery log buffer pool.

**Log Pool Pct. Wait**
Specifies, as a percentage of all recovery log buffer pool pages, the number 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 zero, increase the size of the recovery log buffer pool.

# Chapter 5. 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 | 99 |
| Ensuring client/server authentication | 104 |
| Registering administrators or updating information | 105 |
| Granting administrative authority | 106 |
| Revoking or reducing administrative authority | 110 |
| Managing administrator access | 113 |
| Managing client node registration | 115 |
| Registering an application programming interface to the server | 124 |
| Managing client node access | 119 |
| Requesting information about client nodes or file spaces | 120 |
| Deleting client data and client nodes from the server | 123 |

Most tasks presented in this chapter can be performed using either the graphical user interface or the command-line interface. Table 4 on page 38 shows whether a task can be performed on the graphical user interface, the command line interface, or both.

For information about issuing ADSM commands, refer to *ADSM Administrator's Reference*. For help performing a task on the graphical user interface, see the procedure described in “Using Online Help” on page 20.

## Managing ADSM Licenses

| Task | Required Privilege Class |
|---|---|
| • Register licenses<br>• Audit licenses<br>• 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 about either case, see “License Compliance” on page 103. In either case, you must contact your IBM account representative or authorized reseller to modify your agreement.

## Licensed Features

The base ADSM AIX server license supports an unlimited number of administrative clients, one AIX 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 AIX backup-archive client. If you want to add clients in an environment other than AIX, you must register a new license for that feature also (see the next item in this list). See “Additional Clients.”
- Add support for clients in environments other than AIX. The base license allows only for backup-archive clients on AIX. See “Clients Other Than AIX” on page 101.
- Add support for Disaster Recovery Manager. See Chapter 15, “Using Disaster Recovery Manager” on page 359.
- Add support for storage devices not covered by the existing agreement. See “Device Support Modules 1-4” on page 102.
- Add a secondary server. See “Secondary Server Attachment” on page 102.
- Upgrade device support. See “Device Support Module Upgrades” on page 102.

When you order ADSM features or device support modules from an IBM representative or authorized reseller, you receive one or more license authorization codes. These codes are included with the feature or device support module package.

A license authorization code is a character string unique to each feature or device support module package. You specify the code when you register the license to the server.

**Note:** License authorization codes included in this section are only examples and will not work on your server.

### Additional Clients

You can register the server to support a specified number of clients beyond the one AIX backup-archive client supported by the base license. Those additional clients can be in any environment for which your system is licensed (see “Clients Other Than AIX” on page 101). The following client registration packs are available:

- 1 client
- 5 clients
- 10 clients
- 50 clients

Each type of registration pack has a unique license authorization code.

For example, to register three additional clients, obtain three 1-client licenses. Then register each client separately by using the authorization code for the 1-client registration pack:

```
register license aaaaaaaaaaaaaaaa
```

Where:

*aaaaaaaaaaaaaaaa* is the authorization code

However, to register ten additional clients, obtain ten 1-client licenses and the license authorization code for the 10-client registration pack. Then register all clients at once by using the authorization code for the 10-client registration pack:

```
register license cccccccccccccccc
```

Where:

*cccccccccccccccc* is the authorization code

**Note:** If you register more clients than your server is licensed to support, the server issues a warning message. However, operations continue normally.

## Clients Other Than AIX

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

Environment support features are:

**DESKTOP**
DOS, Macintosh, NetWare, OS/2, or Windows

**UNIX**
AT&T, DEC ULTRIX, HP-UX, SCO UNIX 386, SCO Open Desktop, SINIX-Z, SunOS or Solaris, or UNIX

**OPENEDITION**
Open Edition for MVS

**SPACEMGMT**
HSM clients on AIX, HP-UX, SunOS or Solaris

Each feature has a unique license authorization code.

You can register any or all environment support features. For example, to add one or more HP–UX clients:

1. Issue the REGISTER LICENSE command with the UNIX environment support license authorization code.

2. Issue one or more REGISTER LICENSE commands with the license authorization code for the appropriate client registration pack.

## Device Support Modules 1-4

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.

For example, to let the server attach an optical library, obtain the license authorization code for Device Support Module 1 (or higher). Then register Device Support Module 1 by using the code.

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.

## 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 issuing the REGISTER LICENSE command with the authorization code for secondary server attachment.

## Device Support Module Upgrades

To attach a storage device that is listed in a higher-numbered device support module than you currently have either obtain the higher-numbered device support module or a device support module upgrade. The following upgrades are available:

- Device Support Module upgrade: Module 1 to 2
- Device Support Module upgrade: Module 2 to 3
- Device Support Module upgrade: Module 3 to 4

Each upgrade has a unique license authorization code.

For example, to upgrade from the base module to Device Support Module 3, enter:

```
register license mmmmmmmmmmmmmmmm
register license nnnnnnnnnnnnnnnn
```

Where:

mmmmmmmmmmmmmmmm
: is the authorization code for Device Support Module upgrade 1 to 2

nnnnnnnnnnnnnnnn
: is the authorization code for Device Support Module upgrade 2 to 3

## 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.

## Licensing Example

You can register a license with an ADSM server by using the REGISTER LICENSE command specifying a **license authorization** code 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. The license authorization codes are included with each feature or device support module package.
2. Issue a REGISTER LICENSE command for each feature or device support module.

Always save your original license authorization codes for each licensed feature and device support module. You must have the codes if you need to register your license again for any of the following reasons:

- The server is corrupted.
- The server is moved to a different machine.
- The *adsmserv.licenses* file is destroyed or corrupted. ADSM stores license information in the *adsmserv.licenses* file, which is located in the directory from which the server is started.

## 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.

## Trial License

When purchased from IBM, ADSM includes a *dsmreg.lic* file, that the server uses to access and interpret licenses. This file is located in the */usr/adsmserver/bin* directory. If you start the server from a directory other than */usr/adsmserv/bin*, you must use the DSMSERV_DIR environment variable to specify the directory where the *dsmreg.lic* file resides. For more information on defining environment variables, refer to *ADSM Installing the Server and Administrative Client*.

If, for any reason, the server cannot access the *dsmreg.lic* file, the license manager of ADSM assumes that the server is not licensed. However, a trial license for one AIX client and all device support modules is supported for 60 days after installation. If the server cannot access the *dsmreg.lic* file after that time, all server operations fail.

## Ensuring Client/Server Authentication

| Task | Required Privilege Class |
|---|---|
| Set password authentication<br>Set password expiration | System |

ADSM provides client/server authentication to validate that administrative clients or client nodes are communicating with an authorized server, and that the server is communicating with registered administrators and client nodes. ADSM ensures client/server authentication by requiring a password from each administrative client or client node registered with a server.

You can ensure client/server authentication by:

- Requiring users to enter a password to access the server
- Requiring users to change their passwords regularly

## Setting Client 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. ADSM maintains a list of registered client passwords for authentication.

To allow administrators and client nodes to access ADSM without entering a password, you must set password authentication to off by issuing the SET AUTHENTICATION command.

**Attention:** Setting password authentication off, reduces data security.

### Setting User Password Expiration

You can set a password expiration period for administrators and client node users by issuing the SET PASSEXP command. The valid password expiration period is from 1 to 9999 days. At installation, ADSM sets a password expiration of 90 days.

When an administrator or client node user is first registered to the server, ADSM begins tracking the password expiration period. For example, when using the installed value, users must change their password within 90 days of the password expiration period. For example, when using the installed value, users must change their password within 90 days after they are 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 attempts 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 for the user. You also can provide contact information such as the user name and telephone number. This contact information is displayed when administrators query for 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'
```

After users are registered as administrators, they can display ADSM server information from any computer on which the administrative client program is installed.

**Note:** At installation, the server console is defined with a special user ID, which is named SERVER_CONSOLE. This name is reserved and cannot be used by another administrator. At installation, the SERVER_CONSOLE user ID is granted system privilege so that other administrators can be registered and granted system privilege.

Another administrator with system privilege can revoke or grant new privileges to the SERVER_CONSOLE user ID. However, you cannot update, lock, rename, or remove the SERVER_CONSOLE user ID from ADSM.

See the *ADSM Installing the Server and Administrative Client* for information on using the SERVER_CONSOLE user ID.

If administrator DAVEHIL forgets his password, a system administrator can update the password so that DAVEHIL can access the ADSM server again. To change the password for administrator ID DAVEHIL to *ganymede*, enter:

```
update admin davehil ganymede
```

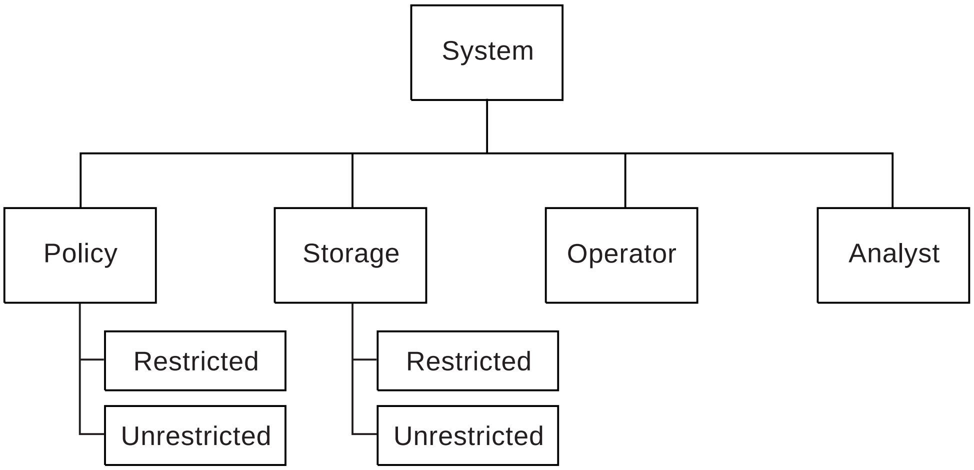
**Note:** The SERVER_CONSOLE administrator's user ID and contact information cannot be updated.

## Granting Administrative Authority

| Task | Required Privilege Class |
|---|---|
| Grant authority to other administrators | System |

After administrators have been 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 the following figure. An administrator with system privilege can perform any ADSM function. Administrators with policy, storage, operator, or analyst privileges can perform a subset of ADSM functions.



Whenever you issue a subsequent GRANT AUTHORITY command, ADSM adds authority to the administrator; it does not override existing authority.

## System Privilege

An administrator with *system privilege* can perform any ADSM administrative task. To perform the following tasks you must have 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

In addition, an administrator with system privilege can grant other administrators system privilege by specifying an administrator user ID and the system privilege class.

For example, to grant the system privilege class to administrator KACZ, enter:

```
grant authority kacz classes=system
```

## Unrestricted Policy Privilege

An administrator with *unrestricted policy privilege* has the authority to 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
- 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
```

However, SMITH cannot copy, define, or delete policy domains.

## Restricted Policy Privilege

An administrator with *restricted policy privilege* is authorized to issue commands only for the policy domains to which they have been authorized.

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 SMITH enter:

```
grant authority smith domains=engpoldom
```

However, DAVEHIL (as established in “Registering Administrators or Updating Information” on page 105) cannot copy a management class from the engineering policy domain to the standard policy domain because the administrator does not have authority over the policy domain named STANDARD.

## 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

To grant unrestricted storage privilege to administrator COYOTE, enter:

```
grant authority coyote classes=storage
```

## Restricted Storage Privilege

Administrators with *restricted storage privilege* can issue a subset of the storage commands only for the storage pools for 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

To grant restricted storage privilege for existing storage pools beginning with the name ADSM to administrator HOLLAND, enter:

```
grant authority holland stgpools=adsm*
```

Administrator HOLLAND is restricted to managing storage pools beginning with ADSM that existed at when the command was first issued; for example, ADSM.BFS.TAPE1, ADSM.BFS.TAPE2, and so on. HOLLAND is not authorized to manage any *new* 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 HOLLAND, enter:

```
grant authority holland classes=operator
```

Now, HOLLAND has operator privilege so that he can manage tape operations, as well as having restricted storage privilege over tape storage pools.

## 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
```

# Revoking or Reducing Administrative Authority

| Task | Required Privilege Class |
|---|---|
| Revoke or reduce administrative privilege classes | System |

You can remove all or part of another administrator's authority. You can reduce administrative authority by revoking:

- The system privilege class from an administrator and then granting one or more of the following privilege classes: storage, policy, operator, or analyst

- Unrestricted policy or storage privileges and then granting restricted policy or storage privileges to the administrator
- Authority over some policy domains from an administrator with restricted policy privilege
- Authority over some storage pools from an administrator with restricted storage privilege

## Revoking All Administrative Privilege Classes

You can revoke all administrative privilege classes from another administrator so that the latter administrator is only allowed to perform those functions that can be performed by any administrators.

To revoke all administrative privilege classes from an administrator, identify the administrator user ID, but do not specify any privilege classes, policy domains, or storage pools. For example, to revoke both the storage and operator privilege classes from administrator HOLLAND issue the REVOKE AUTHORITY command:

```
revoke authority holland
```

HOLLAND can still query the server for information, but he can no can no longer issue administrative commands.

## 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 administrator MARYSMITH no longer has the time to manage the database, recovery log, or disk and tape storage pools. To remove part of her administrative authority by revoking the storage privilege class, enter:

```
revoke authority marysmith classes=storage
```

## Reducing System Authority

You can reduce the authority of another system administrator by revoking the system privilege class and granting one or more of the remaining privilege classes.

For example, to reduce the authority of administrator SERVER_CONSOLE to the operator privilege class do the following:

1. Revoke the system privilege class from administrator SERVER_CONSOLE by entering:

```
revoke authority server_console classes=system
```

2. Grant operator privilege class to administrator SERVER_CONSOLE by entering:

```
grant authority server_console classes=operator
```

## Reducing Unrestricted Policy to Restricted Policy Privilege

You can reduce an administrator's unrestricted policy privilege by revoking the policy privilege class and then granting restricted policy privilege over specified policy domains.

For example, to reduce administrator DSMITH's unrestricted policy privilege to restricted policy privilege for the STANDARD policy domain, do the following:

1. Revoke the unrestricted policy privilege class from DSMITH by entering:

```
revoke authority dsmith classes=policy
```

2. Grant restricted policy privilege over the STANDARD policy domain by entering:

```
grant authority dsmith domains=standard
```

## Reducing Unrestricted Storage to Restricted Storage Privilege

You can reduce an administrator's unrestricted storage privilege by revoking the storage privilege class and granting restricted storage privilege over specified storage pools.

For example, to reduce administrator COYOTE from unrestricted storage privilege to restricted storage privilege for the storage pools named BACKUPPOOL and ARCHIVEPOOL, do the following:

1. Revoke the unrestricted storage privilege class from COYOTE by entering:

```
revoke authority coyote classes=storage
```

2. Grant restricted storage privilege over the BACKUPPOOL and ARCHIVEPOOL storage pools by entering:

```
grant authority coyote stgpools=backuppool,archivepool
```

## Reducing Restricted Policy or Storage Privilege

You can reduce restricted policy privilege by revoking authority over previously authorized policy domains. You can also reduce restricted storage privilege by revoking authority over previously authorized storage pools.

For example to reduce administrator COYOTE's, restricted storage privilege, revoke authority for the storage pool named ARCHIVEPOOL by entering:

```
revoke authority coyote stgpools=archivepool
```

# Managing Administrator Access

A system administrator can control access to the server by renaming an administrator, removing an administrator, or by locking and unlocking an administrator from the server. This section describes these methods and gives examples.

| Task | Required Privilege Class |
|---|---|
| Rename an administrator user ID<br>Remove other administrators from the server<br>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"
```

**Note:** The administrator SERVER_CONSOLE cannot be renamed.

## Removing Administrators

You can remove other administrators from the server so that they no longer have access to administrator functions. For example, to remove registered administrator ID SMITH, enter:

```
remove admin smith
```

**Notes:**

1. ADSM does not allow you to remove the last system administrator from the system.
2. The administrator user ID named SERVER_CONSOLE cannot be removed.

## Locking and Unlocking Administrators from the Server

To temporarily prevent other administrators from accessing the system, a system administrator can lock out other administrators from ADSM.

For example, you can lock out an administrator when an employee takes a leave of absence from your business. You can lock out the administrator user ID MARYSMITH from the server user ID by issuing the LOCK ADMIN command:

```
lock admin marysmith
```

When the employee returns, any system administrator can unlock the administrator ID. For example, you can unlock MARY SMITH’s user ID by issuing the UNLOCK ADMIN command:

```
unlock admin marysmith
```

MARYSMITH can now access ADSM to complete administrative tasks.

**Note:** You cannot lock or unlock the SERVER_CONSOLE administrator user ID from the server.

### Requesting Information about Administrators

Any administrator can query the server to view administrator information. You can also query all administrators authorized with a specific privilege class.

For example, to query the system for a detailed report on administrator ID DAVEHIL, issue the QUERY ADMIN command:

```
query admin davehil format=detailed
```

Figure 23 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 23. A Detailed Administrator Report*

## Managing Client Nodes

| Task | Required Privilege Class |
|---|---|
| Set registration to open or closed | System |
| Register client nodes to any policy domain | System or unrestricted policy |
| Register client nodes to specific policy domains | System, unrestricted policy, or restricted policy for those domains |
| Update, rename, lock, or unlock any client nodes | System, unrestricted policy |
| Update, rename, lock, or unlock client nodes assigned to specific policy domains | System, unrestricted policy, or restricted policy for those domains |
| Request information about client nodes or file spaces | Any administrator |
| Delete any file space from storage pools | System or unrestricted policy |

| Task | Required Privilege Class |
|---|---|
| Delete file spaces defined for client nodes assigned to specific policy domains | System, unrestricted policy, or restricted policy for those domains |
| Remove any client nodes | System or unrestricted policy |
| Remove client nodes assigned to specific policy domains | System, unrestricted policy, or restricted policy for those domains |

Managing client node registration includes:

- Setting client node registration to open or closed
- Registering client nodes
- Updating client node information
- Managing client node access
- Requesting information about client nodes and file spaces
- Deleting file spaces and client nodes

## Setting Client Node Registration

Before a user can begin requesting backup or archive services from an ADSM server, the workstation must be registered with a server.

ADSM provides two methods for registering workstations with an ADSM server:

**Open registration**
Users register their own workstations as client nodes with the server.

**Closed registration**
An administrator registers each workstation as a client node with the server.

At installation, registration is set to closed.

**Note:** You can use the SET REGISTRATION command to change registration at any time. Existing registered client nodes are not affected by changes to the registration process.

### Open Registration

With open registration, users can register their own workstations with the server. When a user attempts to access the server 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. A user with root authority can define whether compression is used by entering the appropriate value for 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 any of these defaults after the client node has been registered, you can update client node registration information as described in "Updating Client Node Information" on page 119.

## Closed Registration

With closed registration, a system administrator, an administrator with unrestricted policy privilege, or an administrator with restricted policy privilege can define the following information for each client node:

- The workstation node name. UNIX users must provide the value returned by the HOSTNAME command to the administrator.
- The user password.
- The policy domain to which the client node is assigned. The policy domain contains the policy objects that control how ADSM manages user backup and archive data.
- Whether the user can compress files at the workstation before sending them to the server. You can select one of three options to specify whether the client program should:
  - Compress files before sending them to the server. Time is saved when objects are compressed before they are backed up or archived to the server. In addition, server storage space is saved.
  - Send files that are not compressed to the server

    Compression can affect ADSM throughput and require more workstation memory. 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.
  - Use the value defined for the COMPRESSION option to determine whether or not to compress objects being sent to the server

    The COMPRESSION option can be set in the application configuration file or in the client system options 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 are not allowed to delete archived or backed up files, an administrator with system or policy privilege must delete any files associated with the client nodes from storage pools. See "Deleting File Spaces and Client Nodes" on page 123 for information on deleting files from storage pools.

## Registering Client Nodes

Authorized administrators can register client nodes with the server. If registration is open, then users can register their own workstation with the server. See “Setting Client Node Registration” on page 116 for information about setting open or closed registration.

To register a client node, you define:

- The workstation node name
- The user password and contact information
- Whether the user client program should compress files
- Whether the user is allowed to delete backed up files from storage
- Whether the user is allowed to delete archived files from storage

For example, you want to register three workstations from your engineering department to the server. Because your engineers have unique storage management needs, you want to assign them to the *ENGPOLDOM* policy domain. Before you can assign client nodes to a policy domain, you must define the policy domain. See Chapter 6, “Managing Policies” on page 127 for information on defining new policy domains.

For each workstation, define a node name, password, and contact information. Set file compression on so that, when users back up or archive files, the client node attempts to compress the files before transmitting them to storage pools. Finally, allow these engineers to delete backed up or archived files from storage pools.

If you are administering the server from an administrative client, you can use the macro facility to register more than one client node at a time.

For this example, create a macro file named REGENG.MAC. Enter the REGISTER NODE commands in the macro file by using your preferred workstation editor:

```
register node ssteiner choir contact='department 21' -
domain=engpoldom compression=yes archdelete=yes cackdelete=yes
register node carolh skiing contact='department 21, second shift' -
domain=engpoldom compression=yes archdelete=yes cackdelete=yes
register node mab guitar contact='department 21, third shift' -
domain=engpoldom compression=yes archdelete=yes backdelete=yes
```

After you create the macro file, invoke the macro from an administrative client by issuing the MACRO command:

```
macro regeng.mac
```

ADSM processes the commands in the macro file and registers ssteiner, carolh and mab client nodes with the server.

## Managing Client Node Access

You can control client node access to ADSM by updating or renaming client nodes or by locking and unlocking a client nodes from the server. This section describes these methods and gives examples.

### Updating Client Node Information

You can update the following client node registration 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, if a user registers the workstation during open registration as administrator TOMC, ADSM:

- Assigns TOMC to the policy domain named STANDARD
- Allows TOMC to define whether or not to compress files from storage pools
- Allows TOMC to delete archived, but not backed up, files from storage pools

Novice ADSM users can be restricted from deleting archived files from storage pools. You can update the node registration for the TOMC client node by issuing the UPDATE NODE command:

```
update node tomc archdelete=no
```

TOMC is now restricted from deleting archived files from storage pools. Only an administrator with system or policy privilege can delete TOMC's archived files.

### Renaming Client Nodes

You can rename a client node if the workstation network name or host name is changed. For example, with UNIX systems, users define their ADSM node named based on the value returned by the HOSTNAME option. 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, you can rename CAROLH to ENGNODE by issuing the RENAME NODE command:

```
rename node carolh engnode
```

The client node named ENGNODE retains contact information and access to backup and archive data. In addition, all files backed up or archived by CAROLH now belong to the client node named ENGNODE.

### Locking and Unlocking Client Nodes

You can lock a client node from accessing the server, and you can unlock the client node to allow a user to reaccess the server. For example, you can lock the client node named MAB from the server by issuing the LOCK NODE command:

```
lock node mab
```

When the client node is locked from the server, a user is prevented from accessing the server to perform ADSM functions, such as backing up or archiving files to the server, or restoring or retrieving files from the server.

You can allow client node MAB to reaccess the server by issuing:

```
unlock node mab
```

## Requesting Information about Client Nodes or File Spaces

After client nodes are registered with the ADSM server, users have the option of defining file spaces on their workstation. A *file space* name identifies a group of files that are stored as a logical unit in data storage.

On client systems 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 than a volume with the label ABC.

On client systems 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 the VIRTUALMOUNTPOINT 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*.

### 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 Specified Policy Domains:*** You can display information about client nodes assigned to specific policy domains. For example, to query the server to view information about any client nodes assigned to the policy domains named STANDARD and ENGPOLDOM, issue the QUERY NODE command:

```
query node * domain=standard,engpoldom
```

Figure 24 displays a standard report.

```
Node Name    Platform     Policy Domain   Days Since   Days Since   Locked?
                          Name                  Last     Password
                                              Access          Set
----------   --------     --------------  ----------   ----------   -------
DEBBYG         DOS        STANDARD                 2           12    No
ENGNODE        AIX        ENGPOLDOM               <1            1    No
HTANG          OS/2       STANDARD                 4           11    No
MAB            AIX        ENGPOLDOM               <1            1    No
PEASE          AIX        STANDARD                 3           12    No
SSTEINER       (?)        ENGPOLDOM               <1            1    No
```

*Figure 24. Requesting Client Node Information*

***A Specific Client Node:*** You can display detailed information about specific client nodes. This is useful when you want to review the registration parameters defined for users. For example, you can display a detailed report for the client node named PEASE, by issuing the QUERY NODE command:

```
query node pease format=detailed
```

The following figure displays the contents of a detailed report for the client named PEASE.

```
                    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 File Space Information

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 actual space consumed 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. Be aware that file space names are case-sensitive. You can request a standard or detailed report.

For example, you can query the server for information about file spaces defined for the client node named PEASE, by issuing the QUERY FILESPACE command:

```
query filespace pease *
```

The figure below displays the output from this command. It shows that node ID PEASE has defined three file spaces on his AIX workstation, which has been registered as a client node named PEASE. It also shows that node ID PEASE is running the *JFS* file system on his AIX workstation. Finally, this report shows how much space is available in each file space on the node ID PEASE's workstation and how much space is being used.

```
Node Name                       Filespace   Platform Filespace Capacity %Util
                                Name                 Type          (MB)
------------------------------- ----------- -------- --------- -------- -----
PEASE                           /home/peas- AIX      JFS          196.0  91.7
                                 e/dir
PEASE                           /home/peas- AIX      JFS          328.0  81.0
                                 e/dir1
PEASE                           /home/peas- AIX      JFS           46.9  96.0
                                 e/dir2
```

*Figure 25. Report Received from the Query Filespace Command*

## Deleting File Spaces and Client Nodes

You can delete a client node from a server. First, however, you must delete any client backup, archive, and client migrated data from storage pools by deleting the filespace 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 to the server. See "Setting Client Node Registration" on page 116 and "Registering Client Nodes" on page 118 for information on allowing users to delete files in storage pools.

  For example, node ID PEASE does not have the authority to delete files that he no longer needs in the file space named */home/pease/dir2*. An administrator must delete the files for him by issuing the DELETE FILESPACE command:

```
delete filespace pease /home/pease/dir2 type=archive
```

- You want to remove a client node from the server

  You must delete a user's files from storage pools before you can remove a client node. For example, to delete all backup and archive files stored in any file spaces defined to the client node ID DEBBYG, enter the following command:

```
delete filespace debbyg * type=any
```

- You want to delete files belonging to a specific owner

  For client nodes that support multiple users, such as UNIX, a file owner name is associated with each file on the server. The owner name is the user ID of the operating system, such as the UNIX user ID. When you delete a file space belonging to a specific owner, only files that have the specified owner name in the file space are deleted.

### Removing Client Nodes

Before you can remove a client node from the server, all backed up or archived files belonging to the client node must be deleted from storage pools. See “Deleting a File Space” on page 123 for information on deleting backed up or archived files from storage pools.

For example, you can remove the client node ID DEBBYG from the server, by issuing the REMOVE NODE command:

```
remove node debbyg
```

A a client node that has been removed from the server can no longer back up, archive, or migrate files to ADSM.

## Registering an Application Programming Interface to the Server

Workstation users can begin requesting services from an ADSM server by using an application that uses the ADSM application programming interface (API). An administrator must register the workstation as a client with an ADSM server. After the workstation is registered with a server, it can begin to back up, archive, restore, and retrieve objects by using the application’s interface.

To register an API to the server obtain the following information from API users:

- Node name of client
- Initial password, if a password is required
- Contact information such as the user name, user ID, and telephone number

To register an application by using the ADSM API as a client node, define the following:

- Workstation node name
- User password and client information
- Compression status of the client
- Policy domain to which your workstation belongs

### 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 by the client before they are sent to an ADSM server
  - 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.

## 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 data 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 6. 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 | 128 |
| Policy objects | 129 |
| Management classes | 131 |
| Expiration processing | 134 |
| File eligibility for policy operations | 134 |
| How client migration works with backup and archive | 137 |
| **Tasks:** | |
| Using the standard storage management policies | 137 |
| Creating your own storage management policies | 138 |
| Defining a policy domain | 142 |
| Defining a policy set | 143 |
| Defining a management class | 144 |
| Defining a backup copy group | 145 |
| Defining an archive copy group | 149 |
| Assigning a default management class | 150 |
| Validating and activating policy sets | 150 |
| Starting expiration processing | 152 |
| Querying policy objects | 152 |
| Deleting policy objects | 155 |

Most tasks presented in this chapter can be performed by using either the graphical user interface or the command-line interface. Table 5 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, refer to *ADSM Administrator's Reference*. For help performing a task on the graphical user interface, see the procedure described in “Using Online Help” on page 20.

## Policy Operations

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, or users can request backups of their own data. ADSM provides two types of backup:

**Incremental backup**
The backup of files that have changed since their last backup. ADSM does the following incremental backup processing:

1. Compares a file on the client workstation with its copy in storage.
2. Determines if the file has changed since the last backup.
3. Backs up files that have been changed and that meet policy eligibility requirements.

**Selective backup**
Backs up files that the user specifies and that meet any requirements defined by the applicable 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. After users archive a file, they can save disk space by erasing the original file from their workstation.

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 data 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.

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 premigrated files at the server can be quickly changed to migrated files. 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. This process defines policy objects that are used to implement ADSM policies. The following figure shows the objects and their relationships:

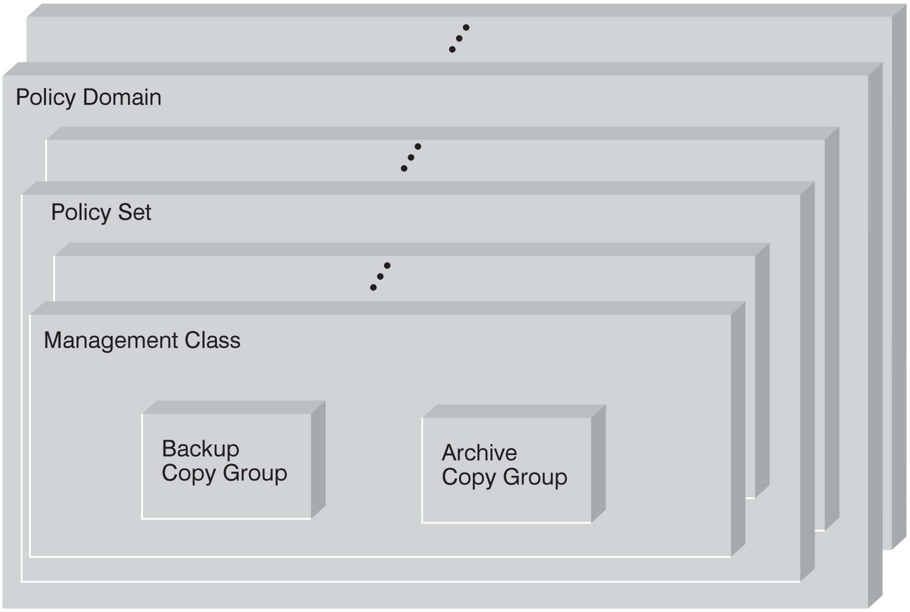


*Figure 26. ADSM Policy Objects*

**Backup copy group**

Controls how ADSM performs backup processing of files associated with it. 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 it. 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.

**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
- Restrict the number of management classes to which users have access

## Management Classes

Each user is assigned to a single policy domain, and the user has access only to the management classes contained in the domain. The management classes specify whether or not space management is to be done. 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 these client nodes to the policy domain. See “Registering Client Nodes” on page 118 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 user can erase the original file without affecting how long the archive copy is retained in data storage. Examples include legal records, patient records, and tax forms.

**Neither a backup nor an archive copy group**

A management class that contains neither a backup nor an archive copy group prevents a file from ever being backed up or archived. If users bind their files to a management class without copy groups, ADSM issues warning messages.

**Attention:** This This type of management class is not recommended for most users. Use such a management class carefully to prevent users from mistakenly selecting it.

## 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 133.
- 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, and how ADSM manages backed up, archived, and migrated 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 migration.

With an include-exclude list, users can:

- Exclude files or directories from backup and client migration operations

  For example, the following figure 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, the following figure shows that the SSTEINER node ID includes the /home/ssteiner/options.scr file. This file is now eligible for backup and client migration.

- Bind a file to a specific management class

  For example, the following figure 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
include /home/ssteiner/options.scr
include /home/ssteiner/driver5/.../* mcengbk2
```

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 files with a management class. The policies defined in the management class then apply to the bound files. A user binds a file to a management class name by using:

- The INCLUDE option in an include-exclude list
- The ARCHMC option when archiving a file
- The DIRMC option when backing up a file

See the user's publication for the appropriate client for details.

When a user does not bind a file to a management class, the client node binds the file to the default management class in the active policy set.

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 data 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” 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* 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.
- The user specifies a different management class by using the DIRMC option when doing a backup.
- An administrator activates a policy set that does not contain a management class with the same name.
- 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.

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.

**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.

### Expiration Processing

Backup and archive copy groups can specify the criteria that make copies of files eligible for deletion from data 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 is not reclaimed from expired client files, and the ADSM server requires increased storage space.

See “Starting Expiration Processing” on page 152 for details about how to invoke expiration processing.

Diagnosis, Modification or Tuning Information

## File Eligibility for Policy Operations

This section describes how ADSM selects files for the following operations:

- Incremental backup
- Selective backup

- Archive
- Migration from a client node (hierarchical storage management)

## Incremental Backup

When a user requests an 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 if the file is backed up only if it has changed since the last backup (*modified*) or whenever a backup is requested (*absolute*).

   **Frequency** Specifies the minimum number of days that must elapse between backups and how files are handled if they are modified during backup processing.

   **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, the server 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, 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, the file is not eligible for backup.

## 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, only files specified on the command line 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 eligible for backup. 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.

## 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 a 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 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.

End of Diagnosis, Modification or Tuning Information

## 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 users restore a backup version of a migrated file, ADSM deletes the migrated copy of the file from server storage the next time reconciliation is run.

If users do not specify that the file is to be erased from the client after it is archived, the file remains migrated. If users specify that the file is to be erased, ADSM deletes the migrated file from ADSM 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 following table shows the attributes of the ADSM-supplied objects.

*Table 14. Standard Policy Objects Provided with ADSM*

| | |
|---|---|
| **Standard Policy Domain**<br>• Backup retention grace period = 30 days<br>• Archive retention grace period = 365 days | |
| **Standard Policy Set (ACTIVE)**<br>• Default Management class = STANDARD | |
| **Standard Management Class**<br>• No space management | |
| **Standard Backup Copy Group**<br>• Destination = BACKUPPOOL<br>• Backup frequency = 0 days<br>• Incremental backup of modified files<br>• Files cannot be backed up if in use<br>• Retain up to 2 versions<br>• Retain version for 60 days | **Standard Archive Copy Group**<br>• Destination = ARCHIVEPOOL<br>• Files cannot be archived if in use<br>• Retain archive copy for 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:<br>• A copy of each policy set from the original domain<br>• A copy of each management class in each original policy set<br>• A copy of each copy group in each original management class |
| Policy Set | A new policy set **in the same policy domain** with:<br>• A copy of each management class in the original policy set<br>• 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 142.
2. Define policy sets for different storage management policies. See page 143.
3. Define management classes to match users' storage management requirements. See page 144.
4. Define backup copy groups to specify which files can be backed up and how to manage backup versions. See page 145.
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 149.
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 150.
7. Validate all policy sets, and activate one policy set for each policy domain. See page 151.
8. Start expiration processing. See page 152.

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 7, “Scheduling Operations” on page 159.

## Example: Sample Policy Objects

The following figure shows the policies for an engineering department. This example is used throughout the rest of this chapter.

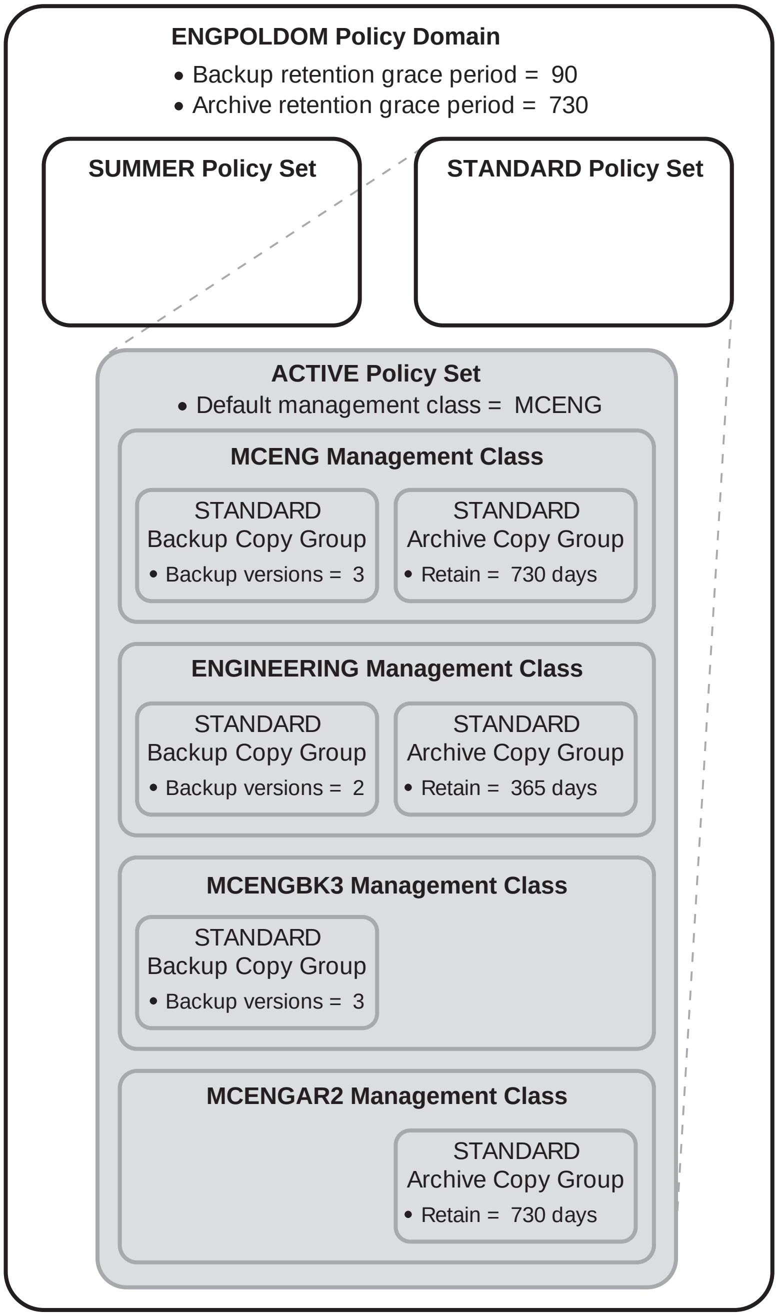

ENGPOLDOM Policy Domain
• Backup retention grace period = 90
• Archive retention grace period = 730
SUMMER Policy Set
STANDARD Policy Set
ACTIVE Policy Set
• Default management class = MCENG
MCENG Management Class
STANDARD Backup Copy Group
• Backup versions = 3
STANDARD Archive Copy Group
• Retain = 730 days
ENGINEERING Management Class
STANDARD Backup Copy Group
• Backup versions = 2
STANDARD Archive Copy Group
• Retain = 365 days
MCENGBK3 Management Class
STANDARD Backup Copy Group
• Backup versions = 3
MCENGAR2 Management Class
STANDARD Archive Copy Group
• Retain = 730 days

*Figure 27. An Example of Policy Objects Defined for an Engineering Department*

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.

## 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 a 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 deleted when:

- A user binds a file to a new management class name that does not contain a backup copy group
- The management class to which a file is bound no longer exists and the default management class does not contain a backup copy group

When users try to back up the file and the management class does not contain a backup copy group, ADSM uses the backup retention grace period to manage all existing backup versions of the file, and the file is not backed up.

Backup versions of the file are retained in data storage only for the backup retention grace period. This period starts from the day of the backup, unless a user binds the file to a management class containing a backup copy group and does a 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.

**Archive Retention Grace Period**

Specifies the number of days to retain an archive copy when the server cannot rebind the file to an appropriate management class. The retention grace period protects archive copies from being immediately deleted if the default management class does not contain an archive copy group.

The archive copy of the file is retained in data storage for the number of days specified by the archive retention grace period starting 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.

### 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, enter:

```
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 space management 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 the 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 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 an 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.

**How frequently files can be backed up**

Specifies the minimum number of days that must elapse between incremental backups. Frequency works with the mode parameter, which specifies whether a file or directory is considered for 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 selective backup for a file. You can select from two modes:

**Modified**

A file is considered for 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 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 considered for backup each time incremental backups are scheduled for or initiated by a user.

**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 28 on page 147, 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 server deletes the oldest version.

For example, if the maximum number of versions allowed for MEMO.DAT is 3, and a user runs a backup process that creates a fourth version, the server deletes the oldest version. In this example, the backup version created on Thursday at 8:05 a.m. is deleted from data storage.

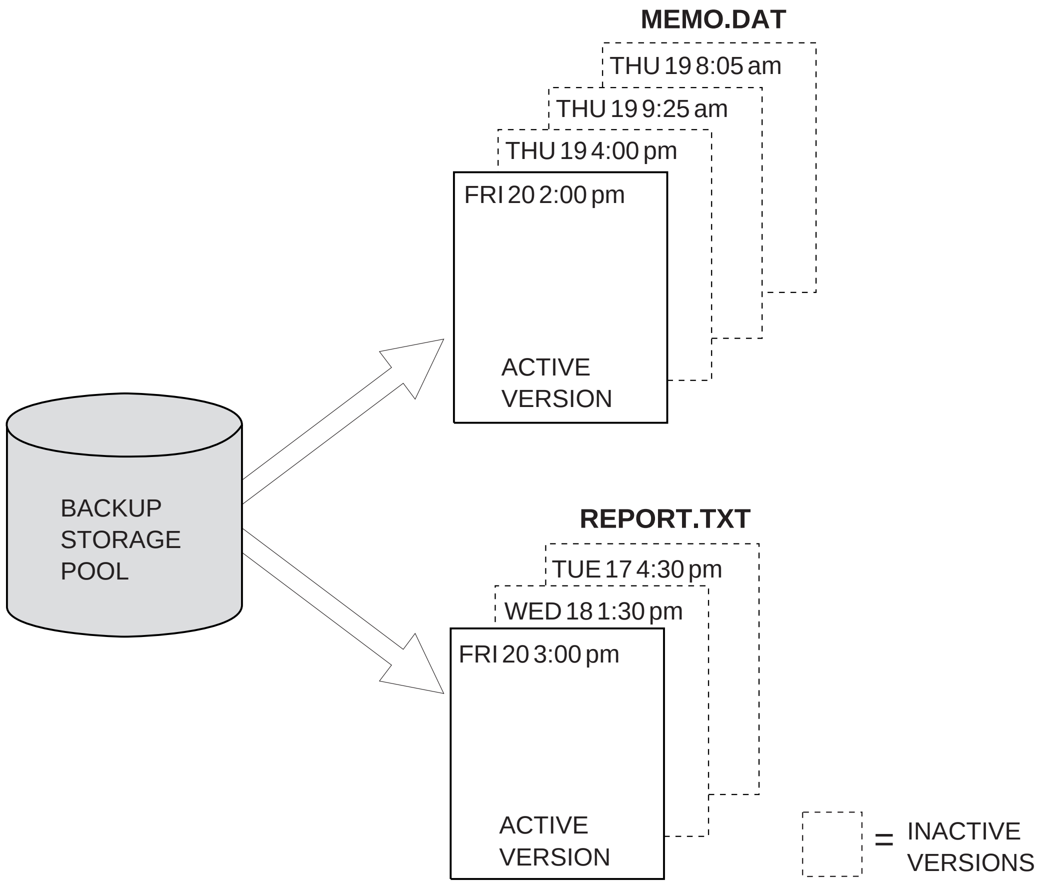


*Figure 28. Example of Active and Inactive Versions of Backed Up Files*

**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 node, then the next time a backup process is run, the server changes the active backup version to inactive and erases the oldest versions that are more than the number specified by this parameter.

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 two parameters that determine 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 space from data 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 3 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
- 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 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 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. There are four options that you can chose from: This attribute, called serialization, can be one of four values:

**Static**

Specifies that if the file or directory is modified during an archiving process, ADSM does not archive it. ADSM does not retry the archive.

**Shared Static**

Specifies that if the file or directory 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 or directory is archived on the first attempt, even if the file or directory is being modified during the archive process.

**Shared Dynamic**

Specifies that if a file or directory is modified during the archive attempt, ADSM archives it on its last try even if the file or directory 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.

**How long to retain an archived copy**

Specifies the number of days to retain an archived copy in storage. When the time elapses, ADSM deletes the file.

### 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 132 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 default management class is copied from the STANDARD policy set to the SUMMER policy set. 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 fails when the operation involves storing a file in a nonexistent storage pool.

- 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 133 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 142 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 backup retention grace period, and the workstation file is not backed up. See “Defining and Updating a Policy Domain” on page 142.

- 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
```

## Starting Expiration Processing

Copies of files that have expired are not deleted from data storage until expiration processing occurs. You can invoke expiration processing either automatically or by command. Automatic expiration processing can be controlled by the EXPINTERVAL parameter specified in the ADSM options file (dsmserv.opt). For details, see *ADSM Installing the Server and Administrative Client*. You can manually start expiration processing by issuing the following command:

```
expire inventory
```

Expiration processing then deletes eligible backup versions and archive file copies.

## 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        Policy        Mgmt          Copy            Retain
Domain        Set Name      Class         Group          Version
Name                        Name          Name
---------     ---------     ---------     ---------     --------
ENGPOLDOM     ACTIVE        MCENG         STANDARD           730
ENGPOLDOM     ACTIVE        STANDARD      STANDARD           365
ENGPOLDOM     STANDARD      MCENG         STANDARD           730
ENGPOLDOM     STANDARD      STANDARD      STANDARD           365
ENGPOLDOM     SUMMER        MCENG         STANDARD           730
ENGPOLDOM     SUMMER        STANDARD      STANDARD           365
```

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       Policy       Mgmt         Default      Description
Domain       Set Name     Class        Mgmt
Name                      Name         Class ?
---------    ---------    ---------    ---------    ------------------------
ENGPOLDOM    ACTIVE       MCENG        Yes          Engineering Management
                                                     Class with Backup and
                                                     Archive Copy Groups
ENGPOLDOM    ACTIVE       STANDARD     No

ENGPOLDOM    STANDARD     MCENG        Yes          Engineering Management
                                                     Class with Backup and
                                                     Archive Copy Groups
                                                     versions
ENGPOLDOM    STANDARD     STANDARD     No

ENGPOLDOM    SUMMER       MCENG        Yes          Engineering Management
                                                     Class with Backup and
                                                     Archive Copy Groups
                                                     versions
ENGPOLDOM    SUMMER       STANDARD     No
```

## Querying Policy Sets

To query the system for information about policy sets in the ENGPOLDOM engineering policy domain, enter:

```
query policyset engpoldom *
```

The following figure is the output from the query. It shows an ACTIVE policy set and two inactive policy sets, STANDARD and SUMMER.

```
Policy        Policy        Default       Description
Domain        Set Name      Mgmt
Name                        Class
                            Name
---------     ---------     ---------     ------------------------
ENGPOLDOM     ACTIVE        MCENG         Policy Set Activated
                                           During Summer
ENGPOLDOM     STANDARD

ENGPOLDOM     SUMMER        MCENG         Policy Set Activated
                                           During Summer
```

## Querying Policy Domains

To request information about a policy domain (for example, to determine if any client nodes are registered to that policy domain), enter:

```
query domain *
```

The following figure is the output from the query.  It shows that both the ENGPOLDOM and STANDARD policy domains have client nodes assigned to them.

```
Policy       Activated    Activated     Number of    Description
Domain       Policy       Default      Registered
Name         Set          Mgmt              Nodes
                          Class
---------    ---------    ---------    ----------    ------------------------
ENGPOLDOM    SUMMER       ENGMC                 3    Engineering Policy
                                                      Domain
STANDARD     STANDARD     STANDARD              3    Installed default policy
                                                      domain.
```

## Deleting Policy Objects

| 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 |

In the following sections, note that you cannot delete the ACTIVE policy set or objects in that policy set.  Also note that when you delete an object, you also delete any objects belonging to it.

## 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.

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. To determine if it belongs to the ACTIVE policy set, issue the QUERY MGMTCLASS command.

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. To determine if it is active, issue the QUERY POLICYSET command. 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 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 command.

To delete the STANDARD policy domain, perform the following steps:

1. Request a list of all client nodes assigned to the 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, create a macro by entering:

```
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 133.

   - 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.

# Chapter 7.  Scheduling Operations

ADSM includes a central scheduling component that allows the automatic processing of administrative commands and client operations.  Administrative and client schedules consist of commands that are automatically processed during a specific time period.

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.  Automating client operations frees users from having to perform them manually.

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

| Section | Page |
|---|---|
| **Concepts:** | |
| Central scheduling | 159 |
| Scheduling modes | 160 |
| **Tasks:** | |
| Setting scheduling modes | 162 |
| Coordinating server and client node schedules | 163 |
| Managing schedules and associations | 167 |
| Managing scheduled events | 174 |

Most tasks presented in this chapter can be performed using either the graphical user interface or the command-line interface.  Table 6 on page 44 shows whether a task can be performed on the graphical user interface, the command-line interface, or both.

For information about issuing ADSM commands, refer to *ADSM Administrator's Reference*.  For help performing a task on the graphical user interface, see the procedure described in "Using Online Help" on page 20.

## Central Scheduling

ADSM lets you define, update, copy, query, and delete administrative command and client operation schedules.  These schedules consist of administrative commands or client operations that are processed during a specified time period when the schedule is activated.

Each administrative command and each 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 space requires.

## Scheduling Modes

The central scheduler uses the *client-polling* and *server-prompted* scheduling modes.

## Client-Polling Scheduling Mode

You can use the client-polling scheduling mode with all communication methods. With this mode, a client node queries the server at prescribed time intervals to obtain a schedule. When the scheduled start time begins, the client node performs the scheduled operation and sends the results to the server. The client node then queries the server for its next scheduled operation.

This sequence is illustrated in Figure 29.

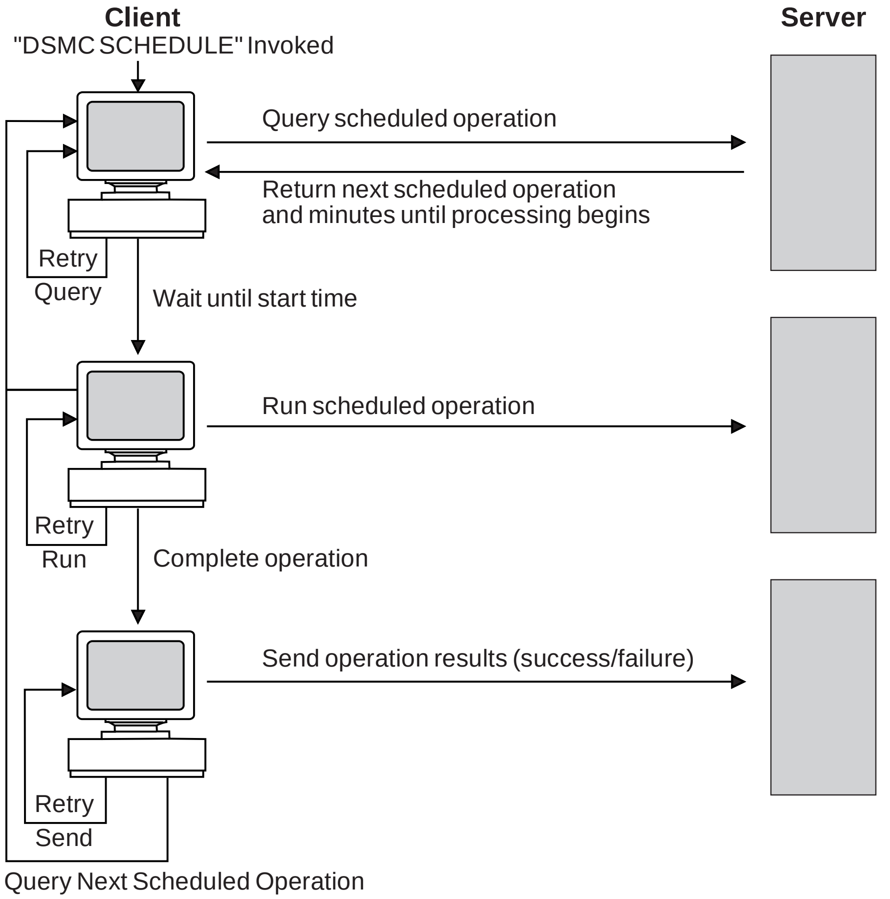

Client
Server
"DSMC SCHEDULE" Invoked
Query scheduled operation
Return next scheduled operation
and minutes until processing begins
Retry
Query
Wait until start time
Run scheduled operation
Retry
Run
Complete operation
Send operation results (success/failure)
Retry
Send
Query Next Scheduled Operation

*Figure 29. Client-Polling Scheduling Mode*

## Server-Prompted Scheduling Mode

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, client nodes register their addresses with the server and then are contacted by the server when scheduled operations need to be performed and a session is available. When contacted, the client node queries the server for the operation, performs the operation, and then sends the results to the server.

This sequence is illustrated in Figure 30.

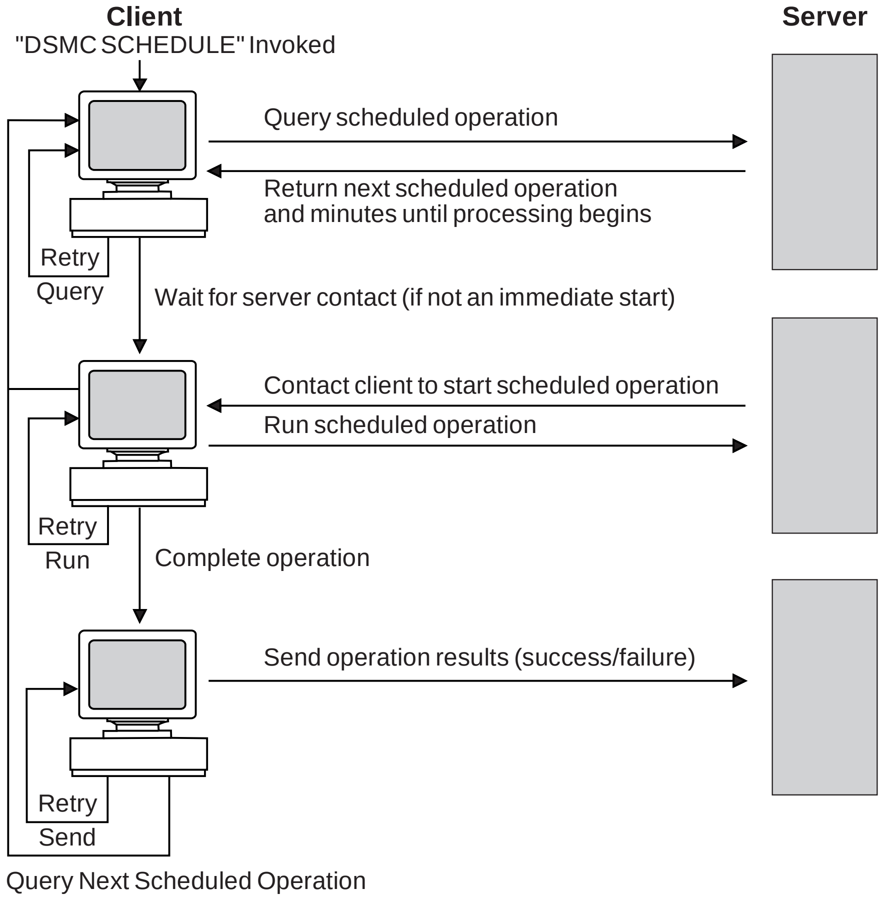

Client
Server
"DSMC SCHEDULE" Invoked
Query scheduled operation
Return next scheduled operation
and minutes until processing begins
Retry
Query
Wait for server contact (if not an immediate start)
Contact client to start scheduled operation
Run scheduled operation
Retry
Run
Complete operation
Send operation results (success/failure)
Retry
Send
Query Next Scheduled Operation

*Figure 30. Server-Prompted Scheduling Mode*

## Setting Scheduling Modes

| Task | Required Privilege Class |
|---|---|
| Set scheduling modes | System |

You can configure the server to use the client-polling scheduling mode, the server-prompted scheduling mode, or both modes.

Users can configure the client nodes to use either the client-polling scheduling mode or the server-prompted scheduling mode.

The following sections describe how to set the scheduling mode on the server and on client nodes.

### Setting the Scheduling Mode on the Server

You can set the scheduling mode on the server to client-polling, to server-prompted, or to both client-polling and server-prompted modes. The scheduling mode is set to both modes at installation.

If the scheduling mode is set to client-polling, the server cannot prompt client nodes:

```
set schedmodes polling
```

In this case, a client node with its scheduling mode set to server-prompted must wait until the server scheduling mode is set to server-prompted or to both server-prompted and client-polling for the scheduled work to begin.

If you set the scheduling mode to server-prompted, client nodes cannot poll the server:

```
set schedmodes prompted
```

In this case, a client node with its scheduling mode set to client-polling must wait until the server scheduling mode is set to client-polling or to both server-prompted and client-polling for the scheduled work to begin.

To let the server support both client-polling and server-prompted scheduling modes, set the scheduling mode to both modes:

```
set schedmodes any
```

In this case, any client node may set any 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*.

## Coordinating Server and Client Node Schedules

| Task | Required Privilege Class |
|---|---|
| • Set the maximum percentage of sessions for scheduled operations<br>• Randomize schedule start times<br>• Set how often clients query the server<br>• Set the maximum number of times the client node scheduler can retry a command that failed<br>• Set the time between retry attempts | System |

By coordinating server and client node schedules, can control the scheduler workload and client node contact with the server.

## 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 145. When you define a schedule, you specify the length of time between processing of the schedule. Do not process the schedule for incremental backups more often than the backup copy group frequency.

## Controlling the Scheduler Workload

An administrator can:

- Set the maximum percentage of concurrent client/server sessions for scheduled operations
- Randomize schedule start times

### Setting the Maximum Percentage of Sessions for Scheduled Operations

The number of concurrent client/server sessions is defined by the MAXSESSIONS option in the server options file, but you can set a maximum percentage of concurrent client/server sessions allowed for processing scheduled operations. Limiting the number of sessions available for scheduled operations ensures that sessions are available when users initiate any unscheduled operations, such as restoring or retrieving files, or backing up or archiving files.

If the number of sessions for scheduled operations is insufficient, you can increase either the total number of sessions or the maximum percentage of scheduled sessions.

However, increasing the total number of sessions can adversely affect server performance, and increasing the maximum percentage of scheduled sessions can reduce the server opportunity to process unscheduled operations.

For example, assume that the maximum number of sessions between client nodes and the server is 80. If you want 25 percent of these sessions to be used by central scheduling, enter:

```
set maxschedsessions 25
```

The server allows 20 sessions to be used for scheduled operations.

For information about the MAXSESSIONS option, refer to *ADSM Installing the Server and Administrative Client*.

## Randomizing Schedule Start Times

To randomize a schedule start time 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.

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.

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.

You might have 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.

For the client-polling scheduling mode, you can specify the percentage of the startup window that the server can use to randomize start times for different client nodes associated with a schedule.

If you set randomization to 0, no randomization occurs. This process can result in communication errors if many client nodes try to contact the server at the same instant.

The maximum percentage of randomization allowed is 50 percent. This limit ensures that half of the startup window is available for retrying scheduled commands that have failed.

It is possible, especially after a client node or the server has been restarted, that a client node may not poll the server until *after* the beginning of the startup window in which the next scheduled event is to start. In this case, the starting time is randomized over the specified percentage of the *remaining* duration of the startup window.

Consider the following situation:

- The startup window for a particular event is from 8:00 to 9:00
- Ten client nodes are associated with the schedule
- Nine client nodes poll the server before 8:00
- One client node does not poll the server until 8:30

To set randomization to 50 percent enter:

```
set randomize 50
```

The result is that the nine client nodes that polled the server *before* the beginning of the startup window are assigned randomly selected starting times between 8:00 and 8:30. The client node that polled at 8:30 receives a randomly selected starting time that is between 8:30 and 8:45.

## Controlling Contact with the Server

To control how often client nodes contact the server to perform a scheduled operation, an administrator can set:

- How often clients query the server
- The number of command retry attempts
- The amount of time between retry attempts

Users (root users on UNIX systems) can also set these values in their client user options files (client system options files for UNIX systems). However, user values are overridden by the values that the administrator specifies.

The client node communication paths to the server can vary widely with regard to response time or the number of gateways. In such cases, you can choose *not* to set these values so that users can tailor them for their own needs.

### Setting How Often Clients Query the Server

For the client-polling scheduling mode, you can specify the maximum number of hours the scheduler on a client node waits between attempts to contact the server to obtain a schedule.

You can set this period to correspond to the frequency with which the schedule changes are being made. If client nodes poll more frequently for schedules, changes to scheduling information (through administrator commands) are propagated more quickly to client nodes. However, increased polling by client nodes also increases network traffic.

If you want to have all clients using polling mode contact the server every 24 hours, enter:

```
set queryschedperiod 24
```

## Setting the Number of Command Retry Attempts

You can specify the maximum number of times the scheduler on a client node can retry a scheduled command that fails.

The maximum number of command retry attempts does not limit the number of times that the client node can contact the server to obtain a schedule. The client node never gives up when trying to query the server for the next schedule.

Be sure not to specify so many retry attempts that the total retry time is longer than the average startup window.

If you want to have all client schedulers retry a failed attempt to process a scheduled command only twice, enter:

```
set maxcmdretries 2
```

## Setting the Amount of Time between Retry Attempts

You can specify the number of minutes the scheduler on a client node waits between retry attempts after a failed attempt to contact the server or after a scheduled command fails to process. You can use this number in conjunction with the number of command retry attempts to control when a client node contacts the server to process a failed command.

Try setting this period to half of the estimated time it takes to process an average schedule.

If you want to have the client scheduler retry failed attempts to contact the server or to process scheduled commands every 15 minutes, enter:

```
set retryperiod 15
```

## Managing Schedules and Associations

| Task | Required Privilege Class |
|---|---|
| Define, update, copy, or delete administrative schedules | System |
| 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 |
| Associate client nodes with any client schedules | System, unrestricted policy, or restricted policy |
| Associate client nodes with client schedules for specific policy domains | System, unrestricted policy, or restricted policy for those domains |
| Display information about scheduled operations | Any administrator |

You can schedule both administrative commands and client operations.

Any administrator can request information about scheduled operations.

To set up an administrative command schedule on the server:

- Define the schedule
- Specify that it is an administrative type command schedule
- Specify if the schedule is to be activated or not

To set up a client schedule on the server:

- Define a schedule

  **Note:** By default, schedules are client schedules. You do not have to specify the type of schedule you are defining unless it is an administrative command schedule.

- Associate client nodes with the schedule

## Defining or Updating Schedules

You can define or update schedules for both administrative commands and client operations. There are parameters on the DEFINE and UPDATE commands that apply to both administrative command and client schedules while others apply only to one type of schedule or the other. The following sections describe these parameters.

### Specifying 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.

You can specify a start time, such as 6:00 p.m.

You can also specify the day of the week on which the startup window begins. 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 might 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. 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.

**How often to run the scheduled service**

You can set the schedule frequency based on a period of hours, days, weeks, months, or years. To have weekly backups, for example, set the period to 1 week.

**Expiration date**

You can specify an expiration date for a schedule 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. 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**

If you are defining an administrative command schedule, you **must** specify the command to be processed on the CMD= parameter of the DEFINE command. If you are updating an administrative command schedule, you **must** specify TYPE=ADMINISTRATIVE on the UPDATE command.

**Command**

When you define an administrative command schedule, you **must** specify the complete command that is processed with the schedule. 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.

For example, to schedule the backup of the ARCHIVEPOOL primary storage pool, enter:

```
define schedule backup_archivepool type=administrative
cmd=''backup stgpool archivepool recoverypool''
active=yes startime=20:00 period=2
```

This command specifies that, starting today, the ARCHIVEPOOL primary storage pool is to be backed up to the RECOVERYPOOL copy storage pool every two days at 8:00 p.m.

To update the BACKUP_ARCHIVEPOOL schedule, you could enter:

```
update schedule backup_archivepool type=administrative
cmd=''backup stgpool archivepool recoverypool''
active=yes startime=22:00 period=3
```

Starting with today, the BACKUP_ARCHIVEPOOL schedule begins the backup every three days at 10:00 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 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

**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 use 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.

You can define a new schedule or update an existing 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 1 week to match the backup copy group frequency of the management class in the STANDARD policy set of the ENGPOLDOM 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 5 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 172 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
```

## Querying Schedules

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 04/21/1995 12:45:14      2 H    2 Mo Sat
ENGPOLDOM      WEEKLY_BACKUP    Inc Bk 04/21/1995 12:46:21      4 H    1 W  Sat
```

*Figure 31. Example of a Schedule*

## Deleting Schedules

When you delete a schedule, all associations with client nodes are also deleted. See "Associating Client Nodes with Schedules."

To delete all schedules in the ENGPOLDOM policy domain, enter:

```
delete schedule engpoldom *
```

## Associating Client Nodes with Schedules

Client nodes process operations according to the schedule associated with the node. Client nodes can be associated with more than one schedule. However, the nodes must be assigned to the policy domain to which a schedule belongs.

After a client schedule has been 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 ENGNOD client node with the ENGWEEKLY schedule, both of which belong to the ENGPOLDOM policy domain, enter:

```
define association engpoldom engweekly engnod
```

## 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
```

This command deletes all schedules in the ENGPOLDOM policy domain.

You might want to delete all associations to a client schedule (rather than delete the schedule) if you want to keep the schedule for future use.

## 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 are tracked by the server.

### Querying Event Records

To help manage event records in the database, you can request information about scheduled or completed events. You can specify a time range to limit the amount of information displayed by querying an event. To minimize the processing time when querying events:

- Minimize the time range for queried events
- For client schedules, restrict policy domains, schedules, and client node names to those for which information is required

You can request general or exception reporting queries. A general query provides information about projected and actual scheduled processes. Exception reporting provides information about scheduled processes that did not complete successfully.

If you specify a future time range, the command shows which events should occur based on current schedules.

You can query events each day to see which events were missed in the previous 24 hours. You can also query events based on the number of days that event records are retained in the database. For example, if the retention period for event records is 3 days, you can query events every 3 days.

The following figure shows the report that is displayed after you enter:

```
query event standard weekly_backup
```

```
Scheduled Start      Actual Start         Schedule Name Node Name     Status
-------------------- -------------------- ------------- ------------- ---------
04/21/1995 18:40:00  04/21/1995 19:38:09  WEEKLY_BACKUP GOODELL       Started
```

*Figure 32. Example of a Report from Query Event*

## 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.

### Setting the Event Record Retention Period

You can specify the retention period for event records in the database. After the retention period has passed, 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 PM on June 30, 1995, enter:

```
delete event 06/30/1995 23:59
```

# Chapter 8. Key Storage Management Concepts

This chapter presents key storage management concepts and information about planning the ADSM storage environment. It describes how ADSM manages devices and media based upon information provided in administrator-defined ADSM storage objects.

| Section | Page |
|---|---|
| **Concepts:** | |
| How ADSM Stores Client Data | 178 |
| How ADSM Represents Devices | 180 |
| How ADSM Represents Storage Media | 182 |
| Overview of ADSM Storage Objects | 182 |
| Putting It All Together | 187 |
| **Tasks:** | |
| Planning to Configure the ADSM Storage Environment | 189 |
| Configuring Devices | 191 |
| Scenario: Configuring an 8mm Automated Tape Library | 194 |

## How ADSM Stores Client Data

When users back up, archive, or migrate files, ADSM:

1. **Determines where to store the file**
   ADSM checks the management class bound to the file. The management class indicates where the file should be stored. For backed up and archived files, storage destinations are assigned in the backup and archive copy groups. For migrated files, storage destinations are indicated in the management class.

   See Chapter 6, "Managing Policies" on page 127 for information on assigning storage destinations, assigning copy groups to management classes, and binding management classes to user files.

2. **Stores information about the file in the ADSM database**
   ADSM saves information about each file that it backs up, archives, or migrates in the ADSM database. 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 4, "Managing the Database and Recovery Log" on page 75 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, optical disk, or tape volumes. These media are associated with ADSM storage pools.

   See "Storage Pool Volumes" on page 285 for information on defining volumes to storage pools.

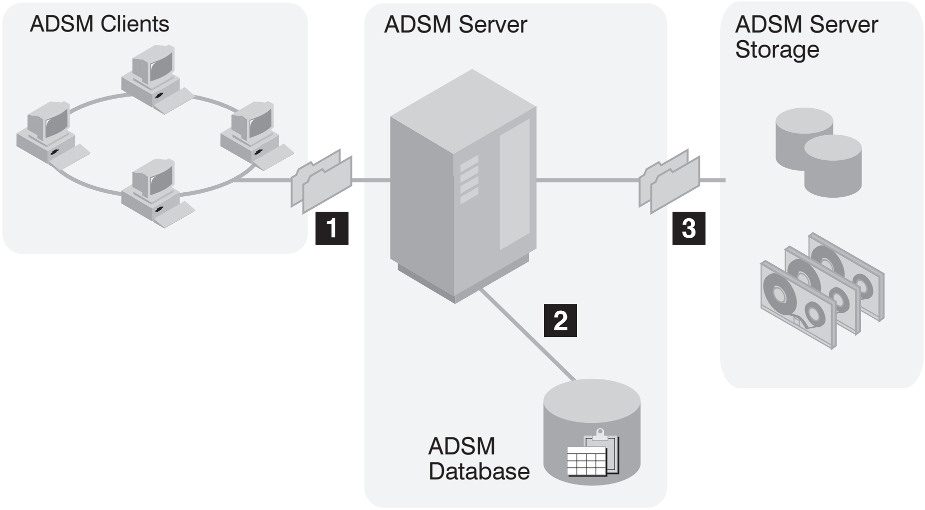


Figure 33 shows the interaction between ADSM policy objects and ADSM backup, archive, and migration services.

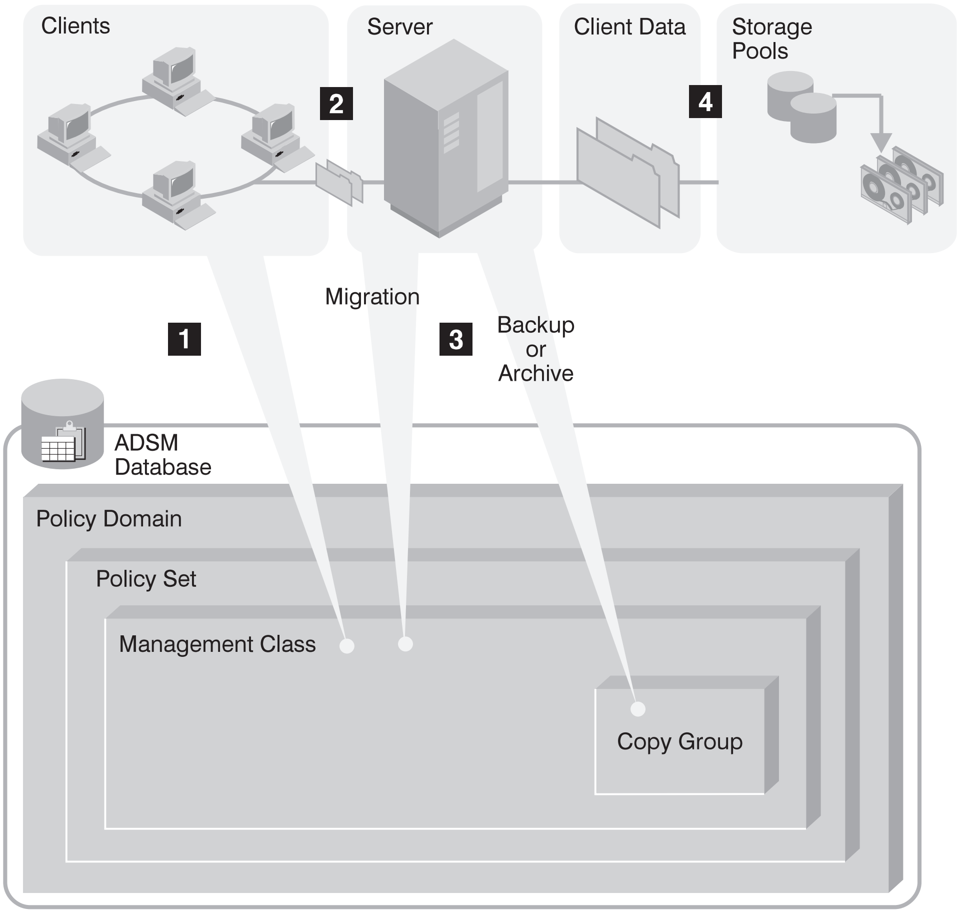


*Figure 33. How ADSM Controls Backup, Archive, and Migration*

1 An ADSM client initiates a backup, archive, or migration operation.

2 The server checks the ADSM database and returns information to the client from which the client determines whether the file is a candidate for backup, archive, or migration. If it is, the client sends the file and file information to the server.

3 The server checks the management class that is bound to the file. The management class indicates where to store the file within ADSM server storage.

4 Where the file is stored depends on the storage destination for that file. The storage destination for migrated files is contained in the management class. The storage destination for backed up and archived files is contained in the copy groups.

## 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 (tape and optical disk) usually require a library and at least one drive specification.

Sequential devices that stand alone require a different library than devices that are associated with an automated, robotic library. 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.

### Disk Devices

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

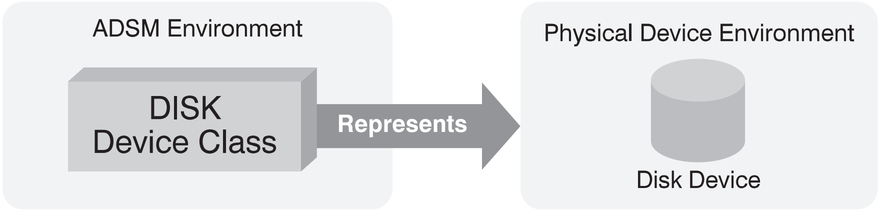


*Figure 34. Magnetic Disk Devices are Represented by Only a Device Class.*

### Tape and Optical Devices

Some devices are represented by more storage objects than others. Figure 35 on page 181 shows that a tape device is represented by a library and a drive in addition to a device class.

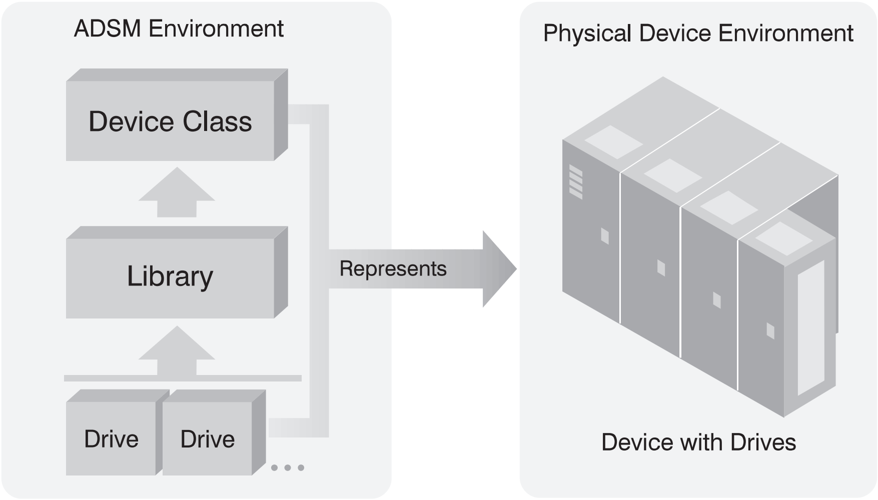


*Figure 35. Tape and Optical Devices are usually represented by a Library, Drive, and Device Class*

## Files as Logical Devices

ADSM allows administrators to create logical volumes on server disk space with the characteristics of sequential tape or optical 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 (logical) 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 36 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.

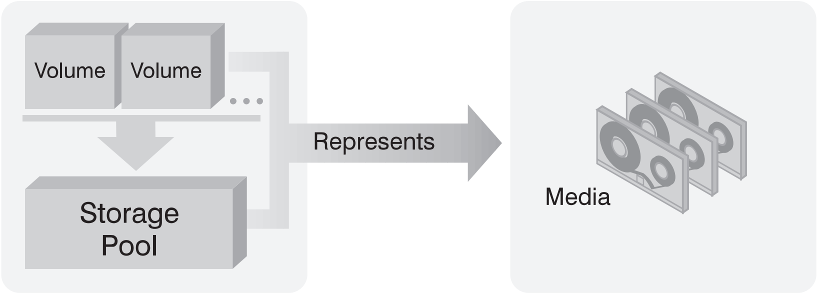


*Figure 36. Relationships of Storage Pool Volumes, Storage Pools, and Media*

## Overview of ADSM Storage Objects

Figure 37 on page 183 shows which ADSM storage objects are required to define random access devices and which are required to define sequential access devices. It also shows the relationship between the ADSM storage objects: storage volumes, storage pools, device classes, libraries, and drives.

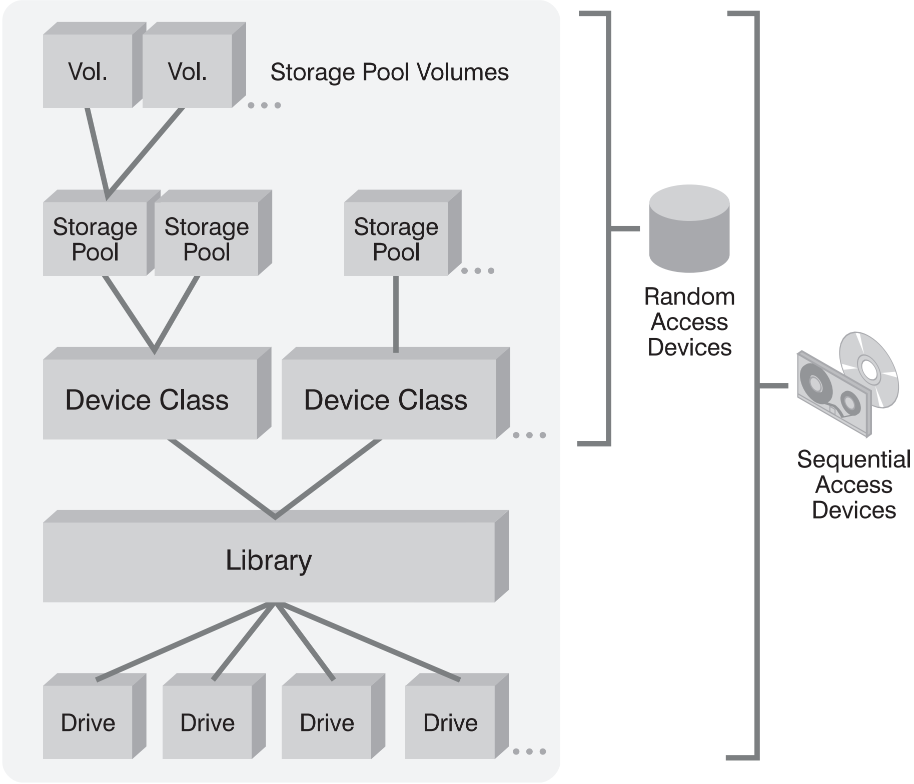


*Figure 37. Relationships of Storage Objects and Device Objects*

## Storage Pool Volumes

A storage pool volume is an ADSM storage object that represents a unit of storage space for ADSM client data. Each storage pool volume is associated with a storage pool. For example, 8mm tapes, QIC tapes, and optical disks become storage pool volumes when they are assigned to their respective ADSM storage pools. See Chapter 12, "Managing Storage Pool Volumes" on page 285 for more information about ADSM storage pool volumes.

## Storage Pools

A storage pool is an ADSM storage object that is a named collection of storage pool volumes. The storage pool represents the physical media available for a device and each storage pool is associated with only one device class. For example, a storage pool for an 8mm tape device contains the 8mm tapes assigned to that device. 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 11, "Managing Storage Pools" on page 237.

### Device Class

A device class is an ADSM storage object that represents a device. A device class contains administrator-defined information about the device type and the way the device manages its media.

For disk type random access devices, the device class alone represents the physical device and ADSM provides a predefined device class of DISK.

For sequentially accessed optical disk and tape devices, the administrator must define the device class so additional storage objects are required for sequential devices because of the many variations in media type (for example, 4mm, 8mm, cartridge, optical disk) and because of the need to manage multiple drives and robotic automation. See Chapter 10, “Managing Storage Devices” on page 219 for more detailed information about device classes.

### Library

A library is an ADSM storage object that represents an administrator-defined collection of drives, and possibly robotic devices (depending on library type) sharing similar media mounting requirements. For example, an automated tape library device is represented by a library object. Each tape device and optical disk device is associated with an ADSM library. The library retains information about drives and media mounting techniques. ADSM accesses the library for this information before mounting a volume. See Chapter 9, “Managing Drives and Libraries” on page 201 for more information about ADSM libraries.

### Drive

A drive is an ADSM storage object that represents the mechanism that can read or write to an optical disk or tape. For devices with multiple drives, each drive is separately defined to the operating system. See Chapter 9, “Managing Drives and Libraries” on page 201 for more information about drives.

## What’s in a Device Class?

The contents of a device class are largely determined by whether the device accesses the data on its media randomly or sequentially. Devices that access their media randomly are relatively straightforward because they all share a common device type (DISK), and they do not require the administrator to define a library object.

### Device Classes for Random Access Devices

Magnetic disk devices are the only devices that access their data randomly. Device classes for random access devices contain only the device type of DISK.

#### Random Access Device Types

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.

Disk devices are the only devices in the random access device category. ADSM provides a predefined, random-access device class, named DISK. You cannot define other random access device classes.

## Device Classes for Sequential Access Devices

Tape and optical disk devices 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 and drive specification. Figure 38 shows the contents of a device class for a typical sequential access device.

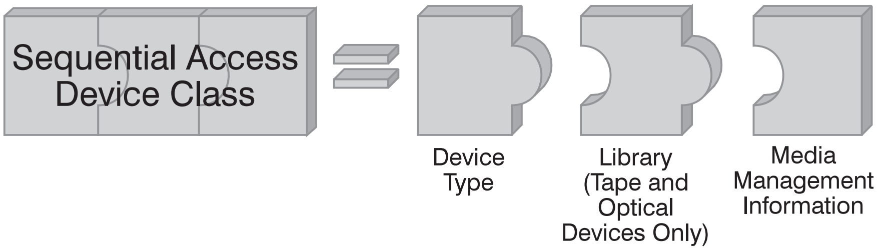

Sequential Access Device Class
Device Type
Library (Tape and Optical Devices Only)
Media Management Information

*Figure 38. Contents of Device Class for Sequential Access Devices*

### Sequential Access Device Types

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 it later by using the reclamation process (see “Space Reclamation for Sequential Access Storage Pools” on page 257 for details).

Tape devices, optical disk 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 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 DISK, 4MM, 8MM, QIC, 3590, CARTRIDGE, OPTICAL, FILE, WORM, and DLT device types. For example, 8mm tape devices require 8mm tapes; all 8mm tape devices share a device type of 8MM.

The device type selected by the administrator determines if the device class is for a random access or sequential access device. This, in turn, determines whether the device requires a library object and a drive object in addition to the device class.

### Library

Tape and optical disk device types (excluding FILE) require a library object to be specified in the device class definition. The library specification in the device class is a pointer to the library object, discussed in more detail in “Library” on page 184.

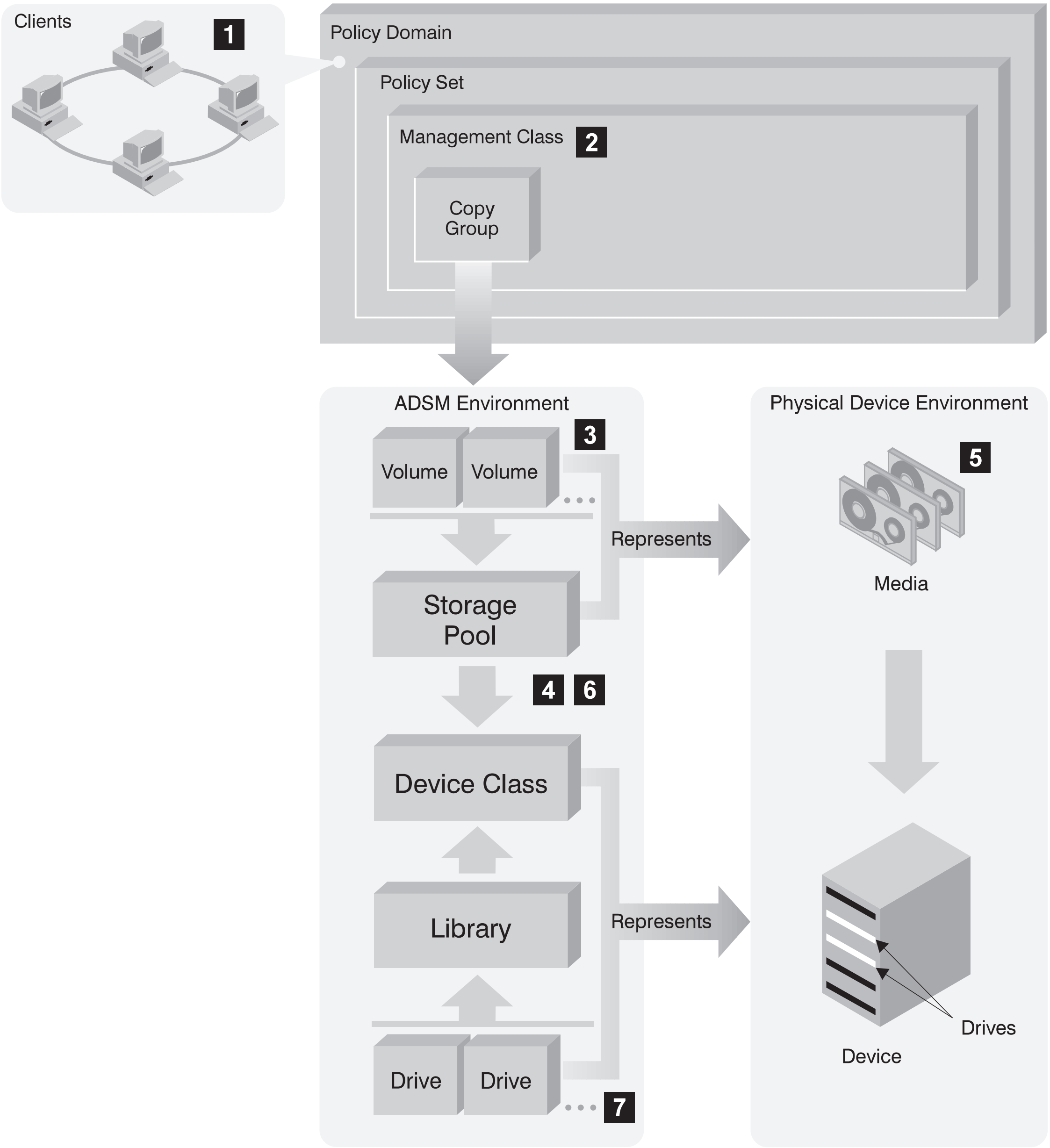
### Media Management Information

Every device class contains media management information. This information is required to manage media (for example, media formatting, estimated capacity, labeling prefixes) in the device class. See Chapter 9, “Managing Drives and Libraries” on page 201 for more information about how ADSM helps to manage media.

|

## Putting It All Together

| This section summarizes the relationships between the physical device environment,
| ADSM storage objects, and ADSM clients.



| *Figure 39. Putting it All Together*

The numbers in this section correspond to the numbers in Figure 39 on page 187.

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

**2** When a client file is backed up, archived, or migrated, it is bound to a management class. Within the management class are copy groups; one for backup and one for archive.

**3** The volumes associated with ADSM storage pools are the destinations for backed up, archived, or migrated files. The copy groups indicate the storage pools to which ADSM backs up or archives files. The management class indicates the storage pool to which ADSM migrates files.

**4** Storage pools are mapped to device classes. Device classes represent devices. Associating a storage pool with a device class effectively ties together client data, media, and a device.

**5** The storage pool contains volumes as indicated in 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).

**6** All devices require a device class containing at least a device type.

**7** Sequential access devices like tape and optical disk will usually require a library and drive to help with the management of media and automation.

## 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.
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 4mm, 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 or enhanced capacity cartridge tapes.
3. Indicate which devices are in an automated library and which devices stand alone. For example, devices in an Exabyte Model 10i, IBM 3494, or IBM 3495 tape library are in an automated tape library.
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:

- 3380 DASDs
- 3390 DASDs
- 3490 Magnetic Tape Subsystem models, including two 3490 Enhanced Capability models (3490E)
- 3480 Magnetic Tape Subsystems, including one with the improved data recording capacity (IDRC) feature

You can map storage devices to device classes as shown in Table 15:

*Table 15 (Page 1 of 2). Mapping Storage Devices to Device Class*

| Device Class | Description |
|---|---|
| Disk | Contains storage volumes that reside on 3390 DASD |

*Table 15 (Page 2 of 2). Mapping Storage Devices to Device Class*

| Device Class | Description |
|---|---|
| Cartridge | Contains standard Cartridge System Tape volumes used with 3480 or 3490 Base tape devices and Enhanced Capacity Cartridge System Tape volumes used with 3494 Magnetic Tape Libraries with 3490E tape devices. |

See Chapter 10, "Managing Storage Devices" on page 219 for information on defining tape device classes to support your physical storage environment.

## Mapping Storage Pools to Device Classes

After you have categorized your storage devices, identify availability, space, and performance requirements for user data stored on disk, tape, or optical disk storage. 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 migrated 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

The administrator defines storage pools that match user requirements to storage devices.

| Storage Pool Name | Device Class | Volumes | Storage Destination |
|---|---|---|---|
| BACKUPPOOL | DISK | Storage volumes that reside on 3390 DASD | For a backup copy group for files requiring immediate access |
| SPACEMGPOOL | DISK | Storage volumes that reside on 3390 DASD | For a management class for space-managed files and requiring immediate access |
| ARCHIVEPOOL | CARTRIDGE | 3494 Tape Library Dataserver | For an archive copy group for files requiring quick, reliable access |
| BACKTAPE | CARTRIDGE | Enhanced Capacity Cartridge System Tape volumes used with 3490E tape devices | For backed up data not requiring immediate access |
| ARCHTAPE | CARTRIDGE | Cartridge System Tape volumes used with 3480 or 3490 tape devices | For archived data not requiring immediate access |

## Configuring Devices

Before a device can be used by ADSM, the device must be configured to the operating system as well as to ADSM.

### Disk Devices

For disk devices, much of the device configuration work has already been taken care of because ADSM provides a predefined DISK device class that is used with all disk devices.

The Administrator performs the following tasks to configure a disk device:

1. Format a random access volume. See Chapter 12, “Managing Storage Pool Volumes” on page 285 for details on using the ADSM random access volume formatting utility.
2. Create a storage pool that is associated with the DISK device class.
3. Define the DISK volumes formatted in step 1 to the new storage pool.
4. Define ADSM policy that links client data with the volumes in the new storage pool. For an example of how to do this, see “Task 4: Define Policy that Links Client Data with Media for the Automated Library” on page 198.

## Tape and Optical Disk Devices

Tape and optical disk devices require more information than magnetic disk devices to manage their different media. Additionally, sequential access devices manage media differently for standalone, operator-mounted devices compared to devices residing in an automated, robotic library.

Sequential access devices typically require the following steps. The numbers in the steps correlate to the numbers in Figure 40 on page 193.

**1** Define the device to the operating system.
See *ADSM Installing the Server and Administrative Client* for more detailed information.

**2** Define each drive mechanism within the device to the operating system.
See *ADSM Installing the Server and Administrative Client* for more detailed information.

**3** Define the device to ADSM.
The administrator issues the define commands required to create the storage objects that represent the physical device and media. See “Overview of ADSM Storage Objects” on page 182 for more detail.

**4** Define ADSM policy objects that link client data with media for the new device.
The administrator issues the define commands required to create the ADSM policy objects that link clients to the pool of storage volumes and to the device.

**5** Register clients to the policy domain created to link client data with storage volumes and devices.

**6** Label the media for the device (this step is not illustrated).

**7** Add the media volumes to the device volume inventory (this step is not illustrated).

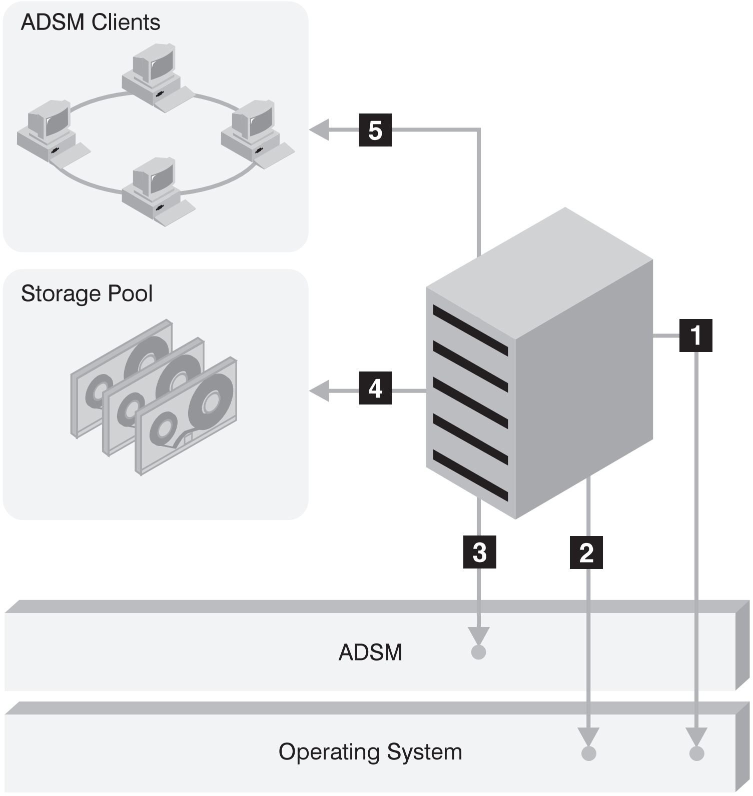


*Figure 40. Overview of Sequential Device Configuration*

## Scenario: Configuring an 8mm Automated Tape Library

This device configuration scenario describes the tasks performed by the administrator to configure an 8mm Exabyte Model 10i automated tape library to ADSM. The scenario includes defining the device to the operating system, creating ADSM policy objects, and creating ADSM storage objects.

The following assumptions are made for this scenario:

**Assumptions**

- The device to configure is an Exabyte Model 10i.
- The ADSM server is running.

  **Note:** If the server is not running, start the server at an operating system prompt by entering:

  ```
  dsmserv
  ```

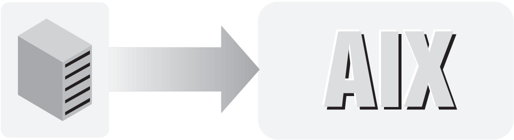
  An operating system prompt is available at an aixterm, a system console, or a tty session.
- The necessary device support module is enabled. When you order ADSM features or device support modules from an IBM representative or authorized reseller, you receive one or more license authorization codes.

  A feature or device support module is enabled when you register the license with ADSM. If the license is not registered, register the license by entering:

  ```
  register license nnnnnnnnnnnnnnnnnnnn
  ```

  Where *nnnnnnnnnnnnnnnnnnnn* is the license authorization code you were given.

## Task 1: Define the Library/MediumChanger (Robot) to the Operating System



1. As the root user, start the system management interface tool (SMIT) by entering:

   ```
   smit &
   ```

2. Select the following choices from SMIT:

   Devices
     ADSM Devices
       Library/MediumChanger
         Add Library/MediumChanger

3. Select “ADSM-SCSI-LB scsi ADSM Library” from the list.
4. Select the parent adapter to which you will be connecting the device.
5. Select the connection address.

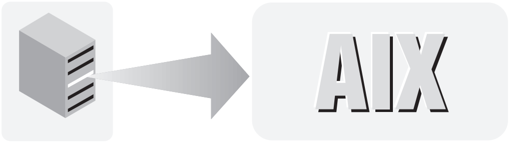
   **Note:** A connection address indicates the SCSI ID and logical unit number (LUN) used to address the device. The format is *xy*, where:

   | | |
   |---|---|
   | **x** | SCSI ID |
   | **y** | LUN |

   Typical SCSI devices have mechanical switches for setting these values. The value specified in SMIT must agree with the physical setting of the switch on the SCSI device. For example, the connection address of a device with the SCSI ID of 5 and the LUN of 0 is 50.

6. Select Do.
7. Upon completion, SMIT displays the device name in its output area. Write down the device name by which the operating system knows the device because that name is required in a later step. The device name in this scenario is lb0.
8. Exit from SMIT.

## Task 2: Define the Drive in the Library/MediumChanger to the Operating System



1. As the root user, start the system management interface tool (SMIT) by entering:

   ```
   smit &
   ```

2. Select the following choices from SMIT:

   Devices
   ADSM Devices
   Tape Drive
   Add a Tape Drive

3. Select “ADSM-SCSI-MT scsi ADSM Library” from the list.
4. Select the parent adapter to which you will be connecting the device.
5. Select the connection address.

   **Note:** A connection address indicates the SCSI ID and logical unit number (LUN) used to address the device. The format is *xy*, where:

   | | |
   |---|---|
   | **x** | SCSI ID |
   | **y** | LUN |

   Typical SCSI devices have mechanical switches for setting these values. The value specified in SMIT must agree with the physical setting of the switch on the SCSI device. For example, the connection address of a device with the SCSI ID of 5 and the LUN of 0 is 50.

6. Select Do.
7. Upon completion, SMIT displays the device name in its output area. Write down the device name by which the operating system knows the device because that identifier is required in a later step. The drive name in this scenario is mt2.
8. Exit from SMIT.

## Task 3: Define the Automated Library to ADSM

1. Define the automated library to ADSM.

   Issue the following command at the ADSM server command prompt:

   ```
   define library 8mmautolib1 libtype=scsi device=/dev/lb0
   ```

2. Define the drive in the automated library to ADSM.

   Issue the following command at the ADSM server command prompt:

   ```
   define drive 8mmautolib1 8mmautolibdrive1 device=/dev/mt2
   ```

3. Define an ADSM device class for the Library/MediumChanger.

   Issue the following command at the ADSM server command prompt:

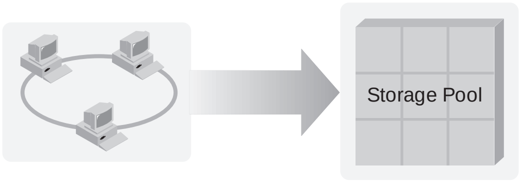
   ```
   define devclass 8mmautolibclass1 devtype=8mm library=8mmautolib1
   ```

4. Define a storage pool for Library/MediumChanger volumes.

   Issue the following command at the ADSM server command prompt:

   ```
   define stgpool 8mmautolibpool1 8mmautolibclass1 maxscratch=100
   ```

## Task 4: Define Policy that Links Client Data with Media for the Automated Library



This task describes the ADSM policy definitions that associate users on specific client nodes with automated library storage pools.

1. Define a policy domain.

   Issue the following command at the ADSM server command prompt to define a domain for client nodes that associates those nodes with a Library/Medium Changer storage pool:

   ```
   define domain adsm_development
   ```

2. Define a policy set.

   Issue the following command at the ADSM server command prompt to define a policy set for client nodes in a domain associated with a Library/Medium Changer:

   ```
   define policyset adsm_development aixpolicy
   ```

3. Define a management class.

   Issue the following command at the ADSM server command prompt to define a managment class for client nodes associated with a Library/MediumChanger storage pool:

   ```
   define mgmtclass adsm_development aixpolicy activefiles
   ```

4. Define a copy group.

   Issue the following command at the ADSM server command prompt to define a management class for client nodes to associate them with a Library/Medium Changer storage pool:

   ```
   define copygroup adsm_development aixpolicy activefiles STANDARD -
   type=backup destination=8mmautolibpool1
   ```

5. Assign a default management class.

   Issue the following command at the ADSM server command prompt to identify the default management class for this domain:

   ```
   assign defmgmtclass adsm_development aixpolicy activefiles
   ```

6. Activate the policy set for the automated library.

   Issue the following command at the ADSM server command prompt to identify the active policy set:

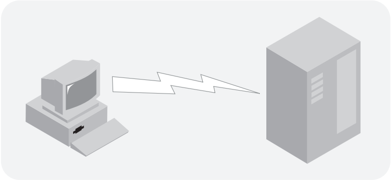
   ```
   activate policyset adsm_development aixpolicy
   ```

**Note:** For this scenario, an archive copy group is not necessary. If you see the message:

```
ANR1554W DEFAULT Management class ACTIVEFILES in policyset
ADSM_DEVELOPMENT AIXPOLICY does not have
an ARCHIVE copygroup:
files will not be archived by default if this set is activated.
Do you wish to proceed?  (Yes/No)
```

Enter Y and then press the Enter key.

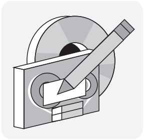
## Task 5: Register a Node to the Automated Library Policy Domain



Issue the following command at the ADSM server command prompt to register a node to use the new policy domain:

```
register node ecollins ecollins domain=adsm_development
```

## Task 6: Label Media for the Automated Library



Issue the following command at an operating system prompt (from an aixterm, a system console, or a tty session) to label a volume for a Library/Medium Changer:

```
dsmlabel -drive=/dev/mt2 -library=/devlb0 -overwrite -search
```

For each tape that is found, ADSM prompts for a label. When prompted for a label, provide a unique name of 6 characters or less. For example:

```
vol1
```

## Task 7: Add the Automated Library Volumes

Issue the following command at the ADSM server command prompt to add volumes to a Library/Medium Changer:

```
checkin libvolume 8mmautolib1 vol1 status=scratch
```

# Chapter 9. Managing Drives and Libraries

This chapter describes how to define and manage libraries and drives. In ADSM, a *library* is a collection of drives for which volume mounts are accomplished by using a common method, typically either manually or by robotic actions. A *drive* is a hardware device capable of performing operations on a specific type of sequential media. ADSM categorizes each drive using a device *device type* value that is based on the attributes of the hardware device.

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

| Section | Page |
|---|---|
| **Concepts:** | |
| How ADSM selects media and devices for storage and retrieval | 202 |
| Library types | 202 |
| Volume status codes | 208 |
| **Tasks:** | |
| Defining and managing libraries | 209 |
| Defining and managing drives | 211 |
| Managing storage volumes in automated libraries | 213 |
| Managing mount operations | 217 |
| **Note:** To configure tape and optical devices, and libraries, before informing the server about the devices, refer to *ADSM Installing the Server and Administrative Client*. | |

Most tasks presented in this chapter can be performed using either the graphical user interface or the command line interface. Table 7 on page 46 shows whether a task can be performed on the graphical user interface, the command line interface, or both.

For information about issuing ADSM commands, refer to *ADSM Administrator's Reference*. For help performing a task on the graphical user interface, see the procedure described in “Using Online Help” on page 20.

## How ADSM Selects Media and Devices for Data Storage and Retrieval

ADSM stores or retrieves data with the following steps:

1. ADSM selects a volume from the appropriate storage pool based on whether the data is being stored or retrieved.

   **Store** If the administrator defines a private volume, ADSM selects the volume by its volume name. If no private volumes are defined, and if the MAXSCRATCH parameter of the storage pool permits it, ADSM requests a scratch mount.

   **Retrieve** ADSM retrieves a volume by the volume name stored in the ADSM server database.

2. ADSM selects an available drive. Drive selection is based on information in the ADSM library object. The library object is located when ADSM checks the device class associated with the storage pool to determine the name of the library containing the drives that can be used for the operation.

3. ADSM initiates the volume mount based on the type of library associated with the drive.

   **Manual Library**
   ADSM displays mount request messages for an operator requesting that a specific volume (private volume in ADSM terms) or a scratch volume be mounted in the selected drive.

   **Automated Library**
   ADSM directs a robotic device to move a specific volume or a scratch volume from a storage slot into the selected drive.

4. ADSM dismounts the volume based on the type of library associated with the drive.

   **Manual Library**
   ADSM ejects the volume from the drive so that a mount operator can place it in an appropriate storage location.

   **Automated Library**
   ADSM interacts with a robotic device to move the volume from the drive back to its original storage slot in the library.

## ADSM Library Types

ADSM categorizes libraries by their *library type*. A library type denotes how volumes are mounted on the drives in that library. The supported library types that can be used are MANUAL, SCSI, 349X, and EXTERNAL. This section describes how to define and manage each library type.

### MANUAL Libraries

In a *MANUAL* library, an operator mounts the volumes. Define a MANUAL library if you have one or more standalone drives that are not part of an automated library. A manual library can have any number of standalone drives.

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 displayed on the server console. They are also sent to the administrative clients that were started by using the special *mount mode* parameter. Mount operators can then connect to the server from remote systems and monitor the server for required volume mount activities.

## SCSI Libraries

A Small Computer System Interface (SCSI) library is an automated library whose robot and drives are physically connected to the server system by using a SCSI bus. Refer to the *ADSM Installing the Server and Administrative Client* for a list of supported devices.

When you define a SCSI library, you must specify the device name of the robot. ADSM issues commands to the robot to mount and dismount volumes from drives within the library.

Because each SCSI library supports only a single device type, do not define drives with different device types in the same SCSI library.

**Note:** ADSM does not support cleaning cartridges on SCSI libraries. The cleaning slots, or fixed slots, are used for data cartridges.

If you have multiple devices of the types listed above, you must define one library to ADSM for each device. For example, if you have two Exabyte EXB-120 libraries, or one EXB-120 and one EXB-10e, you must define two libraries.

In some cases, operators may have to attend to an automated SCSI library. This action can occur, for example, if the library access door is inadvertently left open, and prevents the robot from mounting or dismounting volumes. If the server detects such a situation, it issues a prompt to the server console describing the required action. Prompts are also sent to administrative sessions that were started by using the special *mount mode* parameter, so that operators can work remotely and still be informed when manual intervention is required for a SCSI library.

## 349X Libraries

A 349X library is one of the following automated devices:

- IBM 3494 Tape Library Dataserver
- IBM 3495 Tape Library Dataserver

When you define a 349X library to the ADSM server, you must specify the device name of one or more *library management control points* (LMCP). Each LMCP provides an independent interface to the robot mechanism within a given 349X library. Refer to the *IBM SCSI Tape Drive, Medium Changer, and Library Device Drivers Installation and User's Guide* or the *IBM Parallel and ESCON Channel Tape Attachment/6000 Installation and User's Guide* for details. It is typically sufficient to specify the device name of only one LMCP; however, ADSM accepts up to eight such names when the library is defined or updated.

When a volume is to be mounted in a drive that is in a 349X library, ADSM interacts with the robot, by using a defined LMCP, to move the volume from its storage location in the library to the desired drive. Similarly, when ADSM finishes accessing a volume, the LMCP is used to inform the robot to move the volume back to its storage location.

## 349X Categories

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

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

When a volume is first inserted into the library, either manually or automatically at the convenience I/O station, the volume is assigned to the *insert* category (X'FF00'). A software application, such as ADSM, can then interact with the library control unit to change a volume's category number to a different value. To do this for ADSM, use the CHECKIN LIBVOLUME command (see "Informing the Server about New Volumes" on page 213).

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

***Without 3590 Support Enabled:*** An ADSM server reserves two different category numbers in each 349X library that it accesses. One of these categories is used for private volumes that belong to ADSM, and the other is used for scratch volumes that belong to ADSM. See "Volume Status: Scratch or Private" on page 208 for more information on private and scratch volumes.

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

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

***With 3590 Support Enabled:*** The ENABLE3590LIB server option supports the use of 3590 tape drives within 349X libraries. If support for 3590 drives is enabled for the server, the server reserves three different categories in the 3494 library. The categories are: private, scratch for 3490 drives, and scratch for 3590 drives.

**Attention:** If you are currently sharing a 3494 library between ADSM and other applications or systems, be careful when enabling 3590 support to prevent loss of data. See "Procedure for Existing 3494 Libraries" on page 205.

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

```
define library my3494 libtype=349x device=/dev/lmcp0 -
privatecategory=400 scratchcategory=401
```

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

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

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

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

For information on the ENABLE3590LIB server option, refer to *ADSM Installing the Server and Administrative Client*.

***Procedure for Existing 3494 Libraries:*** If you are currently sharing a 3494 library between ADSM and other applications or systems and you enable 3590 support, the new category that ADSM creates for 3590 scratch volumes can duplicate a category already assigned to another application and cause loss of data. To prevent loss of data, do either one of the following before enabling 3590 support:

- Update other applications and systems to ensure that there will be no conflicting use of category numbers.
- Delete the existing ADSM library definition and then define it again using a new set of category numbers that do not conflict with categories used by other systems or applications using the library. Do the following:
  1. Use an external utility (such as mtlib) to reset all of the ADSM volumes to the insert category.
  2. Delete the 3494 library from ADSM.
  3. Define the 3494 library to ADSM again, using new category numbers.
  4. Enable 3590 support by adding the following line to the server options file (dsmserv.opt):

     ```
     ENABLE3590LIBRARY  YES
     ```

Stop and start the server to make this change effective.

5. Check in the ADSM volumes you put in the insert category in step 1 on page 205. For 3490 volumes, use the following command:

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

For 3590 volumes, use the following command:

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

## External Libraries

An EXTERNAL library is managed by an external media management system. ADSM provides an interface that allows external media management systems to operate in conjunction with the ADSM server. To use the interface for one or more devices, you must define a library with library type EXTERNAL.

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

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

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

The EXTERNAL library type allows flexibility in grouping drives into libraries and storage pools. An EXTERNAL library may be one drive, a collection of drives, or even a partition of an automated library. For example, if you needed to handle the problem of duplicate volume serial numbers in a 3494 library that contains both 3490 and 3590 devices, you can define the library to ADSM as follows:

```
define library l3490 libtype=external externalmanager=/usr/bin/public/tmm
define library l3590 libtype=external externalmanager=/usr/bin/public/tmm

define devclass d3490 devtype=cartridge library=l3490 mountretention=2
define devclass d3590 devtype=3590 library=l3590 mountretention=2

define stgpool archivepool d3490 maxscratch=50
define stgpool backuppool d3590 maxscratch=50
```

For a definition of the interface that ADSM provides to the external media management system, see Appendix A, “External Media Management Interface Description” on page 413.

## Single Drive Libraries

ADSM does not automatically reclaim volumes from a library that contains only one drive because at least two drives in the same device class are required for reclamation. To reclaim a volume in a single drive library, use the MOVE DATA command. See "Moving Files from One Volume to Another Volume" on page 300 for more information.

## Special File Names for Devices

When a device configures successfully, a logical file name is returned in the form of op*x*, mt*x*, or lb*x*, where *x* is a numerical value that indicates the instance of a device for a particular class.

During configuration, a device special file name is created as one of the following. In this table, *x* denotes any integer of 0–9:

| **Special File Name** | **Description** |
|---|---|
| /dev/mt*x* | Used by all tape drives that are not supported by IBM hardware device drivers |
| /dev/lb*x* | Used by most ADSM-supported SCSI libraries |
| /dev/rop*x* | Used by all ADSM-supported optical drives |
| /dev/rmt*x* | Used by 3480, 3490, and 3590 hardware devices |
| /dev/rmt*x*.smc | Used to define the Automatic Cartridge Facility feature of the IBM 3590 B11 as a library |
| /dev/lmcp*x* | Used for 349X automatic tape libraries |

The following are examples of special file names:

*Table 16. Device Special File Names*

| **If the logical file name is:** | **The device special file name is:** |
|---|---|
| mt3 | /dev/mt3 |
| lb0 | /dev/lb0 |
| op1 | /dev/rop1 |
| rmt1 | /dev/rmt1.smc (3590 B11 ACF only) |
| rmt2 | /dev/rmt2 |
| lmcp0 | /dev/lmcp0 |

**Note:** You must know the device special file name when you use the DEFINE DRIVE or DEFINE LIBRARY commands. The special file name is the value provided for the DEVICE= parameter.

| ## Volume Status: Scratch or Private

The ADSM library volume inventory is used to keep track of the names and *status codes* of volumes that reside in automated libraries. Each volume in an automated library is categorized as either scratch or private.

### Scratch Volumes in a Library

A scratch volume is a volume that can be used to satisfy a storage pool scratch mount request. That is, when a scratch mount request is made in a given library, the server can choose *any* volume in the library whose status code indicates that it is a scratch volume. 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.

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. If a scratch volume that was dynamically allocated to a storage pool is deleted from that storage pool, it is automatically returned to the scratch pool for the associated library. This process allows other storage pools to use the volume in the future.

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

| If you are using a 3495 library, and the drives in the library are equipped with automatic cartridge loaders, performance improves if you use scratch volumes. The ADSM server always instructs the library device to preload scratch volumes into the automatic cartridge loaders of drives within the library. When a scratch volume is to be mounted, no robotic intervention is required because the automatic cartridge loader can be quickly indexed to mount a volume. By eliminating the time needed for the robot to get a volume from a storage slot, the overall mount time can be greatly reduced.

### Private Volumes in a Library

A private volume is a volume that cannot be used to satisfy scratch mount requests for a given library. To mount a volume whose status code is private, you must provide the volume name. Thus, for normal data storage operations, a DEFINE VOLUME command must be used to define the volume to the storage pool that is associated with a given library. If you are doing database dump/load or import/export operations, you must explicitly identify the volume names of any volumes in private status in a library.

You can use the private category for volumes if you need to carefully regulate which volumes are used by individual storage pools in your environment.

Note that 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 the data on the volume expires or is moved to other volumes. However, if a scratch volume is used for an export or a database dump operation, its status code changes to private and it is not automatically returned to the scratch pool. When an administrator determines that the volume's data is no longer needed, the UPDATE LIBVOLUME command is used to return the volume to the library scratch pool.

## Defining and Managing Libraries

The following table lists the tasks in this section and the required privilege class.

| Task | Required Privilege Class |
|---|---|
| Define, update, and delete libraries | System or unrestricted storage |
| Query libraries | Any administrator |

### Defining Libraries

Before you can use a drive that requires either manual or robotic mounting, you must define the library to which the drive belongs. This is true even for standalone drives. To define a new library, use the DEFINE LIBRARY command.

All automatic libraries require a device name string. See Table 16 on page 207 for a list of all possible device name strings. These special file device name strings are used in the examples that follow.

This example can apply to any manual library. Suppose you have several standalone tape drives that must be mounted manually by an operator. You can define a library named MANUALMOUNT for these drives using the following command:

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

This example can apply to any SCSI library. This assumes that you have configured the robot device driver, as described in *ADSM Installing the Server and Administrative Client*, and determined the appropriate device name string as shown in the example. If you have an Exabyte EXB-120 device, you may define a library named ROBOTMOUNT using the following command:

```
define library robotmount libtype=scsi device=/dev/lb0
```

If you have an IBM 3590 B11 device with a logical file name of rmt0, you may define a library named MAINMOUNT using the following command:

```
define library mainmount libtype=scsi device=/dev/rmt0.smc
```

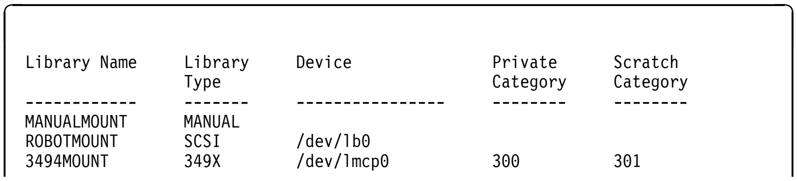
Suppose you have an IBM 3494 Tape Library Dataserver connected to your system, and that you have defined one LMCP whose device name is `/dev/lmcp0`. You can define a library named AUTOMOUNT using the following command:

```
define library automount libtype=349x device=/dev/lmcp0
```

## Querying Libraries

An administrator can query for 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 41 may be generated if the following command is issued:

```
query library
```



```
Library Name     Library     Device              Private      Scratch
                 Type                            Category     Category
------------     -------     ----------------    --------     --------
MANUALMOUNT      MANUAL
ROBOTMOUNT       SCSI        /dev/lb0
3494MOUNT        349X        /dev/lmcp0          300          301
```

*Figure 41. Standard Query Library Report*

## Updating Libraries

The only attribute that can be updated is the DEVICE attribute. This might be necessary if your system or device is reconfigured, causing the device name to change.

For example, if you previously defined a SCSI library named ROBOTMOUNT, but the device name was changed due to reconfiguration of your device, you can issue the following command to inform the ADSM server of the change:

```
update library robotmount device=/dev/lb1
```

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

## Deleting Libraries

Before issuing the DELETE LIBRARY command, all of the drives that have been defined as part of the library must be deleted. This process is described in “Deleting Drives” on page 213.

For example, you may want to delete a library named ROBOTMOUNT. After deleting all of the drives defined as part of this library, you can issue the following command to delete the library itself:

```
delete library robotmount
```

## Defining and Managing Drives

The following table lists the tasks in this section and the required privilege class.

| Task | Required Privilege Class |
|---|---|
| Define, update, and delete drives | System or unrestricted storage |
| Query drives | Any administrator |

## Defining Drives

To inform the server about a drive that can be used to access storage volumes, an administrator can issue the DEFINE DRIVE command. When issuing this command, you must provide the name of the library in which the drive resides, and the device name to be used to access the drive.

If you define a drive in a SCSI library that can hold more than one drive, you must supply the ELEMENT parameter to specify where the drive resides within the library. The element numbers are provided in the worksheet associated with the library, which was filled in when configuring the drive as described in *ADSM Installing the Server and Administrative Client*.

### Examples: Defining Drives

Most SCSI drives are supported by the ADSM device driver.

All tape drives require a device name string. See Table 16 on page 207 for a list of all possible device name strings. These special file device name strings are used in the examples that follow.

If you have defined an Exabyte EXB-120 device as a library named *robotmount*, you can define a drive as follows:

```
define drive robotmount drive_1 device=/dev/mt0
```

If you configured an IBM 3480, 3490, or 3590 drive with the name *mainmount*, you can define a drive as follows:

```
define drive mainmount drive6 device=/dev/rmt0
```

If you defined an optical device as a library named *optmount*, you can define a drive as follows:

```
define drive optmount drive2 device=/dev/rop0
```

## Querying Drives

Use the QUERY DRIVE command to display information about a drive located in a server-attached library.

**Note:** This command accepts wildcard characters for both a library name and a drive name.

Information is available in the two following formats:

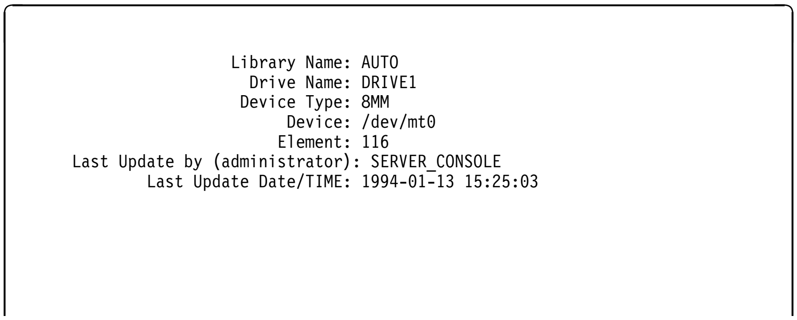
**Standard** Specifies that partial information is displayed for the drive.

**Detailed** Specifies that complete information is displayed for the drive.

For example, to display detailed information about the 8mm drive named DRIVE1 located in the library named AUTO, enter the following command:

```
query drive auto drive1 format=detailed
```

Figure 42 shows an example of the output from this command:



```
                   Library Name: AUTO
                     Drive Name: DRIVE1
                    Device Type: 8MM
                         Device: /dev/mt0
                        Element: 116
 Last Update by (administrator): SERVER_CONSOLE
          Last Update Date/TIME: 1994-01-13 15:25:03
```

*Figure 42. Example of Output from a Detailed Query Drive Command*

## Updating Drives

The UPDATE DRIVE command can be used to change the DEVICE attribute of a drive. This action may be necessary if your system or device has been reconfigured. If the drive resides within a SCSI library, its ELEMENT attribute can also be updated.

A drive cannot be updated if it is currently in use. If a drive has a volume mounted, but the volume is currently idle, it can be explicitly dismounted as described in "Dismounting an Idle Volume" on page 218.

## Deleting Drives

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 218.

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

# Managing Storage Volumes in Automated Libraries

The following table lists the tasks in this section and the required privilege class.

| Task | Required Privilege Class |
|---|---|
| Inform the server when a new volume is available<br>Remove a volume from an automated library<br>Change the status of a volume in an automated library<br>Restore the inventory to a consistent state | System or unrestricted storage is the required privilege class for all the tasks. |

Before ADSM can use volumes that reside in an automated library, you must explicitly tell the server about these volumes by using the CHECKIN LIBVOLUME command. The server then keeps track of them in a *library volume inventory* that is part of the database. This inventory is different from the inventory of volumes defined for storage pools.

The library contains an inventory of cartridges, but volume data is tracked by the storage pool. Removing a cartridge from the library does not remove data from the storage pool. When the data is required, check the cartridge back into the library to read the data.

## Informing the Server about New Volumes

You can inform the server when a new volume is available. This process is called *checking in* the volume and can be done by issuing the CHECKIN LIBVOLUME command. When a volume is checked in, the server adds the volume to its library volume inventory. When issuing the CHECKIN LIBVOLUME command, you must specify the volume status code (PRIVATE or SCRATCH) to be assigned to the new volumes.

When using the CHECKIN LIBVOLUME command, be prepared to supply the following information:

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

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

**Status**
Specifies the status that is assigned to the storage volume being checked in.

**Check label**
Specifies whether ADSM should read sequential media labels of volumes during CHECKIN command processing.

| For optical volumes being checked in to an automated library, you must specify
| CHECKLABEL=YES. ADSM must read the label to determine whether the
| volume is rewritable (OPTICAL device type) or write-once read-many (WORM
| device type).

**Swap**
Specifies whether ADSM will initiate a swap operation when an empty slot is not available during CHECKIN command processing.

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

**Search**
Specifies whether ADSM searches the library to see if the volume has been previously checked into the library. The options are:

**No**
This is the default. Specify this option if you want to check in only a single volume that is not currently in the library. When using this option, ADSM requests that the mount operator load the volume in the entry/exit port of the library. If the library does not have an entry/exit port, ADSM requests that the mount operator load the volume into a slot within the library.

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

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

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

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

- Insert
- PRIVATECATEGORY (specified when the library was defined to ADSM)
- SCRATCHCATEGORY (specified when the library was defined to ADSM)

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

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

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

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

If you check in a volume that has already been defined in a storage pool, you must use a volume status of PRIVATE. This process ensures that the volume is not overwritten when a scratch mount is requested. The server rejects any attempt to check in such a volume using the SCRATCH status.

When you check in a volume, the server tries to read the media label unless the option not to read the label is specified. If the volume is not properly labeled, it cannot be checked in. See “Labeling Sequential Storage Pool Volumes” on page 288 for information on how to label new volumes. The option not to read the label is specified by adding the parameter CHECKLABEL=NO to the CHECKIN LIBVOLUME command.

The process time can vary because the CHECKIN LIBVOLUME command involves device access. For this reason, the command always executes as a background process.

## Removing Volumes from a Library

You may want to remove a volume from an automated library. The following are examples of such situations:

- 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 to make room for new volumes that can be used to store more data.

To remove a volume from an automated library, an administrator can 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 slot within the library.

For CHECKOUT LIBVOLUME operations in SCSI libraries, the server always mounts the volume and checks its label. This process ensures that the correct volume is removed from the library. If the server is unable to read the label, and you want to force its removal, you can use the FORCE=YES option on the CHECKOUT LIBVOLUME command. The only exception to this process is if the CHECKLABEL=NO parameter is specified on the CHECKOUT LIBVOLUME command.

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. To make the volume available again, an administrator can check the volume back into the library and then update the volume's ACCESS value.

When using the CHECKOUT LIBVOLUME command, be prepared to supply the following information:

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

**Volume name**
Specifies the volume name of the storage volume being checked out.

**Check label**
Specifies whether ADSM reads sequential media labels of volumes during CHECKOUT command processing.

**Force**
Specifies whether ADSM checks out a storage volume if there is an input/output (I/O) error reading the label.

**Remove**
Specifies whether ADSM ejects a volume during CHECKOUT command processing from either an IBM 3494 or 3495 library.

## Changing the Status of a Volume in a Library

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 may result in the volume being overwritten during an ensuing scratch mount request.

## Auditing a Library's Volume Inventory

The AUDIT LIBRARY command is used to ensure that an automated library is in a consistent state with respect to the server's internal volume inventory.

If the server's library volume inventory develops inconsistencies due to manual intervention or movement of volumes within the library or to problems with the server database, the AUDIT LIBRARY command can be used to restore the inventory. The AUDIT LIBRARY command restores the volume inventory to a consistent state that reflects the current state of the library.

When auditing a SCSI library, the server tries to read the label for each volume that is found in a slot in the library. The results of this search are then used to rebuild the server's inventory for the library. If inconsistencies are detected and repaired, the server provides messages that describe the actions taken. The time to complete can vary because this process involves significant movement activity.

**Note:** Audit library processing waits until all volumes are 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).

### Special Considerations for SCSI Libraries

The ADSM server's library volume inventory for SCSI libraries is used to track the storage location of each volume that has been checked into the library. If you open the access door and manually move volumes from one slot to another or remove volumes from the library, the server's inventory is disturbed. Consequently, avoid such activity whenever possible. If you must move volumes within the library, issue an AUDIT LIBRARY command so that the server can reevaluate the status of the storage slots in the library and rebuild the inventory.

## Managing Mount Operations

If operator intervention is required for a device, the server writes a request message to the server console. Such messages are also sent to administrative sessions that were started with the *mount mode* parameter.

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

For most types of requests, such as volume mounts, the server detects when the operator performs the action. 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. In such cases, the message that is displayed by the server requests that the operator reply when the activity has been completed.

The following table lists the tasks in this section and their required privilege class.

| Task | Required Privilege Class |
|---|---|
| Query operator requests or mounted volumes | Any administrator |
| Reply to or cancel operator requests | Operator |
| Request a volume dismount | Operator |

## Querying Pending Operator Requests

Any administrator can issue the QUERY REQUEST command to obtain a report of all outstanding operator requests. In addition to displaying the requested action, the output of this command indicates the amount of time remaining before the request times out.

## Replying to Operator Requests

If the server requests that you reply when you complete an action, use the REPLY command. The only parameter for this command is the request identification number that tells the server which request has been completed. This three-digit number is always displayed as part of the request message. It can also be obtained by issuing a QUERY REQUEST command, as described in “Querying Pending Operator Requests.”

## Canceling an Operator Request

If a mount request 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 “Querying Pending Operator Requests.”

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.

## Determining Which Volumes are Mounted

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

After a volume becomes idle, the server may keep it mounted for a while to reduce the access time it if it is needed again. An administrator can explicitly request that such a volume be dismounted by issuing the DISMOUNT VOLUME command.

# Chapter 10. Managing Storage Devices

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:

| | |
|---|---|
| **4MM** | 4mm tape drives |
| **8MM** | 8mm tape drives |
| **DLT** | Digital linear tape (DLT) drives |
| **QIC** | Quarter-inch cartridge tape drives |
| **3590** | IBM 3590 tape drives |
| **CARTRIDGE** | Cartridge tape drives, such as IBM 3480, 3490, and 3490E drives |
| **OPTICAL** | Optical drives that use 5.25-inch rewritable optical cartridges |
| **WORM** | Optical drives that use 5.25-inch write-once ready-many optical cartridges |
| **FILE** | Storage volumes that are actually 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 device classes for tape (4MM, 8MM, DLT, QIC, 3590, and CARTRIDGE) | 221 |
| Defining device classes for optical drives (OPTICAL and WORM) | 230 |
| Defining FILE device classes | 231 |
| Requesting information about device classes | 233 |
| Updating device classes | 234 |
| Deleting device classes | 235 |

Most tasks presented in this chapter can be performed by using either the graphical user interface or the command-line interface. Table 8 on page 48 shows whether a task can be performed on the graphical user interface, the command-line interface, or both.

For information about issuing ADSM commands, refer to *ADSM Administrator's Reference*. For help performing a task on the graphical user interface, see the procedure described in "Using Online Help" on page 20.

## Defining Device Classes for Sequential Media

| Task | Required Privilege Class |
|---|---|
| Define device classes | System or unrestricted storage |

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

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

- 4MM
- 8MM
- DLT
- QIC
- 3590
- CARTRIDGE
- OPTICAL
- WORM

See Chapter 9, "Managing Drives and Libraries" on page 201 for information about defining drives and libraries.

Use the DEFINE DEVCLASS command to define device classes. For example, you may want to define a device class for a storage pool with characteristics similar to these:

- The device class is named 8MMTAPE and used for an 8mm device
- The 8mm device is in a library name AUTO
- The format is DRIVE
- The mount limit is 2
- The mount retention is 10
- The tape volume prefix is named ADSMVOL
- The estimated capacity is 6GB

To define a device class with these characteristics, enter the following:

```
define devclass 8mmtape devtype=8mm library=auto format=drive -
mountlimit=2 mountretention=10 prefix=adsmvol estcapacity=6g
```

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

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

## Defining Device Classes for Tape

To use 4mm, 8mm, DLT, quarter-inch, IBM 3590, or IBM 34xx tape drives and cartridges, you must define a device class whose device type is 4MM, 8MM, DLT, QIC, 3590, or CARTRIDGE, respectively. Do this by issuing a DEFINE DEVCLASS command with the DEVTYPE parameter. Other parameters specify how to manage data storage operations involving the new device class:

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

### Mount Limit

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

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 248 and “Space Reclamation for Sequential Access Storage Pools” on page 257 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 backup database (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, you should increase your mount limit or review your scheduling of operations to reduce the contention for drives.

## Mount Wait Period

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

## Mount Retention Period

The *MOUNTRETENTION* parameter specifies the amount of time that a mounted volume should remain mounted after its last I/O activity. If this idle time limit is reached, the server dismounts the volume.

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.

The default mount retention period is 60 minutes; the maximum value for this parameter is 9999 minutes.

## Tape Label Prefix

The *PREFIX* parameter specifies 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 values are supported for the tape device classes:

***4MM Device Classes***

| Recording Format | Description |
|---|---|
| DDS1 | Specifies that ADSM writes data using the DDS-1 recording format, without compression. The result is an uncompressed cartridge capacity of 1.3GB on 60 meter tapes and 2.0GB on 90-meter tapes. This format only applies to 60- and 90-meter tapes. |
| DDS1C | Specifies that ADSM writes data using the DDS-1 recording format, with compression enabled. The result is a cartridge capacity of approximately 1.3GB on a 60-meter tape, or 2.0GB on a 90-meter tape. However, this format uses the tape drive’s hardware compression feature, and the actual capacity may be greater, depending on the effectiveness of compression. |
| DDS2 | Specifies that ADSM writes data using the DDS-2 recording format, without compression. The result is an uncompressed cartridge capacity of 4.0GB on 120-meter tapes. This format only applies to 120-meter tapes. |

| Recording Format | Description |
|---|---|
| DDS2C | Specifies that ADSM writes data using the DDS-2 recording format, with compression. The result is an uncompressed cartridge capacity of 8.0GB on 120-meter tapes. However, this format uses the tape drive's hardware compression feature, and the actual capacity may be greater, depending on the effectiveness of compression. This format only applies to 120-meter tapes. |
| DRIVE | Lets the server select the recording format to use based on the drive on which the volume is mounted. |

### *8MM Device Classes*

| Recording Format | Description |
|---|---|
| 8200 | Basic recording format for an 8mm tape drive. This format yields a capacity of approximately 2.3GB on a standard 112-meter tape cartridge. |
| 8200C | Specifies that ADSM writes data using the 8200C recording format. The result is a tape capacity of approximately 2.3GB when using standard 112-meter tape cartridges. However, this format uses the tape drive's hardware compression feature, and the actual capacity may be greater than 2.3GB, depending on the effectiveness of the compression feature. |
| 8500 | Enhanced recording format for 8mm tape drives. This format yields a capacity of approximately 5.0GB on a standard 112-meter tape cartridge. |
| 8500C | Enhanced recording format, with hardware compaction, for 8mm tape drives. |
| DRIVE | Lets the server select the recording format to use based on the drive on which the volume is mounted. |

### *DLT Device Classes*

| Recording Format | Description |
|---|---|
| DLT10 | Specifies that ADSM writes data using the DLT10 recording format. The result is a capacity of approximately 10GB on CompacTape III or CompacTape IV tape cartridges. |
| DLT10C | Specifies that ADSM writes data using the DLT10C recording format. The result is a capacity of approximately 10GB on CompacTape III or CompacTape IV tape cartridges. However, this format uses the tape drive's hardware compression feature, and the actual capacity may be greater than 10GB, depending on the effectiveness of the compression feature. |

| Recording Format | Description |
|---|---|
| DLT20 | Specifies that ADSM writes data using the DLT20 recording format. The result is a capacity of approximately 20GB on CompacTape IV tape cartridges.<br>**Attention:** Use this format only with CompacTape IV cartridges in a DLT4000 drive. |
| DLT20C | Specifies that ADSM writes data using the DLT20 recording format. The result is a capacity of approximately 20GB on CompacTape IV tape cartridges. However, this format uses the tape drive's hardware compression feature, and the actual capacity may be greater than 10GB, depending on the effectiveness of the compression feature.<br>**Attention:** Use this format only with CompacTape IV cartridges in a DLT4000 drive. |
| DRIVE | Lets the server select the recording format to use based on the drive on which the volume is mounted. |

### *QIC Device Classes*

| Recording Format | Description |
|---|---|
| QIC120 | Specifies that ADSM writes data using the QIC120 recording format. The result is a cartridge capacity of 26MB–172MB. |
| QIC150 | Specifies that ADSM writes data using the QIC150 recording format. The result is a cartridge capacity of 31MB–207MB. |
| QIC525 | Specifies that ADSM writes data using the QIC525 recording format. The result is a cartridge capacity of 65MB–427MB. |
| QIC1000 | Specifies that ADSM writes data using the QIC1000 recording format. The result is a cartridge capacity of 169MB–1.1GB. |
| DRIVE | Lets the server select the highest recording format to be used based on the drive on which the volume is mounted. |

### *3590 Device Classes*

| **Recording Format** | **Description** |
|---|---|
| 3590B | Specifies that ADSM writes data using the basic (uncompressed) recording format. The result is a cartridge capacity of 10GB. |
| 3590C | Specifies that ADSM writes data using the compressed recording format. The result is a cartridge capacity of 10GB. This format uses the tape drive's hardware compression feature, which may allow for greater capacity depending on the effectiveness of the compression. |
| DRIVE | Lets the server select the highest recording format to be used based on the drive on which the volume is mounted. |

### *CARTRIDGE Device Classes*

| **Recording Format** | **Description** |
|---|---|
| 3480 | 18-track basic recording format |
| 3480XF | 18-track compacted recording format |
| 3490B | 36-track basic recording format |
| 3490C | 36-track compacted recording format |
| DRIVE | Lets the server select the recording format to be used based on the drive on which the volume is mounted. |

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 4mm devices, one capable only of DDS-1 format and the other capable of DDS-2 format. The server might select the high-density recording format of DDS-2 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.

Table 17 on page 227 shows the read/write capabilities of all supported QIC tape drives.

*Table 17. QIC Tape Recording Format Selections*

| Tape/Format | QIC-120 | QIC-150 | QIC-525 | QIC-1000 |
|---|---|---|---|---|
| 3M DC300XLP | – | – | – | – |
| 3M DC600A | Read | – | – | – |
| 3M DC600XTD | Read/Write | Read/Write | – | – |
| 3M DC6150 | Read/Write | Read/Write | – | – |
| 3M DC6320 | Read/Write | Read/Write | Read/Write | – |
| 3M DC6525 | Read/Write | Read/Write | Read/Write | – |
| 3M DC9100 | – | – | – | Read/Write |
| 3M DC9120XL | – | – | – | Read/Write |

**Note:** Tapes 3M DC300XLP and 3M DC600A cannot be used by ADSM.

Table 18 shows the read/write capabilities of all supported CARTRIDGE tape drives.

*Table 18. Recording Formats Supported by CARTRIDGE Tape Drives*

| Recording Format | Tape Drive | | |
|---|---|---|---|
| | 3480 | 3490 | 3490E |
| 3480 | Read/Write | Read/Write | Read |
| 3480XF | Read/Write | Read/Write | Read |
| 3490B | – | – | Read/Write |
| 3490C | – | – | Read/Write |

Refer to the *ADSM Installing the Server and Administrative Client* for specific device support information.

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.

## Estimated Capacity Value

ADSM estimates the capacity of the volumes in a storage pool based on the parameters assigned to the device class that is associated with the storage pool. The estimated capacity value is used by ADSM when making decisions about when to initiate a reclamation process for volumes in the storage pool. It is also used to generate storage pool and volume reports. As a volume is written and filled, the server determines the actual capacity of the volume and uses this instead of the estimated capacity.

You can either accept the default estimated capacity value for a given device class or explicitly specify an estimated capacity that you want the server to use instead of the default.

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.

### 4MM Device Classes

*Table 19. Estimated Capacity for 4mm Tape Volumes*

| Recording Format | Estimated Capacity |
|---|---|
| DDS1 | 1.3GB (60-meter tape) or 2.0GB (90-meter tape) |
| DDS1C * | 1.3GB (60-meter tape) or 2.0GB (90-meter tape) |
| DDS2 | 4.0GB (120-meter tape) |
| DDS2C * | 4.0GB (120-meter tape) |

**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.

### 8MM Device Classes

*Table 20. Estimated Capacity for 8mm Tape Volumes*

| Recording Format | Estimated Capacity |
|---|---|
| 8200 | 2472MB |
| 8200C * | 2472MB |
| 8500 | 5.0GB |
| 8500C * | 5.0GB |

**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.

### DLT Device Classes

*Table 21. Estimated Capacity for DLT Volumes*

| Recording Format | Estimated Capacity |
|---|---|
| DLT10 | 10GB |
| DLT10C * | 10GB |
| DLT20 | 20GB |
| DLT20C * | 20GB |

**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.

### QIC Device Classes

*Table 22. ADSM Default Estimated Capacity for QIC Tape*

| Tape Format | Estimated Capacity (Range) |
|---|---|
| QIC120 | 26MB–172MB |
| QIC150 | 31MB–207MB |
| QIC525 | 65MB–427MB |
| QIC1000 | 169MB–1.1GB |

### 3590 Device Classes

*Table 23. ADSM Default Estimated Capacity for 3590 Tape*

| Tape Format | Estimated Capacity (Range) |
|---|---|
| 3590B | 10GB |
| 3590C * | 10GB |

**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 24. Estimated Capacity for CARTRIDGE Tape Volumes*

| Recording Format | Media Type | Estimated Capacity |
|---|---|---|
| 3480 | CST | 180MB |
| 3480XF * | CST | 180MB |
| 3490B | CST | 360MB |
| 3490C * | CST | 360MB |
| 3490B | ECCST | 720MB |
| 3490C * | ECCST | 720MB |

**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.

### 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 Device Classes for Optical Drives

To use rewritable or WORM optical media on 5.25-inch optical drives, such as the IBM &ibm351., you must define a device class whose device type is OPTICAL or WORM. Use the DEVTYPE=OPTICAL parameter to define a device class that uses rewritable optical media. Use the DEVTYPE=WORM parameter to define a device class that uses write-once optical media. Several other parameters can be supplied when the command is issued to instruct ADSM on how to manage data storage operations involving the new device class. These parameters are described below.

### Mount Limit

The mount limit value for OPTICAL and WORM device classes is used as described in “Mount Limit” on page 221.

### Mount Wait Period

The mount wait period for OPTICAL and WORM device classes is used as described in “Mount Wait Period” on page 222.

### Mount Retention Period

The mount retention period for OPTICAL and WORM device classes is used as described in “Mount Retention Period” on page 222.

### Recording Format

You can use the *FORMAT* parameter to specify the recording format used by ADSM when writing data to an optical cartridge. The following values are supported for OPTICAL and WORM device classes:

| Format | Description |
|---|---|
| 650MB | Used with optical cartridges whose capacity is 650MB. |
| 1300MB | Used with optical cartridges whose capacity is 1300MB. |
| DRIVE | Allows the server to select the recording format to be used based on the capabilities of the drive used to access the optical disk. |

Use the FORMAT=DRIVE parameter only if all drives that can be accessed by the device class are identical. If you specify DRIVE for a device class that has incompatible devices, mount errors can occur.

### Estimated Capacity Value

ADSM estimates the capacity of the volumes in a storage pool based on the parameters assigned to the device class that is associated with the storage pool. The estimated capacity value is used by ADSM when making decisions about when to initiate a reclamation process for volumes in the storage pool. It is also used to generate storage pool and volume reports. As a volume is written and filled, the server determines the actual capacity of the volume and uses this instead of the estimated capacity.

You can either accept the default estimated capacity value for a given device class or explicitly specify an estimated capacity that you want the server to use instead of the default.

### Library

The LIBRARY parameter for OPTICAL and WORM device classes is used as described in “Library” on page 230.

## Defining FILE Device Classes

The FILE device type is used for special device classes whose storage volumes are not physical units, such as tape or optical cartridges, but *simulated* storage volumes. Data is written sequentially into standard files in the file system of the server machine.

**Note:** Because each volume is actually a file, a volume name is a fully qualified file name string.

When you define the FILE device class, you can supply additional parameters. These parameters are used to instruct ADSM how to manage data storage operations involving the new device class. These parameters are described below.

### 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 data storage and retrieval operations. Any attempts to access more volumes than indicated by the mount limit causes the requester to wait.

See “Mount Limit” on page 221 for a discussion of the considerations necessary in determining an appropriate mount limit value for the new device class.

### Maximum Capacity

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 tries to store any new data on a different volume.

The default MAXCAPACITY value for a FILE device class is 4MB.

### Directory for Volumes

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

### Example: Defining a File Device Class

You can define this device class by issuing a DEFINE DEVCLASS command with the DEVTYPE=FILE parameter. For example, to define a device class named PLAINFILES with a device type of FILE and a maximum capacity of 50MB, enter the following:

```
define devclass plainfiles devtype=file maxcapacity=50m
```

## Requesting Information about a Device Class

| Task | Required Privilege Class |
|---|---|
| Request information about device classes | Any administrator |

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

```
query devclass
```

Figure 43 is an example of a standard report for device classes.

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

*Figure 43. 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 44 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): DSMADMIN
         Last Update Date/Time: 01/05/1994 16:02:13
```

*Figure 44. Example of a Detailed Device Class Report*

## Updating Device Classes

| Task | Required Privilege Class |
|---|---|
| Update device class information | System or unrestricted storage |

Use the UPDATE DEVCLASS command to update a defined device class. You can use this command to modify parameters for the specified device class. If you do not explicitly update a parameter, the parameter remains unchanged. The supported parameters for this command vary depending on the device type. If you include the DEVCONFIG option in the dsmserv.opt file, the files you specify with that option are automatically updated with the results of this command. When you use this option, the files specified are automatically updated whenever a device class, library, or drive is defined, updated, or deleted.

For example, if you would like to update a device class named 8MMTAPE and change the mount retention to 15 minutes, enter the following:

```
update devclass 8mmtape mountretention=15
```

Refer to *ADSM Administrator's Reference* for more detailed information regarding the UPDATE DEVCLASS command.

## Deleting Device Classes

| Task | Required Privilege Class |
|---|---|
| Delete a device class | 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 280.
- The device class is not being used by a server process such as export or import.

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

# Chapter 11. 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 sections listed in the following table begin at the indicated pages.

| Section | Page |
|---|---|
| **Concepts:** | |
| Storage pools | 238 |
| Assigning volumes in storage pools | 244 |
| Storage pool hierarchy | 245 |
| Server migration of files | 247 |
| Damaged files | 251 |
| Cache on disk storage pools | 252 |
| Collocation on sequential access storage pools | 253 |
| Space reclamation on sequential access storage pools | 257 |
| **Tasks:** | |
| Estimating space needs for storage pools | 262 |
| Defining or updating storage pools | 262 |
| Using copy storage pools to improve data availability | 267 |
| Backing up storage pools | 269 |
| Monitoring the use of storage pools | 271 |
| Deleting storage pools | 280 |
| Restoring storage pools | 281 |

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 49 shows whether a task can be performed on the graphical user interface, the command line interface, or both.

For information about issuing ADSM commands, refer to *ADSM Administrator's Reference*. For help performing a task on the graphical user interface, see the procedure described in “Using Online Help” on page 20.

## Storage Pools

The types of storage pools are:

**Primary storage pool**
Data that is backed up, archived, or migrated from a client node is stored in a primary storage pool. See “Primary Storage Pools” for details.

**Copy storage pool**
When an administrator backs up a primary storage pool, the data is stored in a copy storage pool. See “Copy Storage Pools” on page 239 and “ Backing Up Storage Pools” on page 269 for details.

Storage pools are categorized by the access method of the volumes contained in the storage pools:

**Random access**
Associated with the DISK device class

**Sequential access**
Associated with sequential device classes, for example, CARTRIDGE

ADSM has three predefined random access storage pools:

**ARCHIVEPOOL** Contains files archived from client nodes

**BACKUPPOOL** Contains files backed up from client nodes

**SPACEMGPOOL** Contains files migrated from client nodes

Although ADSM does not require a separate storage pool for migration, a separate storage pool is recommended. See “Server Migration of Files” on page 247 for more information about migration.

Figure 45 shows an example of data storage. In this example, the data storage defined for the server includes four storage pools: one disk storage pool, one tape cartridge storage pool, one optical storage pool, and one 8mm storage pool.

*Figure 45. Example of Data Storage*

## Primary Storage Pools

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. The primary storage pool is the default storage pool type. Primary storage pool volumes never have an access mode value of offsite. This means that the primary storage pool volumes are always located onsite.

A primary storage pool can use random access or sequential access strategy.

## Copy Storage Pools

Copy storage pools store copies of data backed up from primary storage pools. Multiple primary storage pools can use one copy storage pool. Primary storage pools can also use multiple copy storage pools if multiple copies are deemed 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.

**Note:** A primary storage pool file must always exist for a copy storage pool file to exist. When a primary storage pool file is deleted so are the copies of the files in the copy storage pools.

Copy storage pool volumes can have an access mode value of OFFSITE. The offsite designation allows those volumes to be moved to an offsite location and still be tracked by the database. If the ACCESS=OFFSITE parameter is not specified, the copy storage pool volumes will remain at the onsite location. Locating copy storage pool volumes offsite provides a means of recovering from an onsite disaster.

The copy storage pool concept provides a means of recovering from a disaster, data-integrity errors, or media failure. See “Restoring Storage Pools” on page 281, “Using Copy Storage Pools to Improve Data Availability” on page 267, or “Recovery from Media Loss” on page 354 for details.

Copy storage pools cannot be used as:

- The next storage pool in a migration hierarchy
- A management class destination for space-managed files
- A copy group destination for backed up or archived files

Copy storage pools do not have migration attributes. No hierarchy exists for the migration of file copies that are stored in copy storage pools.

Files in a copy storage pool do not move when primary files are moved. For example, if a file moves within a primary storage pool as a result of reclamation or is migrated to another primary storage pool, the copy file in the copy storage pool does not move. Therefore, you should use the same copy storage pool for each primary storage pool in a given migration hierarchy. This avoids the accumulation of files in different copy storage pools as the primary files are migrated. If you have multiple, non-merging, migration hierarchies, different copy storage pools can be used for primary storage pools in different hierarchies.

Normally a file is obtained from the primary storage pool whenever an attempt is made to access a file. However, 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
- The primary file is stored on a volume that is UNAVAILABLE or DESTROYED
- The file is stored on an offline volume

ADSM also accesses files from a copy storage pool for certain operations (restore, retrieve, or recall of files to end users and export of file data) if the primary file is located in a storage pool that is UNAVAILABLE.

Although copy storage pools can contain only sequential access volumes and are not associated with the DISK device class, you can use disk devices for copy storage pools by specifying a device class that has a device type of FILE.

Copies in a copy storage pool are produced by using the BACKUP STGPOOL command. The copies are made incrementally while the server is operational and available to clients. Central scheduling allows for producing copies at convenient times for your installation.

## Restore Processing

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**
Command 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 281 for more detailed information.

**RESTORE VOLUME**
Command 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 306 for more detailed information.

ADSM uses database information to determine which files should be restored for a volume or storage pool, so restore processing does not require that the original volumes be accessed. For example, if a primary storage pool volume becomes damaged, the RESTORE VOLUME command could be used to recreate files that were stored on that volume, even though the volume itself is not readable. However, if the administrator were to delete the damaged files with DISCARDDATA=YES, the database reference to the files on the primary storage pool volume and all references to copies of the files on copy storage pool volumes, would be removed from the database. It would not be possible to restore those files.

Restore processing obtains files from a copy storage pool and stores these files on new primary storage pool volumes. Database references to files on the original primary storage pool volumes are then deleted. If a primary storage pool volume becomes empty because all files that were stored on that volume have been restored, the empty volume is automatically deleted from the database.

To facilitate restore processing of entire volumes, the DESTROYED volume access mode is used. If a volume has an access mode of DESTROYED, that volume will not be mounted for either read or write access. This mode is used to designate primary volumes for which files are to be restored. The RESTORE VOLUME command automatically changes the access mode of specified volumes to DESTROYED using a volume list provided by the command syntax. The RESTORE STGPOOL command requires the administrator to update volumes to DESTROYED before the RESTORE STGPOOL command is issued.

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:

- It provides a means of designating 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 should be prematurely terminated for any reason, processing could be resumed and only the files that still reside on DESTROYED volumes would need to be restored.

## Expiration Processing

When filespaces are deleted, backup files are versioned off, or archive files pass their archive retention period, these files are expired from the ADSM database. When files expire, any copies of those files made in copy storage pools are also removed from the database. If backup policies are setup 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.

To ensure expired files can be recovered, the REUSEDELAY parameter for copy storage pools (on the DEFINE STGPOOL and UPDATE STGPOOL command) should be set at least as long as your oldest database backup. When backing up primary storage pools, the REUSEDELAY parameter for the primary storage pools should be set to 0 to efficiently reuse primary scratch volumes.

## Migration Processing

Migration of files between primary storage pools does not affect copy storage pool files. Copy storage pool files do not need to be moved 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 backup 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.

**Note:** The migration of files is not allowed for copy storage pools.

See “Server Migration of Files” on page 247 for more information about migration.

### Reclamation and MOVE DATA Command Processing

Reclamation and MOVE DATA command processing of primary storage pool volumes does not affect copy storage pool files.

Reclamation and MOVE DATA command processing of volumes in copy storage pools is similar to that of primary storage pools.

Differences between primary and copy storage pool processing are as follows:

- Most volumes in copy storage pools may be set to an access mode of OFFSITE, making them ineligible to be mounted. During reclamation or MOVE DATA command processing, 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.
- A MOVE DATA command on a primary storage volume can be directed to any primary storage pool. A MOVE DATA command on a copy storage volume can not be directed to another storage pool. It can only go to the same copy storage pool.

  See “Moving Files from One Volume to Another Volume” on page 300 for more information about moving data.

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. One special consideration occurs when doing it automatically for copy storage pools.

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.

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. This would cause a new volume to be created with the same files on it. The volume you take offsite would be empty according to the ADSM database. If you don’t recognize what is happening, you could perpetuate this process by marking the new partially filled volume OFFSITE.

This consideration with automatic reclamation could be resolved by keeping 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 initiate reclamation at desired times by resetting the reclaim threshold with the UPDATE STGPOOL command. 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.

When you do perform offsite reclamation, it is recommended that you turn on reclamation for copy storage pools during your storage pool backup window and before marking copy storage pool volumes as OFFSITE. Next, turn off reclamation and then mark any newly created volumes as OFFSITE.

It is also recommended that you set the REUSEDELAY parameter for your copy storage pool to be at least as long as the oldest database backup you intend to keep. This will ensure that reclaimed volumes are retained long enough to guarantee the recovery of expired files. Offsite volumes that you see are in the PENDING state are empty but are awaiting release based on the REUSEDELAY value.

See “Space Reclamation for Sequential Access Storage Pools” on page 257 for more information about reclamation.

## 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 a 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 253 for more information about collocation.

## Assigning Volumes in Storage Pools

Volumes are assigned differently depending on whether they are stored in random access storage pools or sequential access storage pools.

### Assigning Random Access Storage Pool Volumes

Volumes in random access storage pools must be prepared for use (formatted) and then defined. See Chapter 12, "Managing Storage Pool Volumes" on page 285 for information about formatting and defining volumes.

### Assigning Sequential Access Storage Pool Volumes

You can predefine volumes in a sequential access storage pool or you can specify that ADSM dynamically acquire scratch volumes. You can also use a combination of predefined and scratch volumes.

Use predefined volumes when you want to control 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 12, "Managing Storage Pool Volumes" on page 285 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, and 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.

One of the most important considerations is the number of copy storage pools you need. This will depend on the hierarchies you have set up with your primary storage pools and what type of disaster recovery protection you wish to implement. For most cases, a single copy storage pool could be used for backup of all primary storage pools.

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 “Full Storage Pool Backups” on page 270 for more information.)

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.

**Note:** 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.

## Storage Considerations

A useful guideline for how much primary disk storage should be dedicated for the staging of client data is: enough storage to handle one night's worth of the clients' incremental backups. This is not always feasible but it is a good guideline. This guideline has even more value when considering storage pool backups.

If one day's worth of client incrementals can be kept in a disk storage pool, backup copies of these files can be made while they are residing on disk, before they are migrated to sequential media. This saves many mount requests while performing your storage pool backups. Backing up your storage pools before migration processing will allow you to copy as many files as possible while they are still on disk. The primary sequential storage pools must still be backed up to catch any files that might have been missed while on disk (for example, large files that went directly to sequential media).

See "Estimating Space Needs for Storage Pools" on page 260 for more information about storage pool space.

## How ADSM Stores Files in a Storage Pool Hierarchy

This section explains how the server selects and accesses a primary storage pool. This information can help you estimate the amount of space required for each storage pool in the hierarchy.

When a user back ups, 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. An example of how this might work follows:

Assume a company has a storage pool hierarchy as shown in Figure 46.

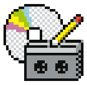


*Figure 46. Storage Hierarchy, Read/Write Access, and Maximum File Size*

The storage pool hierarchy consists of four storage pools:

**FASTDISKPOOL** The top of the storage hierarchy. It contains fast disk volumes for storing data.

**SLOWDISKPOOL** The next storage pool in the hierarchy. It contains slower disk volumes.

**FASTTAPEPOOL** The next storage pool in the hierarchy. It contains tape volumes accessed by high-performance tape drives.

**SLOWTAPEPOOL** The last storage pool in the hierarchy. It contains tape volumes accessed by lower-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 SLOWDISKPOOL, see Figure 46 on page 246.

When the user archives the file, the server determines where to store the file based on the following decision making process:

1. The server selects SLOWDISKPOOL because it is the specified archive storage destination.
2. Because the access mode for SLOWDISKPOOL is read/write, the server checks the maximum file size allowed in the storage pool.
3. The maximum file size allowed in SLOWDISKPOOL is 3MB. FileX is a 5MB file and therefore cannot be stored in SLOWDISKPOOL. The server searches for the next storage pool in the storage hierarchy.
4. The server checks the access mode of FASTTAPEPOOL, which is the next storage pool in the storage hierarchy.
5. The access mode for FASTTAPEPOOL is read-only. Therefore, the file cannot be stored in FASTTAPEPOOL, and the server searches for the next storage pool in the storage hierarchy.
6. The next storage pool in the storage hierarchy is SLOWTAPEPOOL, and the server checks the access mode.
7. The access mode for SLOWTAPEPOOL is read/write. The server then checks the maximum file size allowed in the storage pool.
8. Because SLOWTAPEPOOL is the last storage pool in the storage hierarchy, no maximum file size is specified. Therefore, if there is available space in SLOWTAPEPOOL, FileX can be stored in it.

## Server Migration of Files

ADSM provides an automatic migration facility to maintain free space in a primary storage pool. For example, stored data on a random access disk storage pool can be set to migrate to a less expensive sequential access storage pool. Threshold parameters must be defined before migration can take place.

## 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.

You can use the DEFINE STGPOOL command defaults for the migration threshold, 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 262 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 252 for information about setting the CACHE parameter.

### When Files Are Migrated

Migration from disk storage pools is performed by node, and nodes with the largest file space are migrated first. When the high-migration threshold is reached in a disk storage pool, the server searches for the client file space that consumes the most space in the storage pool. Then the server migrates *all* files from *every* file space belonging to that client. After doing so, the server checks the low-migration threshold for the storage pool to determine if the migration process should be stopped. If, after migrating the given client's files, the amount of used space in the storage pool drops below the low migration threshold, migration ends. If not, another client's file spaces are chosen by using the same criteria as described above, and the migration process continues.

For example, Table 25 displays information contained in the database that is used by the server to determine which files to migrate.

*Table 25. Database Information on Files Stored in DISKPOOL*

| **Client Node** | **Data Storage Used by File Space** |
|---|---|
| TOMC | TOMC/C = 200MB TOMC/D = 100MB |
| HTANG | HTANG = 50MB |
| PEASE | PEASE/home = 150MB PEASE/temp = 175MB |

In addition, Figure 47 on page 249 displays the migration thresholds defined for the disk storage pool *DISKPOOL* and the tape storage pool *TAPEPOOL*.

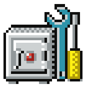


*Figure 47. Setting High- and Low-Migration Thresholds*

When the amount of 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.
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 TAPEPOOL storage pool.

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. For example, by moving data from file spaces TOMC/C and TOMC/D, the space utilization of DISKPOOL could drop below 20%.

If the cache option is enabled, the space utilization of DISKPOOL remains consistent because the server leaves cached copies of migrated files on the storage pool. Cached copies remain on disk storage until space is needed for new files. By using cache, you can improve the retrievability of files. When this option is used, the occupancy value which the server compares against the migration thresholds does not include space occupied by cached copies of files.

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 begins migrating files belonging to the remaining client node with the file space that is using the most space in the storage pool. In this example, the server migrates *all* files that belong to the client node named PEASE to the TAPEPOOL storage pool.

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

### Immediate User Access to Files on Disk Storage

Caching is a good method of providing immediate access to files on disk storage. However, if you need to prevent files on disk storage from migrating to other storage pools, use any of the following methods:

- Do not define the *next* storage pool
- Set the high-migration threshold to 100%

  By setting the high migration threshold to 100%, you can still define the *next* storage pool in the storage hierarchy, so that large files will be moved to a storage pool that supports large files as defined by the maximum file size.

## Migration Thresholds for Sequential Access Storage Pools

Migration from sequential storage pools is performed by volume. This is done to minimize the number of mounts for source volumes. Sequential volumes selected for migration are those that were least recently referenced.

There is no straightforward way to selectively transfer 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.

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.
- 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 262.
- Collocation is set similarly 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 253.
- 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.

  For information about setting a reclamation threshold for tape storage pools, see "Space Reclamation for Sequential Access Storage Pools" on page 257.

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%.

## Damaged Files

If a user experiences a data-integrity error during an attempt to restore, retrieve, or inspect a file during an AUDIT VOLUME operation, the file is marked as damaged. The file is marked as damaged in the database whether the file is in a primary or copy storage pool. If the same file is stored in other storage pools, the status of those file copies is not changed.

If a user attempts to restore, retrieve, or recall a file that is marked as damaged, ADSM will send the user a copy of the file if a copy is available on an onsite copy storage pool volume.

Files that are marked as damaged are not:

- 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 also damaged)

The AUDIT VOLUME command can be used to reset the damaged status of files if the volume in which the file is stored is audited without detecting data-integrity errors. This allows the administrator to correct situations when files are marked damaged as a result of a temporary hardware problem such as a dirty tape head.

If a primary file copy is marked as damaged and a usable backup copy exists in a copy storage pool, the primary file can be recreated using the RESTORE VOLUME or RESTORE STGPOOL command.

ADSM provides commands that allow administrators 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. This command can be used with the preview option to identify primary volumes with damaged files without actually performing the restore operation.
- The QUERY CONTENT command provides a DAMAGED option that allows the administrator to display damaged files on a specific volume.

## 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. However, this option can be changed when a storage pool is defined or updated.

### Considerations for Using 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, the server migrates backed up or archived files, but leaves cached copies 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, that have the oldest retrieval date and occupy the largest amount of disk space. 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.

### Considerations for Not Using Cache

Set cache to *no* if you have limited database space 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 backed up or archived files to the next storage pool and erases the file from the disk storage pool.

If you disable cache, you may want to set higher migration levels for the disk storage pool to prevent 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 file space 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 48 shows an example of collocation enabled, with three clients having separate volumes assigned for each client.

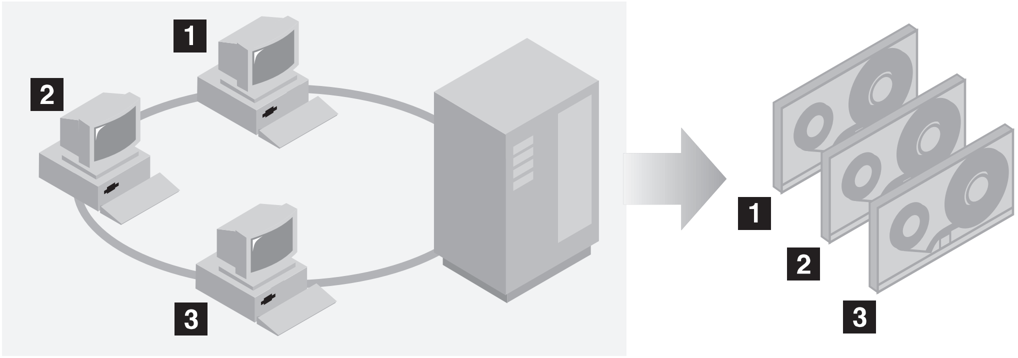


*Figure 48. Example of Collocation Enabled*

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 49 on page 254 shows an example of collocation disabled, with three clients sharing space on a volume.

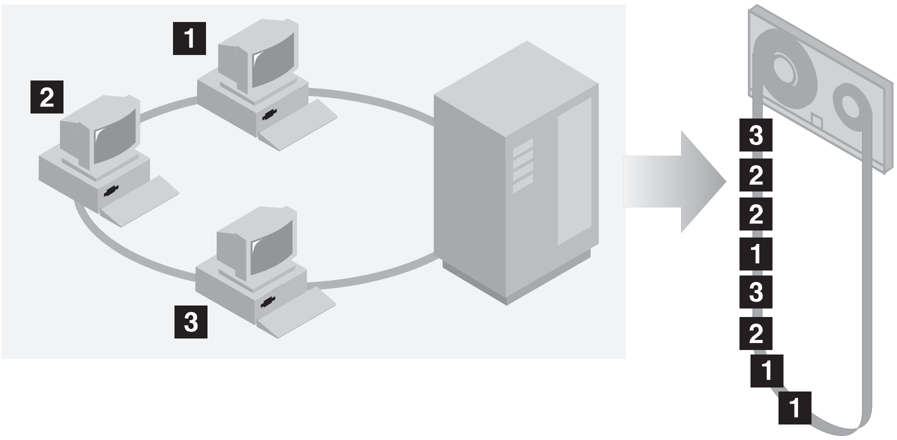


*Figure 49. Example of Collocation Disabled*

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 or not 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 heavily 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 client-migrated 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 user files are backed up, archived, or migrated 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 50 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.

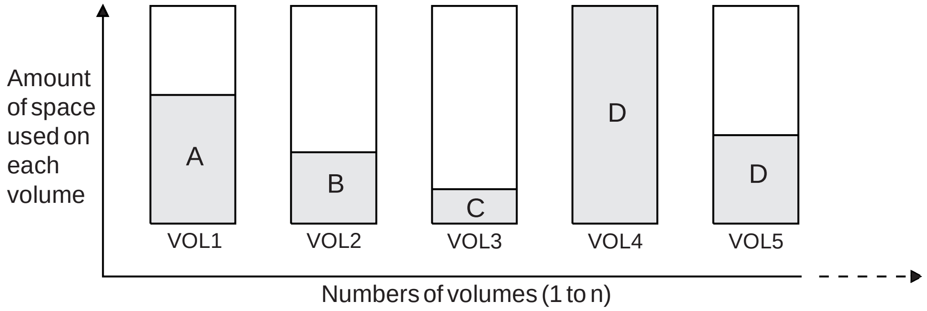


*Figure 50. Using All Available Sequential Access Storage Volumes with Collocation Enabled*

## 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 exist, the server attempts to select any remaining available volume in the storage pool.

Figure 51 on page 257 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.

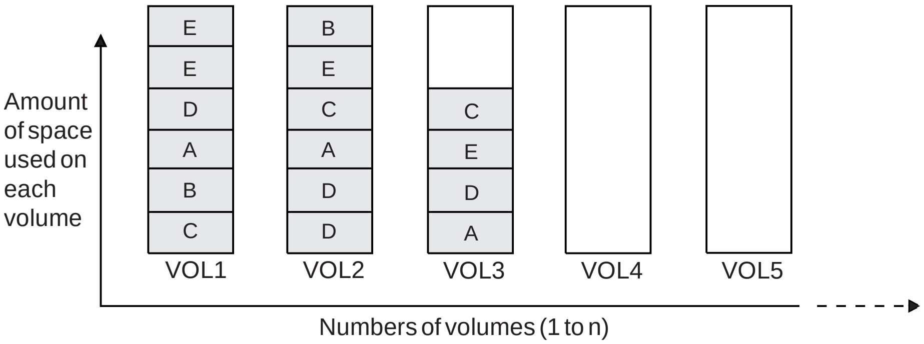

E
E
D
A
B
C
B
E
C
A
D
D
C
E
D
A
Amount of space used on each volume
VOL1
VOL2
VOL3
VOL4
VOL5
Numbers of volumes (1 to n)

*Figure 51. Using All Available Space on Sequential Volumes with Collocation Disabled*

## Space Reclamation for Sequential Access Storage Pools

Space on a sequential volume becomes reclaimable as files become obsolete 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 52 on page 258 shows the active files from tapes 1, 2, and 3, being consolidated on tape 4.

Storage pools that are assigned to device classes with a device type of WORM will have their reclamation value set to 100. This prevents reclamation of WORM optical media, since you cannot rewrite on this type of media. ADSM does not prevent the reclamation value from being set to something lower during definition of the storage pool or from being updated after the storage pool has been defined. This allows administrators to free library space by consolidating data on almost empty volumes to other volumes, and then ejecting the empty WORM volumes.

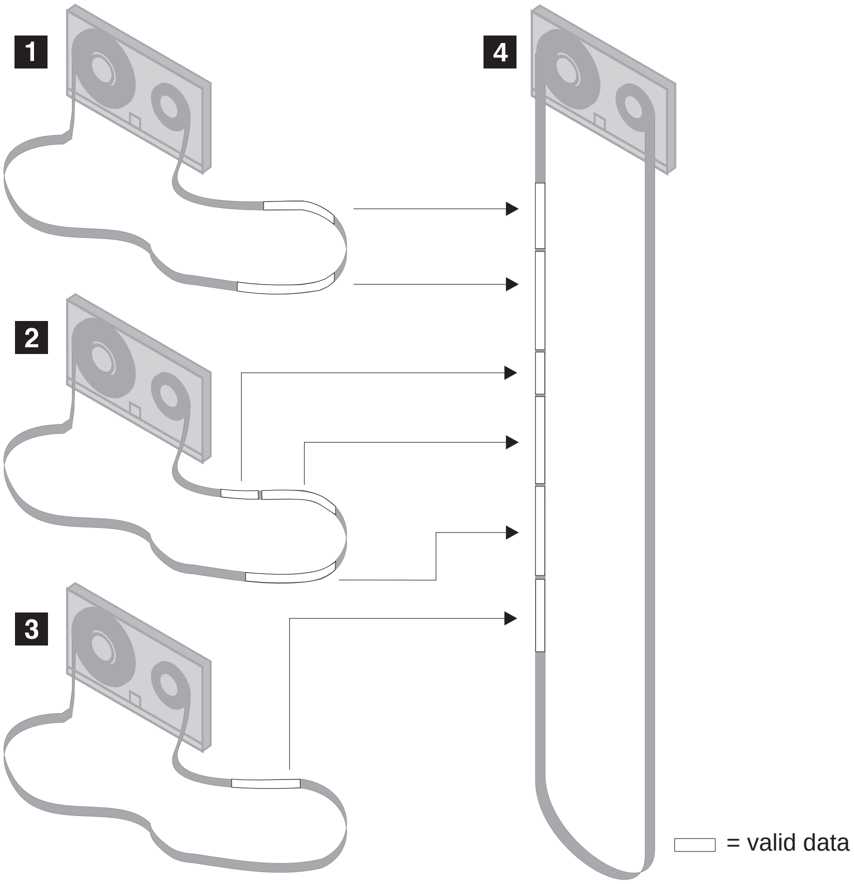


*Figure 52. Tape Reclamation*

After all readable files have been moved to other volumes, one of the following actions is taken for the candidate volume:

- Reused, if the volume has been defined to the storage pool
- Deleted, if the volume has been acquired as a scratch volume

Volumes in a copy storage pool are reclaimed in the same manner as a primary storage pool with the exception of *offsite* volumes.

## 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.

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 default reclamation threshold for copy storage pools is 100%, which means that reclamation is not performed.

Reclamation processing for offsite volumes is performed 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 another onsite volume of a copy storage pool, if necessary).
3. The active files are written to one or more new volumes in the copy storage pool and the database is updated.
4. A message is issued indicating that the offsite volume was reclaimed.

| If you have the Disaster Recovery Manager feature, see “Moving Reclaimed or Expired
| Volumes Back Onsite” on page 366.

## Reclamation Threshold

If the reclamation threshold is low, the server tries to reclaim space occupied by obsolete files more frequently. Frequent reclamation optimizes the use of a sequential access storage pool’s space.

However, each reclamation process requires *at least* two volume mounts. This process could significantly increase the manual intervention needed to mount sequential volumes.

If the reclamation threshold is 50% or lower, ADSM may not be able to combine the usable files from multiple volumes onto a single new volume. At least two drives in the same device class are required for reclamation. For more information about reclamation, see “Mount Limit” on page 221. There must be a sufficient number of volumes, drives (if appropriate), and mount operators (if appropriate) to handle frequent reclamation requests.

If the reclamation threshold is high, reclamation occurs less frequently. A high reclamation threshold is useful if manual volume mounts are required and the operations staff is at a minimum. Setting the reclamation threshold to 100% prevents reclamation from occurring at all.

## 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:

- Observe 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:

- The volume that contains the most data
- Any partially full volume
- An empty predefined volume
- An empty scratch volume

### Reclamation in a Single Drive 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.

## 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

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 provide appropriate levels of service to users.

### 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 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 249 for recommendations on setting migration thresholds.

## Estimating Space for Backed Up Files in a Random Access Storage Pool

To compute the total amount of space needed for all backed up files stored in a single random access (disk) storage pool, such as BACKUPPOOL, use the following formula:

```
backup space = avgwkstsize * utilization * versionexpansion * numwkst
```

**Backup Space** The total amount of storage pool disk space needed.

**AvgWkstSize** The average data storage capacity of a workstation, in MB. For example, if the typical workstation at your installation has two 70MB hard drives, then the average workstation storage capacity is 140MB.

**Utilization** An estimate of the fraction of each workstation disk space used, in the range 0 to 1. For example, if you expect that workstations are 75% full, then use 0.75.

**VersionExpansion** An expansion factor (greater than 1) that takes into account the additional backup versions, as defined in the copy group. A rough estimate allows 5% additional files for each backup copy. For example, for a version limit of 2, use 1.05, and for a version limit of 3, use 1.10.

**NumWkst** The estimated total number of workstations ADSM supports.

If 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 then 50% of the amount of space you have defined for backed up files.

Because additional storage space can be added at any time, you can start with a modest amount of storage space and increase the space by adding storage volumes to the archive storage pool, as required.

## Estimating Space Needs in Sequential Access Storage Pools

To estimate the amount of space required for sequential access storage pools, consider:

- The amount of data being migrated from disk storage pools
- The length of time backed up files are retained, as defined in backup copy groups
- The length of time archived files are retained, as defined in archive copy groups
- How frequently you reclaim unused space on sequential volumes

  See “Space Reclamation for Sequential Access Storage Pools” on page 257 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.

  See “Collocation on Sequential Access Storage Pools” on page 253 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 |
|---|---|
| Defining 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 the following information:

**Note:** Unless otherwise indicated, the following parameters apply to both random access and sequential access storage pools.

**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 primary storage pool (this is the default). Updating a storage pool cannot change whether it is a primary or a copy storage pool.

**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.

**Maximum number of scratch volumes for sequential access storage pools**

By providing a nonzero value, you specify that ADSM dynamically acquires scratch volumes.

**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 data storage.

**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).

**Migration process**

For random access storage pools only, specifies the number of processes that are used for migrating files from this storage pool.

**Cache on random access storage pools**

Enables or disables cache. When cache is enabled, copies of migrated files are left on disk after the migration. In this way, a retrieval request can be satisfied quickly.

**Collocation for sequential access storage pools**

*Collocation* 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.

**Reclamation threshold for sequential access storage pools**

Specifies what percentage of reclaimable space can accumulate on a volume before the server initiates a space reclamation process for the volume.

**Name of the next storage pool**

Specifies the name of the next storage pool where files can be moved.

**Reuse delay period for sequential access storage pools**

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 Storage Pool Hierarchy

The following steps are an example of defining a storage hierarchy.

***Defining a Disk Storage Pool:*** In this example, define a disk storage pool as the destination for files backed up from the engineering department.

1. To help policy administrators identify this storage pool as a storage destination in a backup copy group used by the engineering department, name the storage pool ENGBACK1 and provide it with a meaningful description.
2. Next set a maximum file size of 5MB so that larger files are moved directly to the next storage pool in the hierarchy.
3. Then set a high migration threshold of 85% and a low migration threshold of 40% to provide sufficient free space for normal daily processing. See “Appropriate Migration Threshold Values” on page 249 for guidelines on determining migration thresholds.
4. Finally, use the default settings of read/write for the access mode and *yes* for cache.
5. To define the storage pool named ENGBACK1, enter:

```
define stgpool engback1 disk -
description='disk storage pool for engineering backups' -
maxsize=5M highmig=85 lowmig=40
```

***Defining a Tape Storage Pool:*** Assume that you have already defined a device class named TAPE that describes your tape device environment.

1. The next step is to define a sequential access storage pool named ENGBACK2 and assign it to the device class named TAPE.
2. Use ENGBACK2 as the subordinate storage pool to the disk storage pool ENGBACK1. To ensure that ENGBACK2 is able to accept any size file, use the default of *no limit* for maximum file size.
3. To prevent migration from occurring from this storage pool, set the high migration threshold to 100%.

4. However, to allow the server to move large files to another storage pool if there is insufficient space on ENGBACK2, identify BACKTAPE as the next storage pool in the storage hierarchy.
5. Then to consolidate user files on separate tape volumes, set collocation to *yes* and specify that the server can request up to 100 scratch tape volumes for this storage pool.
6. Finally, use the default settings of read/write for the access mode and 60% for the reclamation threshold.
7. To define the storage pool named ENGBACK2, enter:

```
define stgpool engback2 tape -
description='tape storage pool for engineering backups' -
maxsize=nolimit nextstgpool=backtape highmig=100 -
collocate=yes maxscratch=100
```

***Updating a Disk Storage Pool:*** Finally, specify that ENGBACK2 is the next storage pool defined in the storage hierarchy for ENGBACK1. To update ENGBACK1, enter:

```
update stgpool engback1 nextstgpool=engback2
```

### Example: Defining a Storage Pool Hierarchy in Reverse Order

If you do not want to update each storage pool to establish the storage hierarchy, then begin by defining storage pools at the bottom of the storage hierarchy before you define the storage pools at the top of the storage hierarchy. For example, you could complete the previous three steps in two steps by entering the commands in the following order:

1. Define the bottom of the storage hierarchy first by entering:

```
define stgpool engback2 tape -
description='tape storage pool for engineering backups' -
maxsize=nolimit nextstgpool=backtape highmig=100 -
collocate=yes maxscratch=100
```

2. Define the storage pool at the top of the hierarchy by entering:

```
define stgpool engback1 disk nextstgpool=engback2 -
description='disk storage pool for engineering backups' -
maxsize=360M highmig=85 lowmig=40
```

## Defining a Copy Storage Pool

When you define a copy storage pool, you should be prepared to provide 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 it is primary or copy storage pool.

**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). 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. 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 there is a need to have copies of the files stored in BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL (default disk storage) 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
```

## 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.

The AUDIT VOLUME command marks files that are inaccessible as damaged. Not all access errors cause a file to be marked as damaged during a retrieve, restore, or recall process.

When you use the AUDIT VOLUME command on tapes with errors, the primary files found as damaged are marked as damaged before the user requests access to the files. This process ensures quick user access to the file, because the request is immediately satisfied from the copy storage pool if an undamaged copy exists in a copy storage pool.

The AUDIT VOLUME command also verifies that files previously marked as damaged are still damaged. If a file previously marked as damaged is now valid, the user can obtain the data from the primary storage pool.

For example, a company has a storage hierarchy consisting of one primary random access storage pool. The files stored in the primary random access storage pool are migrated to a primary cartridge tape storage pool (device class is CARTRIDGE) called CART-POOL. Because the files are important to the function of the company, the company backs up every file in the CART-POOL.

**Note:** The company could backup the random access storage pool as well, however they have chosen to only back up the storage pool being used for migration.

The administrator decides to schedule daily incremental backups of the files in the primary storage pool. 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
```

   **Note:** All of the storage volumes in the copy storage pool CART-BACKUP are located onsite.

2. Define the same number of volumes in the copy storage pool (CART-BACKUP) as were already defined in the primary storage pool (CART-POOL) by issuing the following command:

```
define volume cart-backup <volname>
```

3. Define a schedule for backing up the primary storage pool to the copy storage pool by issuing the following command:

```
define schedule backup_cart-pool type=administrative
cmd=''backup stgpool cart-pool cart-backup''
active=yes starttime=20:00 period=1
```

   For more information about scheduling, see Chapter 7, “Scheduling Operations” on page 159.

4. Copy the files that existed in the primary storage pool (CART-POOL) prior to creating the copy storage pool (CART-BACKUP), to the copy storage pool (CART-BACKUP) by issuing the following command:

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

## Recreating Damaged Files

This section explains how to recreate damaged files based on the scenario established in the previous example.

If a user tries to access a file that is stored in CART-POOL and ADSM detects a data integrity problem with the file, the copy of the file in CART-POOL is automatically marked damaged. Future accesses to the file will automatically use the copy in CART-BACKUP as long as the copy in CART-POOL is marked as damaged. To recreate any *damaged* files in CART-POOL, the administrator defines a schedule that executes the following command every month:

```
restore stgpool cart-pool
```

To check for and replace any files that develop data-integrity problems in CART-POOL or in CART-BACKUP, the administrator defines schedules that issue the following commands every three months:

1. For every volume in CART-POOL the following command is executed:

   ```
   audit volume <volname> fix=yes
   ```

   If the AUDIT VOLUME command detects files with data-integrity errors, they are marked *damaged* and an error message is produced.

2. For every volume in CART-BACKUP the following command is executed:

   ```
   audit volume <volname> fix=yes
   ```

   If the AUDIT VOLUME command detects files with data-integrity errors, the copy in CART-BACKUP is deleted and a message produced.

3. Recreate *damaged* primary files by issuing the following command:

   ```
   restore stgpool cart-pool
   ```

4. Produce new copies in CART-BACKUP by issuing the following command:

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

## Backing Up Storage Pools

Administrators can back up primary storage pools into copy storage pools.

| 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 7, “Scheduling Operations” on page 159 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, consideration should be given to the number of mount points and drives that can be dedicated to this operation.

See “Using Storage Pool Backup Features” on page 350 for more information about using storage pool back up in your disaster recovery strategy.

## Full Storage Pool Backups

Incremental backups of storage pools, along with database backups and periodic reclamation of offsite volumes, should provide protection to meet most recovery requirements. However, you may still wish to perform occasional full backups of primary storage pools. Reasons for this may include:

- Concern for shelf life of the media being used for backups
- Legal or audit requirements that specify how backups should be performed
- Familiarity with a full backup and applied incremental approach

A full storage pool backup can be achieved at any time by backing up to a new copy storage pool. Further backups to this new copy storage pool will be incremental.

### Example: Full Storage Pool Backup

Suppose you wish to take weekly full backups and daily incrementals and you want to keep 4 weeks worth of backups. This can be accomplished by backing up to a new copy storage pool each week. When a copy storage pool is 4 weeks old, you can delete all the volumes in that copy storage pool (using the DELETE VOLUME command with DISCARDDATA=YES). The copy storage pool and its volumes could then be reused for the new week's backups.

There are some drawbacks to this approach:

- There will be a lot of database activity introduced with deleting the volumes regularly.
- This approach will maintain 5 copies of each file, including the primary copy .
- Information is kept in the database for each copy of a file in a copy storage pool.

  **Note:** This information is not a full database entry such as is used for the primary copy, but it will still take up database space.

## 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:

- The use of space in your disk and sequential access storage pools
- Migration of data from one to storage pool to the next storage pool in the storage hierarchy
- The use of disk space by cached copies of migrated files

### 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 53 on page 272 shows a standard report with all storage pools defined to the system. To monitor the use of storage pool space, review the *Estimated Capacity* and *%Util* columns.

```
Storage      Device       Estimated  %Util  %Migr  High   Low  Next
Pool Name    Class Name    Capacity               Mig%  Mig%  Storage
                               (MB)                            Pool
-----------  ----------  ----------  -----  -----  ----  ----  -----------
ARCHIVEPOOL  DISK               0.0    0.0    0.0    90    70
BACKTAPE     TAPE             180.0   85.0  100.0    90    70
BACKUPPOOL   DISK              80.0   51.6   51.6    50    30  BACKTAPE
COPYPOOL     TAPE             300.0   42.0
ENGBACK1     DISK               0.0    0.0    0.0    90    70
ENGBACK2     DISK               0.0    0.0    0.0    90    70
```

*Figure 53. Information about Storage Pools*

*Estimated capacity* specifies the available space of 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, the estimated capacity 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. Recall that volumes in a sequential access storage pool, unlike those in a disk storage pool, do not contain preallocated space. Rather, data is written to these volumes as necessary until the end of the volume is reached. It is for this reason that 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 use of each storage pool.

For disk storage pools, this value reflects the total number of disk blocks currently allocated to ADSM. Space is allocated for backed up, archived or client-migrated files that are eligible for server migration, cached files which are copies of server-migrated files, and files that reside on any volumes which 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 53 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 Thresholds

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 client-migrated 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 two volumes are 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 53 on page 272 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 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*
```

If caching is on for a disk storage and files are migrated, the %UTIL value does not change since the cached files still occupy space in the disk pool. However, the %MIGR value decreases since this space is no longer migratable because migration has already occurred. See Figure 54 for an example.

```
Storage      Device       Estimated  %Util  %Migr  High   Low  Next
Pool Name    Class Name    Capacity                Mig%  Mig%  Storage
                               (MB)                            Pool
-----------  ----------  ----------  -----  -----  ----  ----  -----------
BACKTAPE     TAPE             180.0   95.2  100.0    90    70
BACKUPPOOL   DISK              80.0   51.6   28.8    50    30  BACKTAPE
```

*Figure 54. Information on Backup Storage Pools*

At this point, a system administrator can:

- Cancel the migration process
  See “Canceling the Migration Process” on page 275 for additional information.
- End the migration process
  See “Ending the Migration Process” on page 275 for additional information.
- Provide additional space
  See “Providing Additional Space for the Migration Process” on page 276 for additional information.

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

```
query process
```

A message similar to Figure 55 on page 275 is displayed:

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

*Figure 55. Information on the Migration Process*

When migration is finished, the server displays the following message:

```
ANR1101I Migration ended for storage pool BACKUPPOOL.
```

## Canceling the Migration Process

Before a system administrator can cancel the migration process, determine the identification number of the background migration process by entering:

```
query process
```

A message similar to Figure 56 is displayed:

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

*Figure 56. Number of the Migration Process*

Then a system administrator can cancel the migration process by entering:

```
cancel process 1
```

## Ending the Migration Process

| Task | Required Privilege Class |
|---|---|
| Update a storage pool | System or unrestricted storage |

You can update the storage pool to cause an immediate end to the migration process. 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. Depending on your environment, you can:

- Set higher migration thresholds for the disk storage pool to delay the server from initiating migration
- Add volumes to the disk storage pool to increase the storage capacity of BACKUPPOOL, thereby decreasing the migration percentage (%Migr)

  **Note:** This would only be done if you received an out of space message.

### Providing Additional Space for the Migration Process

| Task | Required Privilege Class |
|---|---|
| Update a storage pool | System or unrestricted storage |

You can update the storage pool to provide additional storage volumes to complete the migration process. Add volumes to the tape storage pool or increase the maximum number of scratch tapes to increase the storage capacity of BACKTAPE. The server attempts to restart the migration process every 60 seconds for several minutes and then will terminate the migration process.

## 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 57 on page 277 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 57. 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 migrated files 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 data storage | Any administrator |

Any administrator can request information about data storage occupancy. Use the QUERY OCCUPANCY command for reports with information broken out by node or file space. Use the QUERY STGPOOL command for any other information. Use this report to determine the amount of space used by:

- Client node and file space
- Storage pool or device class
- Type (backup, archive, or space managed)

You can also use this report to evaluate the average size of workstation files stored in data storage.

## Amount of Space Used by Client Node

Any administrator can request information:

- About how much data has been backed up or archived to the server by file space
- About the amount of storage space being used by client node and file space

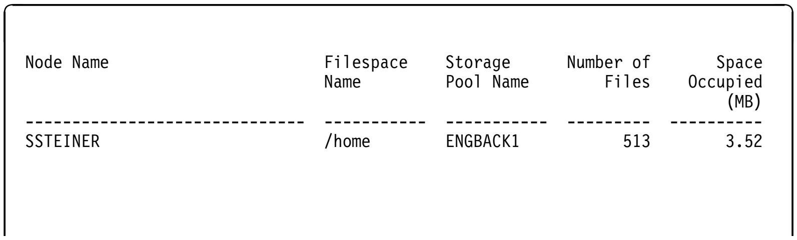
  For information about querying the server for file space information, see “Requesting File Space Information” on page 122.

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 using the exact capitalization used when the file space name was initially defined by the associated ADSM client system. Use the QUERY FILESPACE command to determine the correct capitalization.

Figure 58 shows the number of files backed up or archived from the /home file space belonging to SSTEINER. The report also shows how much space is occupied in each storage pool.



```
Node Name                      Filespace    Storage      Number of       Space
                               Name         Pool Name        Files    Occupied
                                                                          (MB)
------------------------------ -----------  -----------  ---------  ----------
SSTEINER                       /home        ENGBACK1           513        3.52
```

*Figure 58. 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 59 displays a report on the occupancy of tape storage pools assigned to the TAPE8MM device class.

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

*Figure 59. A Report on the Occupancy of Storage Pools by Device Class*

## Amount of Space Used by Backed Up, Archived or Migrated Files

Finally, you can query the server for the amount of space used by backed up and archived files, and files migrated from client nodes. By determining the average size of workstation files stored in data 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 260 and “Estimating Space for Archived Files in a Random Access Storage Pool” on page 261 for information about planning storage space.

To request a report about backup versions stored in the disk storage pool named BACKUPPOOL, for example, enter:

```
query occupancy stgpool=backuppool type=backup
```

Figure 60 on page 280 displays a report on the amount of data storage used for backed up files.

```
Node Name                       Filespace    Storage      Number of       Space
                                Name         Pool Name        Files    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 60. 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 60:

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 304 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 271.

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.

Make sure that the storage pool to be deleted is not specified as the destination for any copy groups in any management classes within the active policy set of any domains. If this pool is a destination and the pool is deleted, backup/archive operations fail because there is no storage space to store the data. Also, a storage pool to be deleted cannot be the destination for space managed files if the destination is specified in any management classes.

See “Defining or Updating Storage Pools” on page 262 for information about updating storage pool definitions.

## Restoring Storage Pools

An administrator can recreate files in a primary storage pool using duplicate copies in a copy storage pool by issuing the RESTORE STGPOOL command.

| Task | Required Privilege Class |
|---|---|
| Restoring storage pools | System, unrestricted storage, or restricted storage |

Use the RESTORE STGPOOL command to restore files from one or more copy storage pools to a primary storage pool. The files must have been copied to the copy storage pools by using the BACKUP STGPOOL command.

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.

  **Note:** Files with data-integrity errors are marked as damaged.

- The primary copy of the file resides on a volume which has an access mode of DESTROYED.

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. Use the QUERY CONTENT command to identify damaged, primary files on a specific volume.

**Note:** Cached copies of files are never restored. Any cached files which have been identified as having data-integrity errors or which reside on a *destroyed* volume will be removed from the database during restore processing.

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 will now locate 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.

If the backup file copies are moved or deleted during restore processing, the restore may be incomplete. Therefore, 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)

In addition, you can delay reclamation processing for your copy storage pools by setting the RECLAIM parameter to 100 (using the UPDATE STGPOOL command) while restore processing is in progress.

**Note:** Copy storage pool volumes that are located offsite must be returned to the onsite location for restore processing.

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 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.

**New storage pool**
Specifies the name of the new primary storage pool to which to restore the files.

**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 “Recovering by Using Backed Up Copies of Storage Pools” on page 351 for an example of using the RESTORE STGPOOL command.

# Chapter 12. Managing 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 |
| Update volumes | System or operator |
| Display information about volumes | Any administrator |

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

| Section | Page |
|---|---|
| **Concepts:** | |
| Storage pool volumes | 285 |
| Access mode for storage pool volumes | 286 |
| **Tasks:** | |
| Allocating space for random access storage pool volumes | 287 |
| Labeling sequential storage pool volumes | 288 |
| Defining storage pool volumes | 290 |
| Updating storage pool volumes | 291 |
| Monitoring the use of storage pool volumes | 292 |
| Managing data storage | 294 |
| Moving files from one volume to another volume | 300 |
| Deleting storage pool volumes | 304 |
| Restoring storage pool volumes | 306 |

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 50 shows whether a task can be performed on the graphical user interface, the command line-interface, or both.

For information about issuing ADSM commands, refer to *ADSM Administrator's Reference*. For help performing a task on the graphical user interface, see the procedure described in "Using Online Help" on page 20.

## Storage Pool Volumes

Volumes in storage pools contain backed up, archived, or migrated data from clients. Storage pools are either random access or sequential access, depending on the type of volume assigned to the pool.

Random access storage pools are always associated with the DISK device class, and all volumes are one of the following:

- Fixed-size files that must be created by using the DSMFMT utility before the ADSM server can access them
- Raw logical volumes that must be defined, typically by using smit, before the server can access them

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:

| | |
|---|---|
| **4MM** | A volume is a 4mm tape cartridge. |
| **DLT** | A volume is a digital linear tape. |
| **8MM** | A volume is an 8mm tape cartridge. |
| **QIC** | A volume is a quarter-inch tape cartridge. |
| **3590** | A volume is a 3590 tape cartridge. |
| **CARTRIDGE** | A volume is a 3480 or 3490 cartridge system tape. |
| **OPTICAL** | A volume is a two-sided 5.25 inch rewritable optical cartridge. |
| **WORM** | A volume is a two-sided 5.25 inch write-once read many optical cartridge. |

## Access Mode for Storage Pool Volumes

Access to any volume in a storage pool is determined by the access mode assigned to that volume:

**Read/write** Allows files to be read from or written to a volume in the storage pool.

If the server cannot write to a read/write access volume, the server automatically changes the access mode to read-only.

**Read-only** Allows files to be read from but not written to a disk or tape volume.

**Unavailable** Specifies that the volume is not available for any type of access by the ADSM server.

**Destroyed** Specifies that a primary storage pool volume has been permanently damaged. Neither users nor system processes (like migration) can access files stored on the volume.

Only volumes in primary storage pools can be updated to destroyed.

If you update a random access storage pool 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.

This access mode is used to indicate an entire volume which should be restored using the RESTORE STGPOOL or RESTORE VOLUME command. After all files on a destroyed volume are restored, the volume is automatically deleted from the database.

If a volume contains no files and the UPDATE VOLUME command is used to change the access mode to destroyed, the volume is deleted from the database.

**Offsite** Specifies that a copy storage pool volume is at an offsite location and therefore cannot be mounted. Use this mode to designate offsite volumes so that:

- You will know which volumes are located offsite
- Mount requests will not be 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.

## Allocating Space for Random Access Storage Pool Volumes

Prepare disk space to be used as a storage volume in a random access storage pool by creating and initializing the volume using the DSMFMT utility. The DSMFMT utility is provided as part of the ADSM server package. For example, create a volume named `stgvol.001` with a size of 20MB for use in a random access storage pool, by entering:

```
dsmfmt -m stgvol.001 20
```

**Note:** This utility program is not an ADSM server command, but it is a program that runs outside of the server. By running the program, you create a standard file. The name of this file is the name used for the storage volume when it is defined to the server.

Another option for preparing a volume is to create a raw logical volume by using smit.

## Labeling Sequential Storage Pool Volumes

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

- 4MM
- 8MM
- DLT
- QIC
- 3590
- CARTRIDGE
- OPTICAL
- WORM

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

### Identifying Drives for Labeling Activities

When running the DSMLABEL utility, you must specify one or more drives for labeling activities. If you specify more than one drive, the program attempts to use them concurrently for maximum performance. In most cases, you are prompted to insert a new volume into a given drive and then to enter the volume name to be written in the label area on the media.

To identify a drive, provide its device name string by using the `-drive=DEVICENAME` argument. If the drive resides in a SCSI library, you must also identify the drive's element address within the library. This value can be obtained from the worksheet that was filled in when the library was configured for use by the server.

### Overwriting Existing Volume Labels

By default, DSMLABEL does not overwrite an existing label on a volume. However, if you want to overwrite existing volume labels, you can invoke DSMLABEL with the `-overwrite` argument.

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

### Labeling Volumes by Using a Library Device

If you want to label volumes by using an automated library, use the `-library=DEVICENAME` argument to identify the library device name. The DSMLABEL labeling utility uses the library to mount volumes. DSMLABEL only allows you to specify one library device. If you have multiple libraries, you must invoke the utility separately for each one. DSMLABEL also assumes that each drive specified with the `-library=DEVICENAME` argument is in the library.

The two modes of operation when using a library are:

**Manual Mode**

Is the default mode of operation when the `-library=DEVICENAME` argument is given. In this mode, the labeling utility assumes that you will be inserting volumes into the library when prompted to do so. The DSMLABEL program then mounts each inserted volume into a drive and writes a label to it using a name that you enter at a prompt.

When operating in manual mode, the labeling utility returns each labeled volume to the entry/exit port of the library or prompts you to remove each labeled volume from a drive if the library is not equipped with an entry/exit port. If you want to alter it so that labeled volumes are stored in storage slots inside the library, you must specify the `-keep` argument when invoking DSMLABEL.

**Search Mode**

Is an optional mode of operation when using a library device. It is selected by specifying the `-search` argument. When operating in this mode, the labeling utility searches all of the storage slots in the library for volumes and tries to label each one that it finds. Upon completion, each volume is returned to its original location in the library, even if the `-keep` argument was not specified.

For SCSI libraries, you are prompted to enter the name of each volume found in the library. It may be convenient for you to redirect input to the labeling utility so that these volume names are read from a file that contains one volume name per line.

For 349X libraries, no prompts are issued for volume names because the name written to the volume's media label always matches the name on the external bar code label.

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

## Volume Labeling Examples

The examples that follow are labeling situations that vary depending on the installation.

### Labeling All of the Volumes in a SCSI Library

Suppose you want to label all of the volumes that reside in a SCSI library. The library device is an Exabyte EXB-120 and, although it contains four drives you only want to use two of them to write labels. Enter the following command:

```
dsmlabel -drive=/dev/mt0,116 -drive=/dev/mt1,117 -library=/dev/lb0 -search
```

### Labeling New Volumes in an Existing Library

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

```
dsmlabel -drive=/dev/mt0,116 -library=/dev/lb0 -overwrite -keep
```

### Labeling Volumes Not in a Library

Suppose you want to label a few new volumes using an IBM 7208-011 tape drive that is not part of an automated library. Enter the following command:

```
dsmlabel -drive=/dev/mt0
```

### Labeling Volumes in an Insert Category

Suppose you want to label all of the volumes that are in the INSERT category in an IBM 3494 tape library and you want to use each of the two drives within the library. Enter the following command:

```
dsmlabel -drive=/dev/rmt1 -drive=/dev/rmt2 -library=/dev/lmcp0
```

## Defining Storage Pool Volumes

The ADSM server can use dynamically acquired scratch volumes, predefined volumes, or a combination in a sequential access storage pool. Volumes in a random access storage pool must be predefined.

When you define a storage pool volume, you inform that server that the volume is available for use when storing backup, archive, or space-managed data. Before a scratch tape volume can be used, it must have a standard label.

To define a volume named VOL1 in the ENGBACK3 storage pool, enter:

```
define volume engback3 vol1
```

**Note on Sequential Access 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; they are also automatically deleted from the storage pool when the server no longer needs them.

## Updating Storage Pool Volumes

Use the UPDATE VOLUME command to update the attributes of an existing random or sequential access storage pool volume. This command can be used to update volumes assigned to primary or copy storage pools.

**Note:** You can reset any error state associated with a volume by updating the volume to an access mode of READWRITE.

A random access volume must be varied offline before you can update the volume to *unavailable* or *destroyed*. To vary a volume offline, use the VARY command.

If a scratch volume with a status of EMPTY and an access mode of OFFSITE is updated so that the access mode is READWRITE, READONLY, or UNAVAILABLE, the volume will be deleted from the database.

When using the UPDATE VOLUME command, be prepared to supply the following information:

**Volume name**
Specifies the name of the storage pool volume to be updated. This parameter is optional.

**Access**
Specifies how users and system processes (like migration) can access files in the storage pool volume.

**Location**
Specifies the location of the volume. This parameter is optional. It can be specified only for volumes in sequential-access storage pools.

**Storage pool**
Specifies the name of the storage pool for volumes to be updated.

**Device class**
Specifies the name of the device class for volumes to be updated. This parameter can be used to restrict the update by device class.

**Access**
Specifies the current access mode of volumes to be updated.

**Status**
Specifies the status of volumes to be updated.

**Preview**

Specifies whether you want to preview the update operation before it is actually performed.

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

You can request that the server produce a report about storage pool volumes. 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

### 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 61 shows the output of this standard query. The example illustrates that data is being stored on the 8mm tape volume named ADSM01, as well as on several other volumes in various storage pools.

```
Volume Name               Storage      Device      Estimated  %Util   Volume
                          Pool Name    Class Name   Capacity           Status
                                                        (MB)
------------------------  -----------  ----------  ---------  -----  --------
/dev/raixvol1             AIXPOOL1     DISK            240.0   26.3  On-Line
/dev/raixvol2             AIXPOOL2     DISK            240.0   36.9  On-Line
/dev/rdosvol1             DOSPOOL1     DISK            240.0   72.2  On-Line
/dev/rdosvol2             DOSPOOL2     DISK            240.0   74.1  On-Line
/dev/ros2vol1             OS2POOL1     DISK            240.0   55.7  On-Line
/dev/ros2vol2             OS2POOL2     DISK            240.0   51.0  On-Line
ADSM00                    TAPEPOOL     TAPE8MM       2,472.0    0.0  Filling
ADSM01                    TAPEPOOL     TAPE8MM       2,472.0    2.2  Filling
```

*Figure 61. Standard Information About Storage Pool Volumes*

## Requesting Detailed Information about Storage Pool Volumes

To query the server for a detailed report on volume ADSM01 in the storage pool named TAPEPOOL, enter:

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

Figure 62 shows the output of this detailed query.

```
                   Volume Name: /adsmfct/adsm01
             Storage Pool Name: TAPEPOOL
             Device Class Name: DISK
       Estimated Capacity (MB): 10.0
                         %Util: 0.0
                 Volume Status: On-Line
                        Access: Read/Write
        Pct. Reclaimable Space:
               Scratch Volume?:
               In Error State?:
      Number of Writable Sides:
       Number of Times Mounted:
             Write Pass Number:
     Approx. Date Last Written:
        Approx. Date Last Read:
           Date Became Pending:
        Number of Write Errors:
         Number of Read Errors:
               Volume Location:
Last Update by (administrator): SERVER_CONSOLE
         Last Update Date/Time: 04/14/1995 17:30:49
```

*Figure 62. 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" 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" for information about auditing a volume.

- Determine the location of an offsite volume.

  The location of a volume in a sequential-access storage pool is shown if the optional LOCATION parameter was used when the volume was defined or during a previous UPDATE VOLUME command.

## Managing Data Storage

| Task | Required Privilege Class |
|---|---|
| Audit or move files from volumes in storage pools over which they have authority | Restricted storage privilege |
| Audit or move files from a volume in any storage pool | System privilege, unrestricted storage privilege |
| Display information about files in storage pool volumes | Any administrator |

Use this section to help you:

- Audit storage pool volumes for data integrity
- Determine what is contained on a storage pool volume
- Move files from one volume to other available volumes

### Auditing a Storage Pool Volume

When users back up, archive, or migrate files, the server retains information about the files in the database, while actually storing backup versions and client-migrated and archive copies in data storage. If there are inconsistencies between the information in the database and the files in a storage pool volume, users cannot restore or retrieve the files from data storage.

For example, you experience a disk failure on a volume in a random access storage pool. The server has already recorded in the database that files were stored on the failed disk. However, because of the disk failure, the backed up files are irretrievable.

To ensure that all recorded files are accessible on volumes in a storage pool, audit any failed volumes by using the AUDIT VOLUME command. An audit checks for any logical inconsistencies between the database and the storage pool volume. You can determine what volumes belong to a storage pool by querying the server for volume information as described in “Requesting General Information about Storage Pool Volumes” on page 292. If files with integrity errors are detected, the handling of these files depends on the following:

- The type of storage pool where the volume belongs
- The FIX option of the AUDIT VOLUME command
- The location of file copies

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. This provides a means for resetting the state of damaged files if it is determined that the errors were caused by correctable hardware problems such as a dirty tape head.

**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 in a file that had previously been marked as damaged, ADSM resets the state of the file so that it can be used. This provides a means for resetting the state of damaged files if it is determined that the errors were caused by correctable hardware problems such as a dirty tape head.

For a volume 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.

## Considerations for Auditing Storage Pool Volumes

When you audit a volume, the server initiates a background process that sends progress messages to the server console and activity log. During the audit process, ADSM checks for any inconsistencies between information about files stored in the database and the actual files stored on a volume in a storage pool. 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

When you audit a volume, a background process is started as the server checks for inconsistencies between the contents of the database and the contents of the volume. During the auditing process, the server:

- Records processing information in the activity log
- Sends informational messages about processing to the server console
- Prevents new files from being written to the volume

You can specify whether you want the server to correct the database in the event that inconsistencies are detected. The system default is to report inconsistencies that are found, but to not correct the errors.

## Auditing a Volume in a Disk Storage Pool

For example, to audit the /dev/raixvol1 disk volume and have only summary messages sent to the activity log and server console, enter:

```
audit volume /dev/raixvol1 quiet=yes
```

The audit volume process is run in the background and the server returns the following message:

```
ANR2313I Audit Volume NOFIX process started for volume
/dev/raixvol1 (process id 4).
```

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

```
query process
```

The following figure displays an example of the audit volume process report.

```
 Process Process Description      Status
  Number
-------- ------------------------ ---------------------------------------------
       4 Audit   (Inspect only)   Storage Pool BACKUPPOOL, Volume
         Volume                    /dev/raixvol1, Files Processed: 680,
                                   Irretrievable Files Found: 0, Partial Files
                                   Skipped: 0
```

To display the results of a volume audit after it has completed, you can issue the QUERY ACTLOG command.

## Auditing Multiple Volumes in a Sequential Access Storage Pool

When you audit a sequential storage volume containing files that span multiple volumes, the server selects all associated volumes and begins the audit process with the first volume on which the first file resides. For example, Figure 63 shows five volumes defined to ENGBACK2. In this example, FileA spans across VOL1 and VOL2, and FileD spans across VOL2, VOL3, VOL4, and VOL5.

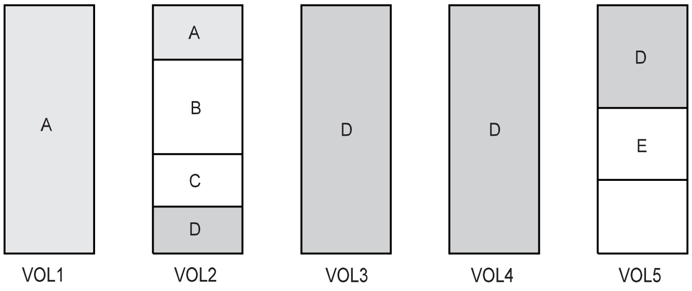


*Figure 63. Auditing Tape Volumes*

If you want to audit volume VOL3, the server first accesses volume VOL2, because FileD begins at VOL2. When volume VOL2 is accessed, the server *only* audits FileD. It does not audit the other files on this volume.

Because FileD spans across 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, this 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 across the single volume to other volumes in the storage pool. This process is useful when a single volume is damaged, yet you want to audit a different volume that has a file which spans onto the damaged volume.

For example, to audit only volume VOL5 and have the server fix any inconsistencies found between the database and the storage volume, enter:

```
audit volume vol5 fix=yes skippartial=yes
```

## 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 by using the exact capitalization used when the file space name was initially defined. 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.

**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 the /usr file space on the TOMC client node, enter:

```
query content adsm01 node=tomc filespace=/usr count=7 type=backup
```

Figure 64 displays a standard report that shows the first seven files from the /usr file space on TOMC stored in ADSM01.

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

*Figure 64. 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 65 on page 300 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 migrated file.

```
             Node Name: PEASE
                  Type: Bkup
        Filespace Name: /home
Client's Name for File: /pease/dir1/code/utl/ test.scr
           Stored Size: 435
        Segment Number: 1/2
          Cached Copy?: No

             Node Name: PEASE
                  Type: Bkup
        Filespace Name: /home
Client's Name for File: /pease/dir1/code/utl/ header.scr
           Stored Size: 514
        Segment Number: 1/1
          Cached Copy?: No

             Node Name: PEASE
                  Type: Bkup
        Filespace Name: /home
Client's Name for File: /pease/dir1/code/utl/ appl.scr
           Stored Size: 1,013
        Segment Number: 1/1
          Cached Copy?: No
```

*Figure 65. Viewing a Detailed Report of the Contents of a Volume*

## 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.

| 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 volumes in any storage pools to which you are authorized | Restricted storage |

**Note:** Files in a copy storage pool do not move when primary files are moved.

## 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 Data Storage Pool Volumes” on page 304 for information about deleting backed up, archived, or client-migrated data before you delete a volume from a storage pool.

- To salvage readable files from a volume that has been damaged

  If the server detects a volume write error, it automatically changes the access mode to read-only.

- 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

A user might want to move all data from a volume in one storage pool to volumes in another storage pool. 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. Ensure that you have sufficient personnel and media to move files from disk to sequential storage.

For example, to move the files stored in the /dev/raixvol1 volume to any available volume in the STGTMP1 storage pool, enter:

```
move data /dev/raixvol1 stgpool=stgtmp1
```

When you move data from a volume, the server starts a background process and sends informational messages, such as:

```
ANR1140I Move Data process started for volume /dev/raixvol1
(process ID 32).
```

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 it cannot read

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.

## Moving Data from an Offsite Volume in a Copy Storage Pool

The MOVE DATA command can be used to move files from an offsite volume in a copy storage pool.

When you use this command to move files from a volume marked as offsite, ADSM:

- Determines which files are still active on the volume to be reclaimed
- Obtains these files from a primary storage pool or from another copy storage pool
- Copies the files to one or more volumes in the destination copy storage pool.

## Preparing to Move Data

Before you move files from one volume to another volume, complete the following steps:

1. Change the access mode of the storage pool to which the volume belongs to read-only. This process prevents users from backing up or archiving files to volumes within the storage pool.

   See “Defining or Updating Storage Pools” on page 262 for information about updating the access mode of a storage pool.

2. Ensure sufficient available space is available on volumes within the specified destination storage pool by:

   - 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 292 for information about requesting information about a storage volume.

   - 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 271 for information about querying a storage pool.

3. If you need more storage space, define volumes or increase the maximum number of scratch volumes in the specified destination storage pool.

   See Chapter 12, “Managing Storage Pool Volumes” on page 285 for preparing volumes to be used for data storage.

4. If you move 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 233 for requesting information about the mount limit value for the device class.

5. Because moving data from tape to tape requires two tape drives, ensure the tape drives are available.

## Requesting Information about the Data Movement Process

To request information on the data movement process, enter:

```
query process
```

The following figure displays an example of the report that you receive about the data movement process.

```
 Process Process Description  Status
  Number
-------- -------------------- -------------------------------------------------
      32 Move Data            Volume /dev/raixvol1, (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
```

## 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 (/dev/raixvol1) source volume in a move operation to available volumes in the BACKTAPE storage pool, enter:

```
query volume * stgpool=backtape
```

Figure 66 on page 304 shows all defined volumes in the BACKTAPE storage pool. This example shows that the storage pool is filling as files are moved from the /dev/raixvol1 volume to available volumes in the BACKTAPE storage pool.

```
Volume Name               Storage      Device      Estimated  %Util   Volume
                          Pool Name    Class Name   Capacity           Status
                                                      (MB)
------------------------  -----------  ----------  ---------  -----  --------
ADSM01                    BACKTAPE     TAPE8MM        2472.0    7.6  Filling
ADSM02                    BACKTAPE     TAPE8MM        2472.0    3.8  Filling
```

*Figure 66. Volume Information Showing Data Movement*

## Deleting Data Storage Pool Volumes

| 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 |

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
  - Filespace deletion
  - Deletion of the primary storage pool volume

In addition, if non-cached files are deleted from a primary storage pool volume, any copies of these files in copy storage pools will also be deleted.

**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.

### Deleting a Storage Pool Volume with Data

To prevent you from accidentally deleting backed up, archived, or client-migrated files from data storage, the server does not allow you to delete a volume without explicitly identifying whether or not you want to discard user data. The only exception to this rule is if the administrator specifies the DISCARDDATA=YES option on the DELETE VOLUME command.

Before you can delete a storage volume, you must do one of the following:

- Move files to another volume
- Explicitly request that the server discard files from the storage volume

See “Moving Files from One Volume to Another Volume” on page 300 for information about moving data from one volume to another volume.

## Deleting an Empty Storage Pool Volume

For example, to delete an empty volume named ADSM03, enter:

```
delete volume adsm03
```

On an administrative client, you will receive the following confirmation messages, unless the client is running with the NOConfirm option:

```
ANR2200W  This command will delete volume ADSM03
from its storage pool after verifying that the volume
contains no data.
Do you wish to proceed? (Y/N)
```

After you respond yes, the server generates a background process to delete the volume.

## Deleting a Storage Pool Volume That Contains Residual Data

Even after you move data, residual data may reside on the volume because of I/O or integrity errors. To delete any volume that contains residual data that cannot be moved, you must explicitly specify that files should be discarded from the volume.

For example, to discard all data from volume ADSM03 before deleting the volume from its storage pool, enter:

```
delete volume adsm03 discarddata=yes
```

When a volume is deleted, the server does not have to access it, because the DELETE VOLUME command only updates the server database.

The server generates a background process and deletes data in a series of batch database transactions. If the delete volume 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.

After all files have been deleted from the volume, the server deletes the volume from the storage pool.

## 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 |
|---|---|
| Define or update volumes in any storage pool for which they have authority | System, unrestricted storage, or restricted storage |

Use the RESTORE VOLUME command to restore files that:

- Are currently stored on one or more volumes in the same primary storage pool
- Were previously copied to one or more copy storage pools by using the BACKUP STGPOOL command

If more than one volume is specified to be restored, this command attempts to minimize volume mounts for the copy storage pool. Therefore, 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.

**Note:** 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.

This command changes the access mode of the specified volumes to DESTROYED.

After files are restored, the old references to these files in the primary storage pool are deleted from the database. This means that 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. If a destroyed volume becomes empty because all files have been restored to other locations, the destroyed volume is automatically deleted from the database.

If the backup file copies are moved or deleted during restore processing, the restore process may be incomplete. Therefore, 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)

In addition, you can delay reclamation processing for your copy storage pools by setting the RECLAIM parameter to 100 (using the UPDATE STGPOOL command) while restore processing is in progress.

This 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 using the RESTORE VOLUME command, be prepared to supply the following information:

**Volume name**
Specifies the name of the volume in the primary storage pool that is being restored.

**Copy storage pool name**
Specifies the name of the copy pool from which the files are to be restored.

**New storage pool**
Specifies the name of the new primary storage pool to which to restore the files.

**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 before it is actually performed.

See “Recovery from Media Loss” on page 354 for an example of using the RESTORE VOLUME command.

# Chapter 13. Exporting and Importing Data

| Task | Required Privilege Class |
|---|---|
| Export: Copy some or all server information to sequential volumes<br>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 | 309 |
| **Tasks:** | |
| Preparing to export or import data | 310 |
| Monitoring export and import processes | 312 |
| Exporting data to sequential media volumes | 315 |
| Importing data from sequential media volumes | 319 |

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 52 shows whether a task can be performed on the graphical user interface, the command-line interface, or both.

For information about issuing ADSM commands, refer to *ADSM Administrator's Reference*. For help performing a task on the graphical user interface, see the procedure described in "Using Online Help" on page 20.

## 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, which consists of file space definitions and authorization rules, and any of the following groups of files from data storage:
  - 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 and active versions of backed up 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 improve client/server performance, 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 balance client/server 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 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 any data being moved. With PREVIEW=NO, the export or import operation is performed.

Issue each EXPORT or IMPORT command with PREVIEW=YES to determine which objects and how much data will be moved. ADSM sends the following types of messages to the server console and 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 and archive copies from data storage 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 request information about the background process, as described in “Requesting Information about an Export or Import Process” on page 312. If necessary, you can cancel an export or import process, as described in “Canceling Server Processes” on page 68.

## 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 server about device classes in order to select a device class supported by both the source and target servers. If the existing device classes are not supported on both the source and target server, then define a new device class, as described in Chapter 10, “Managing Storage Devices” on page 219.

**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 or Optical Disks to Label

To estimate the number of tapes or optical disks 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 or Optical Disks

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 or optical disks have been mounted during the export process.

To ensure that tapes or optical disks are mounted in the correct order, label tapes or optical disks with information that identifies the order in which they are mounted during the import process. For example, label tapes as DSM001, DSM002, DSM003, and so on to indicate the order in which data is stored on the tape volumes.

When you export data, record the date and time for each labeled 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. From the server console or from an administrative client running in console mode, you can display information about:

- A process while it is running
- The activity log for status information when a process has completed

### 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 67.

## Viewing Information from the Server Console

When you issue an IMPORT or EXPORT command, either from the server console or from an administrative client, information is displayed on the server console. Figure 67 shows an example of the information that is displayed after issuing an EXPORT SERVER command.

```
ANR0610I EXPORT SERVER started by SERVER_CONSOLE as process 1.
ANR0639I EXPORT SERVER: Processing domain ENGPOLDOM.
ANR0640I EXPORT SERVER: Processing policy set ACTIVE in policy domain
ENGPOLDOM.
ANR0640I EXPORT SERVER: Processing policy set STANDARD in policy domain
ENGPOLDOM.
ANR0641I EXPORT SERVER: Processing management class STANDARD in domain
ENGPOLDOM, set ACTIVE.
ANR0641I EXPORT SERVER: Processing management class STANDARD in domain
ENGPOLDOM, set STANDARD.
ANR0643I EXPORT SERVER: Processing archive copy group in domain
ENGPOLDOM, set STANDARD, management class ACTIVE.
ANR0643I EXPORT SERVER: Processing archive copy group in domain
ENGPOLDOM, set STANDARD, management class STANDARD.
ANR0643I EXPORT SERVER: Processing backup copy group in domain
ENGPOLDOM, set STANDARD, management class ACTIVE.
ANR0643I EXPORT SERVER: Processing backup copy group in domain
ENGPOLDOM, set STANDARD, management class STANDARD.
ANR0604I EXPORT SERVER: No schedules were found in policy domain * for
exporting.
ANR0635I EXPORT SERVER: Processing node TOMC.
ANR0605I EXPORT SERVER: No schedule associations were found in
policy domain * for exporting.
ANR0637I EXPORT SERVER: Processing file space DRIVED for node TOMC.
ANR0637I EXPORT SERVER: Processing file space OS2 for node TOMC.
ANR0637I EXPORT SERVER: Processing file space OS2VDISK for node TOMC.
ANR0617I EXPORT SERVER: Processing completed successfully.
ANR0620I EXPORT SERVER: Copied 1 domain(s).
ANR0621I EXPORT SERVER: Copied 2 policy set(s).
ANR0622I EXPORT SERVER: Copied 2 management class(es).
ANR0623I EXPORT SERVER: Copied 4 copy group(s).
ANR0626I EXPORT SERVER: Copied 1 node definition(s).
ANR0627I EXPORT SERVER: Copied 3 file space(s), 16 archive file(s)
and 0 backup file(s).
ANR0629I EXPORT SERVER: Copied 3045632 bytes of data.
ANR0611I EXPORT SERVER started by SERVER_CONSOLE as process 1 has ended.
```

*Figure 67. Sample Export Server Output Sent to the Server Console*

## Viewing Information from an Administrative Client

Use the console mode from an administrative client to monitor export or import operations or to capture processing messages to an output file.

To start an administrative session in console mode, 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, like OS/2 or AIX.

If you want ADSM to write all terminal output to a file, specify the OUTFILE option with a destination. For example, to write output to the SAVE.OUT file, enter:

```
dsmadmc -consolemode -outfile=save.out
```

For information about using the CONSOLE mode option and ending an administrative session in console mode, refer to *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 68 on page 315 displays a sample activity log report.

```
Date/Time            Message
-------------------- --------------------------------------------------
05/03/1995 10:50:28  ANR0610I EXPORT SERVER started by SERVER_CONSOLE 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 SERVER_CONSOLE as
process 1 has ended.
```

*Figure 68. 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 or Optical Disks” on page 312 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, restore, or retrieve 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 backed up or archived files. 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 clients 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 65. For more information on locking or unlocking administrators from the server, see “Locking and Unlocking Administrators from the Server” on page 114.

### Preventing Client Nodes from Accessing the Server

If client node information is exported while the same client is performing a backup or archive, 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 65. 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 66.

  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 120.

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 66.
- Unlocking client nodes, as described in “Locking and Unlocking Client Nodes” on page 120

## 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. To ensure that you have sufficient space on tape volumes, allow ADSM to use scratch volumes by using 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 data storage
- Whether the client node ID is locked from server access

Optionally, you can specify whether to export file data. File data consists of file space definitions and authorizations, and any of the following groups of files:

- Backed up files and archive copies
- Active and inactive versions of backed up files
- Active versions of backed up files
- Archive copies
- Archive copies and active versions of backed up files

When client file data is exported, ADSM copies files to export volumes in the order of their physical location in data 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.

In the following example, definitions for client nodes and file spaces in the ENGPOLDOM policy domain will be exported. Also, export any active backup versions of files belonging to these client nodes. Finally, export this information to scratch volumes, which are supported by 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

In the following example, policy and scheduling definitions in the policy domain named ENGPOLDOM will be exported. Identify tape volumes DSM001 and DSM002, which belong to the CARTRIDGE device class. Finally, allow the server to use scratch tape volumes if necessary by using the default of SCRATCH=YES. 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, a system programmer must:

1. Install ADSM from program sequential media on the target server.

   Refer to *ADSM Installing the Server and Administrative Client* for information on installing ADSM.

2. Define disk space for the database and recovery log.

   For information on defining space, refer to *ADSM Installing the Server and Administrative Client*.

3. Define storage pools and volumes.

   Because each server operating system requires different naming conventions for volumes used by storage pools, ADSM does not export data storage definitions. Therefore, you must define initial storage pools and volumes on the target server, as described in the *ADSM Installing the Server and Administrative Client*.

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 data 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 69 on page 321 shows an example of the messages sent to the server console and activity log.

```
ANR0402I Session 3 started for administrator SERVER_CONSOLE (Server).
ANR1363I Import volume DSM001 opened (sequence number 1).
ANR0610I IMPORT SERVER started by SERVER_CONSOLE as process 2.
ANR0612I IMPORT SERVER: Reading EXPORT SERVER data from server ADSM exported
05/07/1995 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 SERVER_CONSOLE as process 2 has ended.
```

*Figure 69. 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 69 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 Data Storage Definitions on the Target Server” on page 324 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 310. For information on reviewing the results of a preview operation, see “Monitoring Export and Import Processes” on page 312.

## 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, decide whether to replace existing definitions. You should also understand how ADSM handles the activation of policy definitions.

After you understand how data is imported to the target server, you can start an administrative client session in console mode to capture import messages to an output file. Then import the server control information from specified tape volumes.

### Determining Whether to Replace Existing Definitions

By using the REPLACEDEFS option, you can specify whether to replace existing definitions on the target server when ADSM encounters an object with the same name during the import process.

For example, if a definition exists for the ENGPOLDOM policy domain on the target server before you import policy definitions, then you must specify REPLACEDEFS=YES to have ADSM replace the existing definition with the data from the export tape.

Definitions that can be replaced include administrator, client node, policy, or schedule definitions. The default is to not replace existing definitions on the target server.

## Understanding How ADSM Imports Active Policy Sets

When ADSM imports policy definitions, the following objects are imported to the target server:

- Policy domain definitions
- Policy set definitions, including the ACTIVE policy set
- Management class definitions
- Backup copy group definitions
- Archive copy group definitions
- Schedule definitions defined for each policy domain
- Client node associations, if the client node definition exists on the target server

If ADSM encounters a policy set named ACTIVE on the tape volume during the import process, it uses a temporary policy set named $$ACTIVE$$ to import the active policy set.

After $$ACTIVE$$ is imported to the target server, ADSM activates this policy set. During the activation process, the server validates the policy set by examining the management class and copy group definitions.

ADSM reports on the following conditions, which result in warning messages during validation:

- The storage destinations specified in the backup copy groups and the archive copy groups do not refer to defined storage pools.
- The default management class does not contain a backup or archive copy group.
- The current ACTIVE policy set contains management class names that are not defined in the policy set to be activated.
- The current ACTIVE policy set contains copy group names that are not defined in the policy set to be activated.

After each $$ACTIVE$$ policy set has been activated, ADSM deletes that $$ACTIVE$$ policy set from the target server. To view information about active policy on the target server, you can use the following commands:

- QUERY COPYGROUP
- QUERY MGMTCLASS
- QUERY POLICYSET

Results from issuing the QUERY DOMAIN command show the activated policy set as $$ACTIVE$$. ADSM uses the $$ACTIVE$$ name to show you that the policy set which is currently activated for this domain is the policy set that was active at the time the export was performed.

## Directing Import Messages to an Output File

The information generated by the validation process can help you define a storage hierarchy that supports the storage destinations currently defined in the import data.

You can direct import messages to an output file to capture any error messages that are detected during the import process by starting an administrative client session in console mode before you invoke this import command.

For example, to direct messages to an output file named IMPSERV.OUT, enter:

```
dsadm -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 or from the server console, enter:

```
import server filedata=none -
devclass=cartridge volumenames=dsm001
```

## Step 3: Tailoring Data Storage Definitions on the Target Server

After you import definition information, use the reports generated by the import process to help you tailor your storage hierarchy on the target server.

To tailor data storage definitions on the target server, complete the following steps:

1. Identify any storage destinations defined in copy groups that do not match defined storage pools by:

   - Reviewing the error messages generated during the validation process, which have been directed to an output file on the administrative client using console mode
   - Querying copy group definitions to compare the storage destinations defined in the copy group with the name of storage pools on the target server

     To request detailed reports for all backup copy groups and archive copy groups, enter:

```
query copygroup * active * standard type=backup format=detailed
query copygroup * active * standard type=archive format=detailed
```

2. Ensure that all copy group definitions in ACTIVE policy sets refer to defined storage pools by doing one of the following:

- Defining storage pools that match the storage destination names for each copy group, as described in “Defining or Updating Storage Pools” on page 262
- Changing the names of the storage pools referenced by the copy groups in the ACTIVE policy set; refer them to storage pools defined on the target server by:

  a. Copying the ACTIVE policy set to another policy set
  b. Modifying the storage destinations of copy groups in that policy set
  c. Activating the new policy set

  For information on copying policy sets, see “Defining and Updating a Policy Set” on page 143.

Depending upon 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, and that secondary storage pools are defined for migration if or when the storage pools specified by copy groups fill up with data.

## 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 consists of all file space definitions and authorization rules and any of the following groups of files:

- Backed up files and archive copies
- Active and inactive versions of backed up files
- Active versions of backed up files
- Archive copies
- Archive copies and active versions of backed up files

Before you import file data information:

- Understand how ADSM handles duplicate file space names
- Decide whether to keep the original creation date for backup versions and archive copies or to import file data using an adjusted date.

Then you can import file data to the target server.

### Understanding How Duplicate File Spaces Are Handled

When ADSM imports file data information, it imports any file spaces belonging to each specified client node. If a file space definition already exists on the target server for the node, ADSM does *not* replace the existing file space name.

If ADSM encounters duplicate file space names when it imports file data information, it creates a new file space name for the imported definition by replacing the final character or characters with a number. A message showing the old and new file space names is written to the server console and to the activity log.

For example, if the C_DRIVE and D_DRIVE file space names reside on the target server for node FRED and on the tape volume for FRED, then the server imports the C_DRIVE file space as C_DRIV1 file space and the D_DRIVE file space as D_DRIV1 file space, both assigned to node FRED.

## Deciding Whether to Use a Relative Date When Importing File Data

When you import file data, you can keep the original creation date for backup versions and archive copies, or you can specify that ADSM use an adjusted date.

Because export-import tape volumes might not be used for some time after an export operation, 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 the export-import tape contains 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 archive and backup copies 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 are imported to the target server

**Backup**

Specifies that only backup copies, whether active or inactive, are imported to the target server

**Backupactive**

Specifies that only active backup copies will be imported to the target server

**Allactive**

Specifies that only archive copies and active versions of backed up files are to be imported

**Spacemanaged**

Specifies that only files that have been migrated from a user's local file system are to be imported

In the following example, you will import all backup and archive copies to the target server. However, do not 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

Although you can issue any import command, ADSM only imports information that can be found on the tape volumes mounted during import. In addition, you can use any import command to copy pertinent information from export-import tapes to the target server, no matter what export command was used to create the export-import tapes.

### Importing a Subset of Information from Tapes

Issue the following operations to import a subset of information to a target server:

- An IMPORT ADMIN command against tapes created with the EXPORT SERVER command
- An IMPORT NODE command against tapes created with the EXPORT SERVER command
- An IMPORT POLICY command against tapes created with the EXPORT SERVER command
- An IMPORT SERVER command against tapes created with the EXPORT NODE command
- An IMPORT SERVER command against tapes created with the EXPORT ADMIN command
- An IMPORT SERVER command against tapes created with the EXPORT POLICY command

### Issuing Useless Import Commands

While ADSM allows you to issue any import command, data cannot be imported to the server if it has not been exported to tape. You cannot issue the following operations to import data:

- The IMPORT POLICY command against tapes created with the EXPORT NODE command
- The IMPORT POLICY command against tapes created with the EXPORT ADMIN command

- The IMPORT ADMIN command against tapes created with the EXPORT POLICY command
- The IMPORT ADMIN command against tapes created with the EXPORT NODE command
- The IMPORT NODE command against tapes created with the EXPORT POLICY command
- The IMPORT NODE command against tapes created with the EXPORT ADMIN command

## Recovering from Errors during the Import Process

During import processing, the server may encounter invalid data due to corruption during storage on tape or in the database prior to the export operation. If invalid data is encountered during an import operation, the server does the following:

- If a new object is being defined, the default value is used
- If the object already exists, the existing parameter is not changed

The server reports on the affected objects to the server console and 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 325.

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.

# Chapter 14. Recovering Data

The ADSM database, recovery log, and storage pools are critical to the operation of the server. If the database or recovery log are unusable, the entire server is unavailable. If a database is lost and cannot be recovered, the backup, archive, and space-managed data for that server is lost. If a storage pool volume is lost and cannot be recovered, the data on the volume is also lost.

To help in the recovery of your data if it is lost, you should use the following strategy:

- Mirror the server database and recovery log
- Back up the server database periodically to removable media and store the media offsite
- Back up the storage pools periodically to removable media and store the media offsite

ADSM also provides standalone database salvage utilities that can be used as a final measure (see “Database Salvage Utilities” on page 355 for details).

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

**MIRRORING**

| Section | Page |
|---|---|
| **Concepts:** | |
| Database and recovery log mirroring | 330 |
| **Tasks:** | |
| Using mirroring | 331 |
| Recovering by using mirrored copies of the database | 333 |

**DATABASE BACKUP AND RECOVERY**

| Section | Page |
|---|---|
| **Concepts:** | |
| Database backup and recovery | 334 |
| **Tasks:** | |
| Using database backup features | 338 |
| Recovering by using backed up copies of the database | 346 |

**STORAGE POOL BACKUP AND RECOVERY**

| Section | Page |
|---|---|
| **Concepts:** | |
| Storage pool backup and recovery | 349 |
| **Tasks:** | |
| Using storage pool backup features | 350 |
| Recovering by using backed up copies of storage pools | 351 |

**BACKUP AND RECOVERY EXAMPLES**

| Section | Page |
|---|---|
| ADSM backup and recovery scenarios | 351 |
| Backup | 351 |
| Recovery from a disaster | 353 |
| Recovery from media loss | 354 |

Most tasks presented in this chapter can be performed using either the graphical user interface or the command line interface. Table 12 on page 54 shows whether a task can be performed on the graphical user interface, the command line interface, or both.

For information about issuing ADSM commands, refer to *ADSM Administrator's Reference*. For help performing a task on the graphical user interface, see the procedure described in "Using Online Help" on page 20.

## Database and Recovery Log Mirroring

One of the most basic but catastrophic failures in ADSM is the loss of the database or recovery log due to a hardware failure. ADSM helps to protect against this by mirroring the database and recovery log, which writes the same data to multiple disks simultaneously. However, mirroring does not protect against a hardware failure that affects multiple drives or causes the loss of the entire system. Administrator control of mirroring (starting, stopping, and so on) can be performed dynamically while ADSM is running. The capacity of the database can also be increased or decreased dynamically using server commands.

Mirroring provides the following advantages:

- Protects against media failures within the database or recovery log by providing up to three exact copies.
- Continues database operations without interruption if a database or recovery log volume fails by using a mirrored copy of the failed volume. The failed volume is placed offline and is automatically synchronized when brought back online.
- Provides a way to avoid costly database recoveries.

- Increases the database read performance by allowing the server to read data from any available volume copy.

Of course, mirroring requires additional DASD for the mirrored volumes.

The following scenario shows the importance of mirroring in the recovery process:

A sudden power outage occurs on the system on which an ADSM server is running. A partial page write occurs, thus corrupting the recovery log. The recovery log is now not completely readable. Without mirroring, transaction recovery operations cannot complete when the server is restarted. However, if the recovery log had been mirrored and a partial write is detected, a mirror volume can be used to construct valid images of the missing pages.

## Using Mirroring

| Task | Required Privilege Class |
|---|---|
| Define database and recovery log volumes | System or unrestricted storage |
| Query mirrored volumes | Any administrator |

This section explains how to:

- Allocate disk volumes to mirror the database and recovery log
- Define mirrored volume copies
- Monitor mirrored volume copies

### Allocating Volume Copies to Separate Physical Disks

By separating volume copies on different physical devices, you protect the server against media failure and increase the availability of the database and recovery log. You should place mirrored copies on separate devices, preferably behind separate controllers (if applicable).

**Note:** Mirrored volumes should have at least the same capacity as the original volumes.

### Defining Database or Recovery Log Mirrored Volumes

To ensure that the entire database or recovery log is mirrored, define a volume copy for each volume in the database or recovery log. For example, if the database consists of five volumes named VOL1, VOL2, VOL3, VOL4, and VOL5, then you must define five volume copies to mirror the database.

Figure 70 on page 332 shows a mirrored database. In this example, VOL3 and VOLC are a group of mirrored volumes with the same portion of the database.

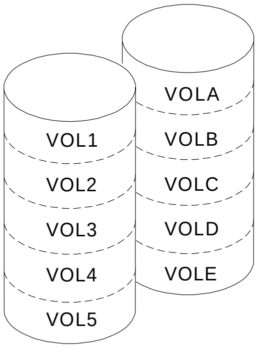


*Figure 70. Mirrored Volumes*

If you define volume copies from the command line and have multiple volumes to mirror, consider using a macro to issue more than one DEFINE command at a time. For example, to define a group of mirrored volumes for the database, use a macro to issue the following commands:

```
define dbcopy VOL1 VOLA
define dbcopy VOL2 VOLB
define dbcopy VOL3 VOLC
define dbcopy VOL4 VOLD
define dbcopy VOL5 VOLE
commit
```

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.

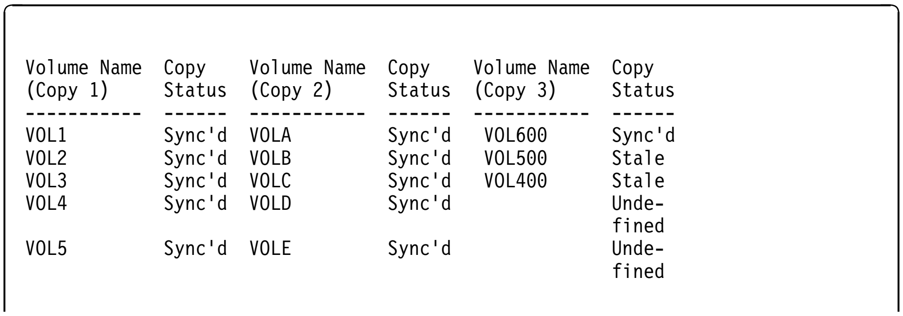
## Requesting Information about Mirrored Volumes

Any administrator can query the server to request information about database or recovery log volumes by using the QUERY DBVOLUME and QUERY LOGVOLUME commands. For example, to query the server for information about mirrored database volumes, enter:

```
query dbvolume
```

Figure 71 on page 333 shows the database volume report you receive after issuing the QUERY DBVOLUME command. In this report, each vertical column displays an entire image of the database or recovery log. For example, VOLA, VOLB, VOLC, VOLD, and VOLE represent one image of the database. On the graphical user interface, each vertical column is called a *volume set*.

Each horizontal column displays a *group of mirrored volumes*. For example, VOL1, VOLA, and VOL600 represent three volume copies that are an exact image of one another.



```
Volume Name  Copy    Volume Name  Copy    Volume Name  Copy
(Copy 1)     Status  (Copy 2)     Status  (Copy 3)     Status
-----------  ------  -----------  ------  -----------  ------
VOL1         Sync'd  VOLA         Sync'd   VOL600      Sync'd
VOL2         Sync'd  VOLB         Sync'd   VOL500      Stale
VOL3         Sync'd  VOLC         Sync'd   VOL400      Stale
VOL4         Sync'd  VOLD         Sync'd               Unde-
                                                       fined
VOL5         Sync'd  VOLE         Sync'd               Unde-
                                                       fined
```

*Figure 71. Information about Database Volumes*

When you query for database or recovery log information, the status of a mirrored volume can be:

**Sync'd** The volume is the only copy or is synchronized with other volumes in the group.

**Stale** The volume is not available for recovery because synchronization has not completed or begun.

**Offline** A volume has been made unavailable to the server by varying the volume offline. See "Varying Disk Volumes Online or Offline" on page 68 for details.

**Undefined** No volume has been defined as a mirrored copy.

### Controlling Database and Recovery Log Performance

You can control database and recovery log processing and performance by using two options in the server options file. The MIRRORREAD option and the MIRRORWRITE option specify modes for reading and writing database and recovery log pages. See *ADSM Installing the Server and Administrative Client* for details.

## Recovering by Using Mirrored Copies of the Database

If a volume fails because of media failure and mirroring has been in effect, you can recover the volume by taking the following steps:

1. View the current status on database and recovery log volumes and volume copies (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 the new database or recovery log volume copy (DSMFMT).
5. Bring the ADSM 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.

## Database Backup and Recovery

You can take full or incremental backups of the database while the server is operational and available to clients. If a disaster occurs, you can use the backed up copies to restore the database. ADSM lets you recover the database to its most current state (roll-forward recovery) or to a specific point in time.

The best database protection is to mirror the database and recovery log and periodically back up the database. To take the fullest advantage of ADSM's database backup and recovery feature, you must create volume history files (see "Establishing Volume History Backup Files" on page 339) and device configuration files (see "Establishing Device Configuration Backup Files" on page 340).

To ensure the fastest recovery time and highest availability of the database, both the database and recovery log should be mirrored. However, if you do not mirror the database and if the recovery log is available, you can still restore the database to its most current state. Therefore, whatever strategy you use, you should always mirror the recovery log. Mirroring only the recovery log requires much less storage space than mirroring the database.

### Roll-Forward Recovery

With the recovery log available, roll-forward recovery lets you restore a database to its most current state. Together, mirroring and roll-forward recovery provide the most comprehensive protection for an ADSM database.

When enabling roll-forward recovery, consider the following:

- The recovery log must be large enough to store additional recovery log records.

  Operating in roll-forward mode significantly increases recovery log storage requirements. The extent of the increase is determined by the number of database transactions since the last database backup. Database transactions are created by server operations including client backups and archives, storage pool migrations, tape reclamations, and expiration processing. In roll-forward mode, the recovery log tracks all transactions since the last database backup. In using point-in-time mode, only current, uncommitted transactions are kept in the recovery log.

- More frequent database backups help control recovery log size.

  When a full or incremental database backup is completed, recovery log records preceding the backup are deleted, freeing up recovery log storage for reuse. Using the database backup trigger automatically starts database backups based on recovery log utilization.

You should monitor recovery log storage, recovery log usage, and the database backup trigger to achieve a balance that best meets your needs.

- Mirroring a larger recovery log also requires additional disk space.
- The frequency of required database backups depends on the size of the recovery log, how the database backup trigger is set, and the volume of ADSM transactions.
- Roll-forward mode is an alternative to database mirroring. It lets you recover to the most recent state following a media failure, without doubling your database storage requirements. However, unlike mirroring, roll-forward recovery does not provide continuous operations on media failure.
- Roll-forward mode is not a disaster recovery utility. When the recovery log is not available, as with a disaster or with a recovery log media failure when not using mirroring, you must recover to a point in time.

## Point-in-Time Recovery

Point-in-time recovery is normally used for exceptional situations such as disaster recovery or to remove the effects of errors that can cause inconsistencies in the database. You can also use point-in-time recovery if you do not want to create a larger recovery log size for ADSM. However, if the server supports HSM clients roll-forward recovery is preferable. Space-managed files should be protected as fully as possible from hardware failure. If you enable roll-forward recovery, point-in-time recovery is also available.

For disaster recovery purposes, you can run full or incremental backups as often as necessary to ensure that an ADSM database can be restored to an acceptable point in time. You can send the backups to an offsite location for safekeeping.

You can also choose to enable point-in-time recovery only. The disadvantages of enabling only point-in-time recovery are:

- You cannot restore the database to its most current state. You can only restore it to a time at which a backup was taken.
- Because restoring a single database volume requires roll-forward recovery, you must restore the entire data base even if only one volume is damaged.
- You should mirror the database. Mirrored copies of a database can consume a significant amount of disk space.

The advantage of using point-in-time, however, is that the recovery log can be defined with a much smaller size.

## Determining Which Recovery Mode to Run

There are several key tradeoffs in deciding whether to run in roll-forward mode or point-in-time mode.

Roll-forward mode provides the ability to recover the database to the most recent state upon a media failure on a database volume. Point-in-time (NORMAL) mode requires significantly less recovery log space.

For most situations, running in point-in-time mode is sufficient if you can afford to mirror the database and recovery log or place them on devices that guarantee availability. If you are mirroring the database, roll-forward mode only has value if all copies of a database volume fail concurrently and all recovery log volumes are still available.

If your database is so large that you cannot afford to mirror it, roll-forward mode provides a very good alternative. It allows you to recover to the time of failure. However, some of the storage you save from not mirroring the database will need to be reinvested in a much larger recovery log.

## Determining Which Type of Backup to Run

A full backup takes longer to run than an incremental backup because it copies the entire database. However, recovery time is faster with a full backup because only one set of volumes needs to be loaded to restore the entire database. A full backup is required under specific conditions, but you can run up to 32 incremental backups between each full backup.

An incremental backup takes less time to run because it copies only those database pages that have changed since the last time the database was backed up. However, incremental backups increase the time it takes to recover a database because a full backup must be loaded first, followed by some or all of the incremental backups in the same database backup series.

## When to Perform Backups

To ensure that you have the most recent database information, you should back up the database if:

- Significant client backup or archive activities have taken place on the server
- Migration has moved data between storage pools
- Reclamation has moved client files on sequential storage volumes
- Move data or delete volume commands have been used to rearrange data or remove volumes from a storage pool
- Storage pool backups have been done

Under typical conditions, you should probably back up your storage pools each day, immediately followed by a database backup. Depending on the amount of client data and frequency of the activities mentioned above, you may back up less often.

**Note:** In roll-forward mode with a database backup trigger defined, backups are automatic.

## Database Backup Trigger

When you are running in roll-forward mode, a database backup trigger causes ADSM to run a backup of the database automatically, based on the percentage of space used in the recovery log. When the amount of space occupied in the recovery log reaches the percentage that you specify, ADSM automatically runs a full or incremental backup of the database and deletes any unnecessary recovery log records.

**Note:** ADSM uses the database backup trigger only when roll-forward mode is in effect.

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 is run.

If the recovery log runs out of space and you cannot start the server for normal operation, you can format an additional recovery log volume and use the dsmserv command with the EXTEND LOG parameter to start the server and extend the recovery log to use the new volume. Refer to the *ADSM Administrator's Reference* for additional information.

## Volume History Backup Files

To perform a point-in-time restore, ADSM needs information about the volumes used for the backups. After a point-in-time restore, you need to know which sequential access storage pool volumes to audit.

The following information, which the ADSM server stores in the database, is needed for a point-in-time restore:

- Volumes used for database backups
- Volumes used to export administrator, node, policy, or server data
- Sequential access storage pool volumes that have been added, reused, or deleted

During a restore of the database, this information cannot be obtained from the database. Therefore, you should have at least one backup copy of the volume history information. You can specify one or more files in which the server backs up the volume history information. Whenever ADSM updates volume history information in the database, it also updates the volume history files. For details about setting up the volume history file, see "Establishing Volume History Backup Files" on page 339.

**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.

## Device Configuration Backup Files

When you define, update, or delete a device class, drive, or library ADSM updates the database. To restore a database, ADSM requires a definition for the device class from which backup data is to be read. When the database is being restored no definitions can be read from the database. Therefore, you should have at least one backup copy of the device configuration information. You can specify one or more files in which the server backs up the device configuration information. Whenever ADSM updates device configuration information in the database, it also updates the information in the backup files. For details about setting up the device configuration file, see "Establishing Device Configuration Backup Files" on page 340.

During a database restore operation, ADSM tries to open the first device configuration file. If it cannot open or read that file successfully, ADSM tries to use any remaining device configuration files (in the order in which they appear in the dsmserv.opt file) until it finds one that is usable. If none can be found, you must recreate the file. See “Recreating a Device Configuration File” on page 341 for details.

## Using Database Backup Features

| Task | Required Privilege Class |
|---|---|
| Set the recovery log mode | System or unrestricted storage |
| Extend or reduce the size of the recovery log | System or unrestricted storage |
| Define, update, or delete the database backup trigger | System or unrestricted storage |
| Query the database backup trigger | Any administrator |
| Back up or delete the volume history file or the device configuration file | System or unrestricted storage |
| Query the volume history file or the device configuration file | Any administrator |
| Backup the database | System or unrestricted storage |

This section discusses how to set up and manage database backup. The following topics are included:

- Defining device classes for backup
- Establishing volume history backup files
- Establishing device configuration backup files
- Performing the database backups
- Setting the recovery log mode
- Setting the database backup trigger
- Adjusting the size of the recovery log

### Defining Device Classes for Backup

You can use existing device classes to use 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 for incremental backups run automatically by ADSM based on the setting specified with the DEFINE DBBACKUPTRIGGER command.

You should specify the same device class for all incremental backups that apply to the same full backup to make restoring those backups as fast as possible.

In addition, to avoid a situation in which a backup needs to be run based on the database backup trigger but no mount point is available, you can define one or more device classes to be used only for automatic backups. This process reserves one or more mount points specifically for this purpose. However, if you share the device class with other operations and because BACKUP DB is a high priority operation, it automatically cancel lower priority operations such as reclamation if all the mount points are in use. This frees a mount point for the database backup.

## Establishing Volume History Backup Files

Because volume history information is vital for a point-in-time recovery on a lost or damaged database, you should ensure that you have at least one backup copy of this information. You should put the volume history file on a disk other than the one on which your database volumes reside. You can also store a remote copy, for example, on an NFS-mounted file system.

You can specify one or more backup copies by defining them with the VOLUMEHISTORY option in the server options file (dsmserv.opt). Maintaining multiple backup copies of the volume history information in different locations can help to ensure that the information is available if it is needed. For disaster recovery purposes, you can also print the volume history information and store it offsite.

The Disaster Recovery Manager feature of ADSM also saves a snapshot of the volume history information file in its disaster recovery plan file.

For example, to define the **volhist.bk1** and **volhist.bk2** volume history backup files, enter the following in the dsmserv.opt file:

```
volumehistory  volhist.bk1
volumehistory  volhist.bk2
```

**Note:** With the VOLUMEHISTORY option, you can specify files in different subdirectories to have copies on different hard disks or network-mounted file systems.

After you have made these entries, halt and restart the server for them to become effective. When you halt the server, ensure that you are not affecting any client sessions. For information on how to stop the server, see "Halting the Server" on page 62.

After you have restarted the server, whenever ADSM updates information about volumes in the database, it also updates the same information in the files you have defined in the VOLUMEHISTORY option.

You can also use the BACKUP VOLHISTORY command at any time to store a backup copy of volume history information in one or more specified files.

For example, to back up the volume history information in the **volhist.bk3** and **volhist.bk4** files, enter:

```
backup volhistory filenames=volhist.bk3,volhist.bk4
```

If you do not specify files in the command, ADSM backs up the volume history information in *all* files specified with the VOLUMEHISTORY option in the dsmserv.opt file.

You should specify volume history files in the server options file or schedule the BACKUP VOLHISTORY command to follow your scheduled database backups. A copy of your most recent volume history file should be stored offsite with your database backups.

You can use the QUERY VOLHISTORY command to display volume history information. To display all volume history information up to yesterday, enter:

```
query volhistory enddate=today-1
```

You can delete volume history information by using the DELETE VOLHISTORY command.

For example, to delete information that is 30 days old or older, enter:

```
delete volhistory todate=today-30
```

**Note:** When volumes not in storage pools are deleted from the volume history with the DELETE VOLHISTORY command, 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 volume history information about volumes in storage pools is deleted, the volumes themselves are not affected.

## Establishing Device Configuration Backup Files

Because device configuration information is vital to restoring a lost or damaged database, you should ensure that you have at least one backup copy of this information. You should specify one or more backup copies by defining them with the DEVCONFIG option in the server options file (dsmserv.opt). When you define, update, or delete a device class, drive, or library, ADSM writes backup copies of the resulting information in the files specified with the DEVCONFIG option.

Maintaining multiple backup copies of device configuration information in different locations can help ensure that the information is available if it is needed. For disaster recovery purposes, you can also print the device configuration information and store it offsite.

The Disaster Recovery Manager feature of ADSM also saves a snapshot of the volume history information file in its disaster recovery plan file.

For example, to define the **devconf.bk1** and **devconf.bk2** device configuration backup files, enter the following in the dsmserv.opt file:

```
devconfig  devconf.bk1
devconfig  devconf.bk2
```

After you have made these entries, halt and restart the server for them to become effective. When you halt the server, ensure that you are not impacting any client sessions. For information on how to stop the server, refer to “Halting the Server” on page 62.

After you have restarted the server, whenever you use the DEFINE, UPDATE, or DELETE commands for DEVCLASS, DRIVE or LIBRARY, ADSM backs up copies of the resulting definitions to the files that you have defined in the DEVCONFIG option.

You can also use the BACKUP DEVCONFIG command at any time to store a backup copy of device configuration information in one or more specified files.

For example, to back up the device configuration information in the **devconf.bk3** and **devconf.bk4** files, enter:

```
backup devconfig filenames=devconf.bk3,devconf.bk4
```

If you do not specify files in the command, ADSM stores copies of the device configuration file in *all* files specified with the DEVCONFIG option in the dsmserv.opt file. A copy of your most recent device configuration file should be stored offsite with your volume history file and database backups.

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 SCSI or 3494 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 operation to locate the slot that the volume is in. If you must recreate the device configuration file, you cannot recreate the volume location information. Therefore, you must define your library as a manual library and manually mount the volumes during the DSMSERV processing.

### Recreating a Device Configuration File

The following commands read and execute the device configuration file:

- DSMSERV RESTORE DB
- DSMSERV LOADDB
- DSMSERV DISPLAY DBBACKUPVOLUME

If no device configuration file is found, you must recreate it before you can start the restore operation. The file consists of DEFINE commands that ADSM runs when you issue one of the previous DSMSERV commands. The device configuration file must follow these conventions:

- The commands must be in this order:
  - DEFINE DEVCLASS
  - DEFINE LIBRARY
  - DEFINE DRIVE

  You need to provide only those definitions needed to mount the volumes read by the DSMSERV command. 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 1550 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=/dev/mt0
```

## Performing the Database Backups

Before using the database backup and recovery feature, you must first do a full back up of your database.

For example, to perform a full backup of your database to the TAPE1 device class, enter:

```
backup db type=full devclass=tape1
```

In this example, ADSM writes the backup data to scratch volumes. You can also specify volumes by name. After you have done 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 TAPE1 device class, enter:

```
backup db type=incremental devclass=tape1
```

## Setting the Database Backup Trigger

The DEFINE DBBACKUPTRIGGER command specifies a percentage of used space in the recovery log. When that percentage is reached and if the log mode is set to ROLLFORWARD, ADSM automatically runs a full or incremental database backup database and deletes any unnecessary recovery log records.

Setting a database backup trigger is optional, but recommended to ensure that the recovery log does not run out of space before the next backup is run.

**Note:** If the log mode is set from NORMAL to ROLLFORWARD, the next database backup must be a FULL backup. The server does not start saving log records for roll-forward recovery until this full backup completes successfully. If a database backup trigger is defined when you set the log mode to ROLLFORWARD, the full backup is done automatically.

To set the database backup trigger at 60 percent for an automatic backup of the database, enter:

```
define dbbackuptrigger logfullpct=60 devclass=TAPE1
```

If you do not specify the LOGFULLPCT parameter in the command, the backup trigger defaults to 50 percent. In addition, ADSM runs 6 incremental backups to every full backup unless you specify a different value in the command.

For example, to run 20 incremental backups to every full backup, enter:

```
define dbbackuptrigger logfullpct=60 devclass=tape1 numincremental=20
```

Each incremental backup, whether run automatically by ADSM or run by issuing the BACKUP DB command, is added to the count of incremental backups run.

Each full backup, whether run automatically by ADSM or run by issuing the BACKUP DB 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 initially set the database backup trigger, you might find that automatic backups are being run too often or not enough. Before taking any action, check the current settings in order to understand how much to change the backup trigger percentage.

To check your current settings, enter:

```
query dbbackuptrigger
```

ADSM displays the following information:

```
                       Backup Trigger Status: Enabled
               Device Class for Full Backups: FULL
        Device Class for Incremental Backups: INCR
                         Log Full Percentage: 60
Number of Incremental Backups per Backup Series: 7
Incremental Backups taken in this Backup Series: 0
                  Next Triggered Backup Type: Incremental
               Database Backup in Progress?: No
                        Database Backup Type:
             Last Update by (administrator): HINES
                       Last Update Date/Time: 03/06/1995 10:49:23
```

For example, assume that the database backup trigger is automatically running backups more often than you want. The above information shows that the Log Full Percentage attribute is set at 60 percent. To increase that attribute to 70 percent, enter:

```
update dbbackuptrigger logfullpct=70
```

If you no longer want to use the database backup trigger, enter:

```
delete dbbackuptrigger
```

When you delete the database backup trigger, ADSM does not run backups of the ADSM database automatically and does not remove unnecessary log records from the recovery log.

After you delete the database backup trigger, be sure to use the SET LOGMODE command to change the log mode to NORMAL. If the database backup trigger automatically runs backups more often than you want and the setting is high (for example, 90%), you should probably increase the recovery log size.

**Note:** If you turn off the trigger and stay in roll-forward mode, transactions fail when the log fills. If you set the log mode to NORMAL, you cannot perform roll-forward recovery.

## Setting the Recovery Log Mode

The recovery log mode determines how long ADSM saves records in the recovery log. You use the SET LOGMODE command to specify whether or not to save enough log records to perform roll-forward recovery.

The default log mode is NORMAL, which does not allow for roll-forward recovery. ADSM saves only those records required to restore the database to the point it was at when the last backup was run.

The advantage to using NORMAL log mode is that it requires less storage space for the recovery log than running with roll-forward log mode requires.

To set the log mode to normal, enter:

```
set logmode normal
```

To make roll-forward recovery available, you must set the logmode to ROLLFORWARD. When you set the logmode to ROLLFORWARD, ADSM saves all recovery log records that reflect changes to the database since the last time a database backup was run. ADSM saves the recovery log records needed to roll the database forward to its most current state after loading the most recent backup series. Each time the database is backed up, ADSM deletes any recovery log records it no longer needs for that purpose.

To set the log mode to ROLLFORWARD, issue the following command:

```
set logmode rollforward
```

## Adjusting the Size of the Recovery Log

The frequency of automatic backups depends not only on the percentage set for the database backup trigger, but also on the size of the recovery log and the volume of ADSM transactions. The considerations for adjusting the size of the recovery log differ depending on the log mode that you select.

If the log mode is set to NORMAL, adjust the size of the recovery log based only on the volume of concurrent ADSM transactions. As more clients are added and the volume of concurrent transactions increases, you can extend the size of the log.

In ROLLFORWARD mode, the size of the recovery log, the percentage set for the database backup trigger, and the volume of ADSM transactions all affect the frequency at which backups are required.

Generally, extending the size of the recovery log decreases the frequency of automatic backups, and reducing the size of the recovery log increases the frequency. However, after the size of the recovery log is established, you can also increase or decrease the frequency of automatic backups by adjusting the percentage specified for the database backup trigger. In addition, fluctuations in the volume of ADSM transactions can also affect the frequency of automatic backups.

To increase the size of the recovery log, use the EXTEND LOG command. For example, to increase the size of the recovery log by 100MB, enter:

```
extend log 100
```

**Note:** You can extend the log only if there is space. The QUERY LOG command displays the maximum extension permitted. If there is no space for extension, you must add volumes to the recovery log before extending it.

## Recovering by Using Backed Up Copies of the Database

This section explains how to recover from a disaster by using backed up copies of the database and recovery log. The following topics are included:

- Restoring a database by using point-in-time recovery
- Restoring a database to its most current state

To perform a restore, you should have the following information, preferably stored offsite:

- Back up volumes of the database
- Copy storage pool volumes
- On the same type of tape volumes as the backups or on diskette, or as printouts:
  - The server options file (dsmserv.opt)
  - Volume history file
  - Device configuration file
  - Output from QUERY DBVOLUME FORMAT=DETAILED and QUERY LOGVOLUME FORMAT=DETAILED (for details of database and recovery log setup)

The Disaster Recovery Manager feature of ADSM assists with many of the recovery steps discussed in this section. For more details about the DRM feature of ADSM, see Chapter 15, “Using Disaster Recovery Manager” on page 359.

### Restoring a Database by Using Point-in-Time Recovery

You need backup copies of volume history and device configuration information for the recovery operation. See “Volume History Backup Files” on page 337 and “Device Configuration Backup Files” on page 337 for details about defining these files.

If the volume history file is not available, you must mount tape volumes in the correct order or specify their order on the DSMSERV RESTORE DB command.

**Note:** After you restore the database, you lose any volume history information pointed to by the options file. You will need this information to identify the volumes to be audited. Therefore, before issuing the DSMSERV RESTORE DB command, rename and save a copy of the volume history file.

If you lose the device configuration file, you can restore it manually (see “Recreating a Device Configuration File” on page 341).

To restore the database to a specific point in time, use the DSMSERV RESTORE DB command. ADSM restores the database as follows:

- Reads the volume history file to locate the last full backup that occurred on or before the specified date and time.
- Using the device configuration file, requests a mount of the first volume, which should contain the beginning of the full backup.
- Restores the backup data from the first volume.
- 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.

For example, to restore the database to a backup series that was created on January 3, 1995, enter:

```
dsmserv restore db todate=01/03/95
```

From the old volume history information, you need a list of all the volumes that were reused, added, and deleted since the original backup. Using this list, perform the following steps:

1. Audit all disk volumes, all reused volumes (STGREUSE), and any deleted volumes that you could locate. 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. If the audit detects any damaged files, issue the RESTORE STGPOOL command to restore those files after you have audited all the volumes in that storage pool.
2. 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 DISCARDDATA=YES.
3. Re-add any volumes that were added since the backup.

Some files may be lost if they were moved since the backup (due to migration, reclamation, or move data requests) and ADSM cannot locate them. You can minimize this loss by using the REUSEDELAY parameter when defining or updating storage pools. This parameter delays volumes from being returned to scratch or being reused.

If you have backed up your storage pools using the BACKUP STGPOOL command, instead of auditing volumes, you can update the reused volumes (STGREUSE) and the deleted volumes as DESTROYED. Then restore the storage pools. By backing up your storage pool data in addition to backing up your database, you reduce the risk of losing data. To further minimize loss of data, you can:

- 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 during your back up of the database.
- Use the REUSEDELAY interval to prevent your copy storage pool volumes from being reused or deleted before they might be needed.
- Use offsite copy storage pool volumes. If, after backing up the storage pools, you mark as OFFSITE all of the volumes in your copy storage pool, they 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.

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, and you will receive message during your restore operation.

## Restoring a Database to its Most Current State

You can use roll-forward recovery to restore a database to its most current state if:

- The log mode was set to ROLLFORWARD continuously from the time the last full backup was created until the time the database was damaged or lost.
- At least one mirror copy of the recovery log is available and all recovery log volumes are intact.

For roll-forward recovery, ADSM uses the last backup series created for the database. A backup series includes a full backup and any incremental backups that apply to that full backup, plus all recovery log records that reflect changes to the database since the last backup in the series was run. To restore the database to its most current state, use the DSMSERV RESTORE DB command.

For example, to restore the database to its most current state, enter:

```
dsmserv restore db
```

The following example shows the importance of storage pool backups with a point-in-time restore. In this example, caching is not active, and the storage pool was not backed up with the BACKUP STGPOOL command.

*9:30am* Client A backs up its data to Volume 1.

*Noon* The system administrator backs up the database.

*1:30pm* Volume 1, containing Client A’s files, is migrated to tape (Volume 2).

*3:00pm* 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.

*3:30pm* The server goes down.

*3:40pm* The system administrator reloads the noon version of the database by using the DSMSERV RESTORE DB command.

*4:40pm* Volume 1 is audited. The following then occurs:

1. The server compares the information on Volume 1 and Volume 2 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 in the previous scenario, the database would have been rolled forward to 3:30pm 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 data base 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. See Chapter 14, “Recovering Data” on page 329 for information about recovery modes.

## Storage Pool Backup and Recovery

You can create backup copies of client files that are stored in your primary backup, archive, and space management pools. The backup copies are stored in *copy storage pools* that can be used to restore the original files in case they become damaged, lost, or unusable.

A typical storage hierarchy migrates from disk to tape. These primary storage pools should be backed up incrementally to the same copy storage pool each day. By backing up to the same copy storage pool, you ensure that files are not recopied as they migrate to the next storage pool.

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.

Backing up storage pools introduces additional space requirements on the database. ADSM maintains control information concerning the location, name, and characteristics of copy storage pool files in the server database. A copy storage pool file does not require an additional database entry. Instead, it adds to the existing database entry for the file information about the copy pool location. The information kept for each file copy requires about 200 bytes of space. In addition, a small amount of space is needed for internal database indexing. As more files are added to the copy storage pools, reevaluate your database size requirements.

## Using Storage Pool Backup Features

| Task | Required Privilege Class |
|---|---|
| Define, back up, or restore storage pools | System, unrestricted storage, or restricted storage (only for those pools to which you authorized) |
| Update volumes | System or operator |
| Restore volumes | System, unrestricted storage, or restricted storage (only for those pools to which you authorized) |
| Query volumes or storage pools | Any administrator |

ADSM provides storage pool backup features that can prepare you for an orderly and successful recovery if a disaster or media loss occurs. With ADSM you can perform incremental backups of primary storage pools. Incremental backups of storage pools, along with database backups and periodic reclamation of offsite volumes, should provide protection to meet most recovery requirements. However, you may sometimes want to do full backups of primary storage pools for the following reasons:

- A concern for the shelf life of the backup media
- Legal or audit requirements that specify how backups should be done
- A preference for the full backup and applied incremental approach

You can do full backups of storage pools by backing up to a new copy storage pool. Additional backups to this new copy pool would be incremental. A full backup process can be set up using this approach.

For example, you want to take full backups weekly and incremental backups daily, and you want to keep four weeks' worth of backups. You can do this by backing up to a new copy storage pool each week. When a copy pool is 4 weeks old, you can delete all the volumes in that copy pool (using the DELETE VOLUME command with DISCARDDATA=YES). You could then reuse the copy pool and its volumes for the next weeks' backups. You would also set the REUSEDELAY value for the copy storage pool to 28 days.

There are some drawbacks to this approach:

- This approach maintains five copies of each file, the primary copy and the four backups. Information on each copy must be kept in the database. The backup copies do not require full database entries (as do the primary copies), but they do take up database space.

## Recovering by Using Backed Up Copies of Storage Pools

By using backed up copies of storage pools, you can recover from a disaster or media loss. “Recovery from a Disaster” on page 353 and “Recovery from Media Loss” on page 354 present scenarios for these situations. Also, the Disaster Recovery Manager features assists in recovering storage pools. See Chapter 15, “Using Disaster Recovery Manager” on page 359 for more details on the DRM feature.

## ADSM Backup and Recovery Scenarios

This section presents scenarios for protecting and recovering an ADSM server. You may want to modify the procedures to meet your requirements.

In these scenarios, a company’s storage hierarchy consists of the default BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL storage pools. All the storage pools use random access media. The server migrates the files in these storage pools to the TAPEPOOL storage pool, which uses cartridge media.

### Backup

The company’s standard procedures include the following:

- Perform a full backup of the database once a week and incremental backups on the other days.
- Perform reclamation of its copy storage pool, once a week. Reclamation for the copy storage pools is turned off at other times.
- Back up its storage pools every night and the server database immediately after the storage pool backups.
- Ship the database and data storage backups to an offsite location every day.

To prepare for a possible disaster, 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. Defines schedules to do the following operations every day:

   a. Incremental backups of the primary storage pools are done each night by issuing the following commands:

```
backup stgpool backuppool disaster-recovery maxprocess=4
backup stgpool archivepool disaster-recovery maxprocess=4
backup stgpool spacemgpool 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 that are backed up to the DISASTER-RECOVERY pool are those files for which a copy does not already exist in the copy storage pool.

b. Select volumes in the DISASTER-RECOVERY copy storage pool that have read-write or read-only access and which are onsite, at least partially filled, and change the volume's access mode to OFFSITE by entering:

```
update volume * access=offsite location='vault site info' -
wherestgpool=disaster-recovery whereaccess=readwrite,readonly -
wherestatus=filling,full
```

c. Backup the database by using the BACKUP DB command:

```
backup db type=incremental devclass=devclassname scratch=yes
```

3. Performs the following operations each night after the scheduled operations previously discussed have completed:

   a. Back up the volume history, device configuration, and server options files to facilitate recovery.

   b. Move the volumes marked offsite and the database backup volumes, volume history files, device configuration files, and server options files (all produced in the previous steps) to the offsite location.

   c. Identify the offsite volumes that are ready to be returned to the onsite location 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.

4. Return the volumes identified in step 3c to the onsite location and update their access to read-write. The REUSEDELAY setting that the company uses for the DISASTER-RECOVERY copy storage pool is set to a large enough value to ensure that no volume is returned to scratch that might be needed if a "non-outdated" database is restored. The volumes that have been empty for the number of days specified for REUSEDELAY are then automatically returned to scratch.

## Recovery from a Disaster

If a company makes the preparations described in “Backup” on page 351 it can recover from a disaster by using ADSM features.

In this scenario, the company’s database and all of its onsite data storage volumes are destroyed by fire. The processor with ADSM installed is not damaged. An administrator restores the server to the state that existed when the last backup was performed by doing the following:

1. Move the latest backup and all of the DISASTER-RECOVERY volumes from the offsite location to the onsite recovery location.
2. If a current, undamaged volume history file exists, save it.
3. Restore the volume history, device configuration, and server options files.
4. Restore the database from the latest backup level by issuing the DSMSERV RESTORE DB command, as described in “Recovering by Using Backed Up Copies of the Database” on page 346.
5. Change the access mode of all the existing primary storage pool volumes in the damaged storage pools to destroyed by issuing the UPDATE VOLUME command:

```
update volume * access=destroyed wherestgpool=backuppool
update volume * access=destroyed wherestgpool=archivepool
update volume * access=destroyed wherestgpool=spacemgpool
update volume * access=destroyed wherestgpool=tapepool
```

6. Identify whether any volumes in the DISASTER-RECOVERY storage pool are currently onsite by using the QUERY VOLUME command. If any are reported, these volumes were destroyed in the disaster and cannot be used for restore processing. Delete each of these copy storage pool volumes from the database by using the DELETE VOLUME command with the DISCARDDATA option. Any files backed up to these volumes will not be restored.
7. Change the access mode of the remaining copy storage pool volumes in the DISASTER-RECOVERY pool to READWRITE by using the UPDATE VOLUME command:

```
update volume * access=readwrite wherestgpool=disaster-recovery
```

**Note:** Users can get files from ADSM at this point. If a user 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 be recovered with no data movement at all. When you update volumes brought from offsite to change their access, you greatly speed recovery time.

8. 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 the end users 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.

9. Recreate files in the primary storage pool from the copies located in the DISASTER-RECOVERY pool by entering:

```
restore stgpool backuppool maxprocess=4
restore stgpool archivepool maxprocess=4
restore stgpool spacemgpool maxprocess=4
restore stgpool tapepool maxprocess=4
```

   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.

10. 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.

## Recovery from Media Loss

If a company makes the preparations described in “Backup” on page 351 it can recover from a disaster 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. Change the access mode of tape volume dsm087 to UNAVAILABLE by using the UPDATE VOLUME command.

2. 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.

3. Set the access mode of the copy volumes identified to UNAVAILABLE.

   **Note:** This precaution prevents the movement of files stored on these volumes until volume dsm087 is restored.

4. Bring the identified volumes to the onsite location and set their access mode to READWRITE.

5. Restore the destroyed files by entering:

```
restore volume dsm087
```

   This command sets the access mode of the dsm087 to DESTROYED and restores all of the files that were stored 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.

6. Set the access mode of the volumes used to restore dsm087 to OFFSITE using the UPDATE VOLUME command.
7. Return the volumes to the offsite location.

## Database Salvage Utilities

In the unlikely event of a seriously corrupted database and when no database backups are available, you can use the ADSM stand-alone database salvage utilities to try to recover. These utilities include the following three components:

- **DSMSERV DUMPDB**, which retrieves as many logical database records as possible into a database dump.

  **Note:** The DSMSERV DUMPDB process does not access the recovery log. Any uncommitted transactions held in the log database are lost.

- **DSMSERV LOADDB**, which loads the dumped database to newly installed server. A badly corrupted database must be salvaged by reinitializing the server before LOADDB can be performed. Use the **DSMSERV INSTALL** program to initialize a new set of database and recovery log volumes.

  You should allocate a new database and recovery log rather than destroy the damaged set, just in case the dump and load process needs to be repeated.

  Storage pool volumes should not be reinitialized. Information in the salvaged database refers to client files stored on these volumes.

  Ensure that the total size of the new database and log volumes are at least as large as the old ones.

- **DSMSERV AUDITDB**, which ensures that the database is returned to a consistent state. You should run AUDITDB with FIX=YES to recover the database. You must also set the recovery log mode to NORMAL or have just completed a database backup. Otherwise, the server may run out of recovery log space during the operation.

After the database audit completes and any errors are corrected, the ADSM server can be restarted. Its configuration should be identical to that just prior to the failure. The only exception could be transactions being processed at the time of failure. Any uncommitted are lost. You should review client activity at the time of failure. Updates to database pages maintained in the recovery log from the database buffer pool are also lost.

## Examples of Recovering a Corrupted Database Using DSMSERV Salvage Utilities

If the server database volumes are still readable, you can try to recover the existing database before reloading from a dump taken previously with the DSMSERV DUMPDB utility. In most cases, the information in existing database volumes is the most current.

### Recovery without a Previous Dump

To recover from existing database volumes, do the following:

1. Halt the server.
2. Ensure that a device configuration file is available and that the DEVCONFIG option is specified for this file in the server options file. If no device configuration file is available, create one manually (see “Recreating a Device Configuration File” on page 341 for details).
3. Dump the database by using the DSMSERV DUMPDB command.
4. Reinitialize database and recovery log by using the DSMSERV INSTALL command.
5. Reload the database by using the DSMSERV LOADDB command.
6. Audit the database by using the DSMSERV AUDITDB command and correct any problems.
7. Start the server.
8. Synchronize the database with volumes in the storage pool by using the AUDIT VOLUME.

## Recovery Using a Previous Dump

To recover from a database copy created by using the DSMSERV DUMPDB command, do the following:

1. Reinitialize the database and recovery log volumes by using the DSMSERV INSTALL command.
2. Ensure that a device configuration file is available and that the DEVCONFIG option is specified for this file in the server options file. If no device configuration file is available, create one manually (see “Recreating a Device Configuration File” on page 341 for details).
3. Reload the database by using the DSMSERV LOADDB command.
4. Audit the database by using the DSMSERV AUDITDB command and correct any problems.
5. Start the server.
6. Synchronize the database with volumes in the storage pool by using the AUDIT VOLUME.

# Chapter 15. Using Disaster Recovery Manager

ADSTAR Distributed Storage Manager offers Disaster Recovery Manager (DRM) as an optional feature that assists with preparing a disaster recovery plan. DRM facilitates an ADSM-based recovery of business applications from backup data that is stored offsite. Recovery may potentially be performed at an alternate site, on replacement computer hardware, by people not familiar with the backed up applications.

The disaster recovery plan can be used to guide an administrator through disaster recovery, as well as for audit purposes to certify the recoverability of the ADSM server. DRM provides automated generation of the server disaster recovery plan file, offsite recovery media management, and storage of client recovery information.

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

| **Section** | **Page** |
|---|---|
| **Concepts:** | |
| Comparing Availability Management to Disaster Recovery Management | 360 |
| Features of Disaster Recovery Manager | 360 |
| About the Disaster Recovery Plan File | 369 |
| Offsite Recovery Media Management | 364 |
| **Tasks:** | |
| Enabling Disaster Recovery Manager | 363 |
| Creating a backup copy of server primary storage pools and database | 363 |
| Sending server backup volumes offsite | 364 |
| Moving reclaimed or expired volumes back onsite | 366 |
| Creating the ADSM server disaster recovery plan file | 368 |
| Storing client machine information | 391 |
| Defining and tracking client recovery media | 394 |
| Customizing disaster recovery manager | 402 |

## Comparing Availability Management to Disaster Recovery Management

This section compares the definitions of availability management with disaster recovery management to show how DRM works with existing backup features of ADSM to provide disaster recovery.

**Availability Management**

Recovery from incidental computer system outages such as disk drive crashes. Down time is often minimized by using disk mirroring and other forms of RAID technology or by maintaining onsite backup copies of data.

Availability management for the ADSM server can be accomplished with ADSM by:

- Mirroring the server database and recovery log
- Backing up storage pools and storing them onsite

**Disaster Recovery Management**

A disaster is a catastrophic interruption of business processing that destroys the ADSM server or clients, or both. Backup data is located offsite to protect it from damage.

Disaster recovery management is accomplished with ADSM by:

- Backing up client data to the ADSM server
- Backing up the server database to removable media and storing the media offsite
- Backing up the primary storage pools and storing the media offsite.
- Using the disaster recovery plan file to assist with the ADSM server recovery

## Features of Disaster Recovery Manager

Disaster Recovery Manager provides the following features:

- Automated generation of a server disaster recovery plan
- Offsite recovery media management
- Storage of client recovery information

### Automated Generation of a Server Disaster Recovery Plan

The PREPARE command automatically queries the required information from the ADSM server and generates a recovery plan file that is based on a pre-defined recovery strategy for the server. The PREPARE command can be scheduled using the ADSM central scheduling capabilities to maintain an up-to-date recovery plan.

The recovery plan file contains the information and procedures necessary to assist with the recovery of the ADSM server. The information in the plan file includes:

- Site-specific server recovery instructions as defined by the administrator (for example, contact names and telephone numbers).
- The sequence of steps necessary to recover an ADSM server.

- List of ADSM database backup and copy storage pool volumes required to perform the recovery. The offsite location where the volumes reside is included.
- Devices required to read the database backup and copy storage pool volumes.
- Space requirements for the ADSM database and recovery log.
- Copy of ADSM server options file, device configuration file, and volume history information file.
- Commands for performing server database recovery and primary storage pool recovery.

## Offsite Recovery Media Management

Knowing the location of offsite recovery media is critical to successful disaster recovery. You can perform the following with DRM's offsite recovery media management:

- Determine what database backup volumes and copy storage pool volumes need to be moved offsite and back onsite.
- Track the media location in the ADSM database.

Database backup volumes and copy storage pool volumes can be treated as logical collections that are selected to move offsite for safekeeping and onsite for reuse or disposal. The reclamation of offsite volumes includes the capability to perform expiration of an ADSM database backup series.

## Storage of Client Recovery Information

DRM allows the following client recovery information to be saved in the ADSM database:

- Business priority
- Machine location, machine characteristics, and machine recovery instructions
- Boot media requirements

In the event of a disaster, DRM query commands provide assistance to help you determine:

- What client machines were lost in the disaster and need to be recovered.
- The priority of the client machines to identify the order to recover machines.
- The machine requirements and boot media requirements.

## Overview of Set Up for Disaster Recovery Manager

This section provides an overview of the tasks involved to begin using DRM. Additional details are provided in subsequent sections.

**Enabling Disaster Recovery Manager**

1. Enable the ADSM server to support Disaster Recovery Manager by using the REGISTER LICENSE command.

**Set Up for Server Recovery**

1. Create backup copies of the server primary storage pools.
2. Create a backup copy of the database.
3. Track the movement of server backup volumes offsite using the MOVE DRMEDIA commands.
4. Create the disaster recovery plan file for the ADSM server by using the PREPARE command.

**Set Up for Storage of Client Recovery Information**

1. Identify and prioritize ADSM clients based on application or business needs, and establish automatic schedules for backing up client data.
2. Define your disaster recovery information for the clients by saving machine information in the ADSM database to include:
   - Business priority and machine location.
   - Associate one or more nodes with a machine.
   - Machine characteristics.
   - Recovery instructions.
3. Define the boot media requirements for the client machines in the ADSM database.
4. Associate one or more machines with the recovery media.

## Enabling Disaster Recovery Manager

| Task | Required Privilege Class |
|---|---|
| Register license for Disaster Recovery Manager | System |

To enable DRM, issue the following command with the license authorization code you received with the DRM feature:

```
register license aaaaaaaaaaaaaaaa
```

## Creating a Backup Copy of Server Primary Storage Pools and Database

Before using DRM to create an ADSM server disaster recovery plan file, you must create a backup copy of your primary storage pools and database.

The following table lists the required privilege classes for performing the tasks in this section.

| Task | Required Privilege Class |
|---|---|
| Backing up your primary storage pools | System, unrestricted storage, or restricted storage |
| Backing up your database | System or unrestricted storage |

Use the following backup commands to create a backup copy of the server primary storage pools and database volumes.

1. Back up your primary storage pools. For example:

```
backup stgpool backuppool cstoragepf
```

   For more information on backing up your storage pools, see “Storage Pool Backup and Recovery” on page 349.

2. Back up your database. For example:

```
backup db devclass=lib8mm type=full volumename=bk06
```

For more information on backing up your database, see “Performing the Database Backups” on page 342.

After you create your backup media, send it offsite for safekeeping. For more information, see “Sending Server Backup Volumes Offsite” on page 364.

When your backup media is offsite, you are ready to create a disaster recovery plan file. For more information, see “Creating the ADSM Server Disaster Recovery Plan File” on page 368.

## Offsite Recovery Media Management

| Task | Required Privilege Class |
|---|---|
| Sending backup volumes offsite and back onsite | Unrestricted storage or operator |

Offsite recovery media management is used during routine operations and defines a process for the following:

- Moving ADSM database backup and copy storage pool volumes offsite for disaster recovery protection.
- Moving ADSM database backup and copy storage pool volumes onsite when they no longer contain valid data.

You can indicate the movement of the volumes with the MOVE DRMEDIA command and display and track their location with the QUERY DRMEDIA command.

Backup volume location information is included in the disaster recovery plan file that is generated by the PREPARE command. In the event of an actual disaster, for example the ADSM server is destroyed, the disaster recovery plan file can be used to provide a list of offsite volumes required at the recovery site. Refer to *ADSM Administrator’s Reference* for a description of the PREPARE command.

### Sending Server Backup Volumes Offsite

DRM uses the following states for database backup and copy storage pool volumes that are sent offsite for disaster recovery protection. The location of a volume is known at each state.

| Volume State | Description |
|---|---|
| **MOUNTABLE** | The volume contains valid data and is accessible to the ADSM server. |
| **NOTMOUNTABLE** | The volume contains valid data and is unavailable to the ADSM server, but is still onsite. |
| **COURIER** | The volume contains valid data and is with the courier. |
| **VAULT** | The volume contains valid data and is at the vault. |

After you have created a backup copy of your primary storage pools and database, you can send your backup media offsite.

To send server backup media offsite, you must mark the volumes as unavailable for ADSM access, and then give the volumes to the courier. Use the following commands to identify the backup volumes written to by the ADSM server backup database and backup storage pool commands, and move these volumes offsite.

1. Issue the following command to identify the newly created copy storage pool and database backup volumes to be moved offsite:

```
query drmedia * wherestate=mountable
```

   ADSM displays information similar to the following:

```
Volume Name                     State                  Last Update
                                                       Date/Time
-----------------------------   ---------------------  -------------------
 tpbk05                         Mountable              06/01/1995 12:00:31
 tpbk99                         Mountable              06/01/1995 12:00:32
 tpbk06                         Mountable              06/01/1995 12:01:03
```

2. Indicate the movement of copy storage pool volumes and database backup volumes whose current state is MOUNTABLE by issuing the following command:

```
move drmedia * wherestate=mountable
```

   This command automatically completes the following process for all volumes with a current state of MOUNTABLE:

   - If the volume resides in a library, the volume is checked out of the library.
   - Updates the volumes' state to NOTMOUNTABLE

3. Package the volumes and give them to the courier for transport to the offsite vault. Issue the following command to have ADSM select volumes whose current state is NOTMOUNTABLE, and record the fact that the volumes have been given to the courier.

```
move drmedia * wherestate=notmountable
```

   This command automatically completes the following process for all volumes with a current state of NOTMOUNTABLE:

   - Updates the volumes' state to COURIER.
   - Updates the volumes' location according to the SET DRMCOURIERNAME. If the SET command has not yet been issued, the default location is COURIER. For more information, see "Courier Name" on page 405.

   Your media containing backed up storage pools and database are now offsite.

4. When the vault location confirms receipt of the volumes, issue the MOVE DRMEDIA command with the WHERESTATE=COURIER parameter. For example:

```
move drmedia * wherestate=courier
```

This command automatically completes the following process for all volumes with a current state of COURIER:

- Updates the volumes' state to VAULT.
- Updates the volumes' location according to the SET VAULTNAME command. If the SET command has not yet been issued, the default location is VAULT. For more information, see "Specify the Vault Name" on page 406.

5. To display a list of volumes that contain valid data at the vault, issue the following command:

```
query drmedia wherestate=vault
```

ADSM displays information similar to the following:

```
Volume Name                                   State     Last Update
                                                        Date/Time
----------------------------------            --------  -------------------
TAPE0P                                        Vault     09/05/1995 10:53:20
TAPE1P                                        Vault     09/05/1995 10:53:20
DBT02                                         Vault     09/05/1995 10:53:20
TAPE3S                                        Vault     09/05/1995 10:53:20
```

See "Example: Routine Operations Using Disaster Recovery Manager" on page 388 for an example that demonstrates sending server backup volumes offsite using MOVE DRMEDIA and QUERY DRMEDIA commands.

## Moving Reclaimed or Expired Volumes Back Onsite

DRM uses the following states for backup volumes that are reclaimed or no longer contain valid data and are to be moved back onsite.

**Volume State** **Description**

**VAULTRETRIEVE** The volumes no longer contain valid data. These volumes are to be returned. They should be given to the courier by the vault operator. For more information on reclamation of offsite copy storage pool volumes, see "Reclamation of Offsite Volumes" on page 258. For information on expiration of database backup volumes, see step 1 on page 367 below.

**COURIERRETRIEVE** The volumes no longer contain valid data and are in the process of being returned by the courier.

When backup volumes stored at the vault location no longer contain valid data, use the following procedure to move those volumes back onsite for reuse or disposal.

1. Use the SET DRMDBBACKUPEXPIREDAYS command to specify the number of days before a database backup series is expired. To ensure that the database can be returned to an earlier level and database references to files in the copy storage pool are still valid, specify this same value for the REUSEDELAY parameter in your storage pool definition.

   A database backup volume is considered eligible for expiration if all of the following conditions are true:

   - The last volume of the series has exceeded the expiration value specified with SET DRMDBBACKUPEXPIREDAYS. The expiration value specifies the number of days that must elapse since the volume was used by database backup.
   - The volume's state is VAULT.
   - The volume is not part of the most recent series (DRM will not expire the most recent database backup series).

   The following example sets the number of days to 30.

```
set drmdbbackupexpiredays 30
```

2. When a backup volume is reclaimed and the ADSM status for a copy storage pool volume is EMPTY or the database backup series is EXPIRED, the volume should be moved back onsite for reuse or disposal. To determine which volumes to retrieve, issue the following command:

```
query drmedia * wherestate=vaultretrieve
```

3. After you request the reclaimed volumes be moved back onsite, and the vault location acknowledges that the volumes have been given to the courier, issue the following command:

```
move drmedia * wherestate=vaultretrieve
```

   This command automatically completes the following process for all volumes with a current state of VAULTRETRIEVE:

   - The state of the volume is changed to COURIERRETRIEVE.
   - The location of the volume is updated according to what is specified in the SET DRMCOURIERNAME command. For more information, see "Courier Name" on page 405.

4. When the courier delivers the volumes, issue the following command to acknowledge that the courier has returned the volumes onsite:

```
move drmedia * wherestate=courierretrieve
```

   This command automatically completes the following process for all volumes with a current state of COURIERRETRIEVE:

   - The volumes are now onsite and can be reused or disposed.
   - The database backup volumes are deleted from the volume history table.
   - For scratch copy storage pool volumes, the record in the ADSM database is deleted. For private copy storage pool volumes, the access is updated to read/write.

For an example scenario that demonstrates moving volumes back onsite, see "Example: Routine Operations Using Disaster Recovery Manager" on page 388.

## Creating the ADSM Server Disaster Recovery Plan File

When the system administrator invokes the PREPARE command, DRM automatically queries the ADSM server for required information to generate a disaster recovery plan file.

To create a disaster recovery plan file, issue the PREPARE command.

In the following example, the PREPARE command is issued with the PLANPREFIX parameter to generate the recovery plan file in directory /u/server/recoveryplans/:

```
prepare planprefix=/u/server/recoveryplans/
```

The plan file name always includes the date and time (yyyymmdd.hhmmss) when the PREPARE command is issued. For example:

```
/u/server/recoveryplans/19950925.120532
```

For details about specifying the location of the disaster recovery plan file, see "Prefix for Recovery Plan File" on page 404, and also refer to the PREPARE command in the *ADSM Administrator's Reference*.

DRM creates one copy of the disaster recovery plan file. It is recommended that you create multiple copies of your disaster recovery plan for safekeeping. For example, keep copies in print, on diskettes, or on NFS-mounted disk space that is physically located offsite.

The PREPARE command should be issued or scheduled to run after back up of your storage pools and database, and the volumes have been marked to be sent offsite. This ensures that your disaster recovery plan file is kept up-to-date.

Each time the PREPARE command generates a new disaster recovery plan file, the previous file is not deleted. It is recommended that you periodically delete downlevel recovery plan files.

## About the Disaster Recovery Plan File

The disaster recovery plan file contains the information required for recovery of an ADSM server to the point in time represented by the last database backup operation that is completed before the PREPARE command is issued.

The recovery information is organized into stanzas within the disaster recovery plan file. Each stanza in the recovery plan file has a unique name. These names are listed in Table 26 on page 377.

In the event of a disaster, the administrator can use the recovery plan as a guide to recovering the ADSM server. Optionally, the administrator can use an editor or a locally written procedure (for example, a modified version of the awk script *planexpl.awk.smp* that is shipped with DRM) to break out the recovery plan file stanzas into multiple useful files.

## About the Recovery Plan File Stanzas

This section describes the stanzas in the recovery plan file and how to use the stanzas.

These stanza files can be categorized as follows:

**Command stanzas**
Consist of shell scripts and ADSM macros. These stanzas can be viewed, printed, updated, or executed as part of the disaster recovery process.

**Site-specific instruction stanzas**
These stanzas include recovery instructions specific to your site. They can be printed, updated, and used during server recovery.

**Server requirements stanzas**
These stanzas include the database and recovery log requirements, and volume and device requirements. They can be viewed or printed.

**Configuration file stanzas**
Consist of the volume history, device configuration, and server options files.

The stanzas are presented in the expected order that they would be used by the recovery team.

**Note:** The RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE and RECOVERY.SCRIPT.NORMAL.MODE stanzas contain the shell scripts that *invoke* the scripts and macros contained in the other stanzas.

The following are descriptions of the stanzas:

- PLANFILE.DESCRIPTION

  Identifies the server for this recovery plan, and the date and time the recovery plan is created.

- PLANFILE.TABLE.OF.CONTENTS

  Provides a list of the stanzas in this recovery plan.

- SERVER.REQUIREMENTS

  Identifies the database and recovery log storage requirements for this server. At the recovery site, you will need a replacement server machine that has enough disk space to install the database and recovery log volumes.

- RECOVERY.INSTRUCTIONS.GENERAL

  Identifies site specific instructions the server administrator has manually edited in the source text file identified by *instructionsprefix*RECOVERY.INSTRUCTIONS.GENERAL. It is recommended that the instructions include the overall recovery strategy, key contact names, overview of key applications backed up by this server and so on.

  **Note:** *Instructionsprefix* is the prefix portion of the file name; see “Prefix for Recovery Instructions” on page 404.

  For more information on editing the text source file, see “Customizing the Site Specific RECOVERY.INSTRUCTIONS” on page 407.

- RECOVERY.INSTRUCTIONS.OFFSITE

  Identifies site specific instructions the server administrator has manually edited to the source text file identified by *instructionsprefix*RECOVERY.INSTRUCTIONS.OFFSITE. It is recommended that the instructions include the name and location of the offsite vault and how to contact the vault.

  **Note:** *Instructionsprefix* is the prefix portion of the file name; see “Prefix for Recovery Instructions” on page 404.

  For more information on editing the text source file, see “Customizing the Site Specific RECOVERY.INSTRUCTIONS” on page 407.

- RECOVERY.INSTRUCTIONS.INSTALL

  Identifies site specific instructions the server administrator has manually edited to the source text file identified by *instructionsprefix*RECOVERY.INSTRUCTIONS.INSTALL. It is recommended that the instructions include how to rebuild the base server machine and where backup copies of the system image are located.

  **Note:** *Instructionsprefix* is the prefix portion of the file name; see “Prefix for Recovery Instructions” on page 404.

  For more information on editing the text source file, see “Customizing the Site Specific RECOVERY.INSTRUCTIONS” on page 407.

- RECOVERY.INSTRUCTIONS.DATABASE

  Identifies site specific instructions the server administrator has manually edited to the source text file identified by *instructionsprefix*RECOVERY.INSTRUCTIONS.DATABASE. Information in this stanza should include how to prepare for the ADSM server database recovery. For example, if the backup device is an 8mm library, you may want to provide instructions on how to initialize or load the backup volumes

  **Note:** *Instructionsprefix* is the prefix portion of the file name; see "Prefix for Recovery Instructions" on page 404.

  For more information on editing the text source file, see "Customizing the Site Specific RECOVERY.INSTRUCTIONS" on page 407.

- RECOVERY.INSTRUCTIONS.STGPOOL

  Identifies site specific instructions the server administrator has manually edited to the source text file identified by *instructionsprefix*RECOVERY.INSTRUCTIONS.STGPOOL. It is recommended that the instructions include what applications are backed up in what copy storage pools.

  **Note:** *Instructionsprefix* is the prefix portion of the file name; see "Prefix for Recovery Instructions" on page 404.

  For more information on editing the text source file, see "Customizing the Site Specific RECOVERY.INSTRUCTIONS" on page 407.

- RECOVERY.VOLUMES.REQUIRED

  Provides a list of the database backup and copy storage pool volumes required to recover the ADSM server. The location and device class names for the required volumes are also displayed. If you are using the MOVE DRMEDIA command for offsite recovery media management, a blank location field means that the volumes are onsite. This list can be used as the basis of periodic audits for the inventory of volumes at the courier and offsite vault. In the event of a disaster, this list would be used to collect the required volumes before recovery of the server is started.

- RECOVERY.DEVICES.REQUIRED

  Provides details about the devices required to read the backup volumes.

- RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE

  Contains a shell script with the command required to restore the server database and restart the server. Use it as a guide and execute the commands as needed from a command line, or optionally copy it to a file, modify it and the files it references, and execute the shell script. At the completion of these steps, client requests for file restores will be satisfied directly from copy storage pool volumes.

  This script will often need modification at the recovery site because of differences between the original and the replacement systems.

  This script restores the server options, volume history, and device configuration information files.

This script invokes shell scripts contained in the following stanzas:

LOGANDDB.VOLUMES.CREATE
LOGANDDB.VOLUMES.INSTALL

This script also invokes the ADSM macros contained in the following stanzas:

COPYSTGPOOL.VOLUMES.AVAILABLE
COPYSTGPOOL.VOLUMES.DESTROYED
PRIMARY.VOLUMES.DESTROYED.

To help understand the operations being performed in this shell script, see "ADSM Backup and Recovery Scenarios" on page 351.

To invoke this script, the following three positional parameters must be specified:

- $1 (the administrator ID)
- $2 (the administrator password)
- $3 (the servername)

For example, to invoke this script using an administrator ID of don, password of mox, server name of prodadsm, enter the following command:

```
planprefix/RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE don mox prodadsm
```

For additional information on planprefix, see "Prefix for Recovery Plan File" on page 404.

- RECOVERY.SCRIPT.NORMAL.MODE

  Contains a shell script with the commands required to restore the server primary storage pools. Use it as a guide and execute the commands as needed from a command line, or optionally copy it to a file, modify it and the files it references, and execute it.

  At the completion of these steps, client requests for file restores are satisfied from primary storage pool volumes. Clients should also be able to resume file backup, archive, and migration functions.

  This script will often need modification at the recovery site because of differences between the original and the replacement systems.

  This script invokes the shell script contained in the following stanza:

  PRIMARY.VOLUMES.REPLACEMENT.CREATE

  This script also invokes the ADSM macros contained in stanzas:

  PRIMARY.VOLUMES.REPLACEMENT
  STGPOOLS.RESTORE

  To help understand the operations being performed in this shell script, see "ADSM Backup and Recovery Scenarios" on page 351.

- LOGANDDB.VOLUMES.CREATE

  Contains a shell script with the command required to recreate the ADSM server database and log volumes that existed before the disaster. You can use it as a guide and execute the commands as needed from a command line, or optionally copy it to a file, modify it, and execute it.

This shell script is invoked by the shell script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

- LOGANDDB.VOLUMES.INSTALL

  Contains a shell script with the commands required to install the ADSM server database and log volumes that existed before the disaster. You can use it as a guide and execute the commands as needed from a command line, or optionally copy it to a file, modify it, and execute it.

  This shell script is invoked by the shell script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

- COPYSTGPOOL.VOLUMES.AVAILABLE

  Contains an ADSM macro to mark copy storage pool volumes that were moved offsite as moved back onsite. You can use it as a guide and execute the ADSM administrative commands as needed from a command line or graphical user interface, or optionally copy it to a file, modify it, and execute it.

  This macro is invoked by the shell script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

  In the event of a disaster, compare the copy storage pool volumes listed in this stanza with the volumes you have obtained from the courier and the offsite vault. If you have not physically obtained all volumes, you should remove the entries for the missing volumes from this stanza.

- COPYSTGPOOL.VOLUMES.DESTROYED

  Contains an ADSM macro to mark copy storage pool volumes as unavailable that were onsite at the time of the disaster. You can use it as a guide and execute the ADSM administrative commands as needed from a command line or graphical user interface, or optionally copy it to a file, modify it, and execute it.

  This macro is invoked by the shell script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

  In the event of a disaster, compare the copy storage pool volumes listed in this stanza with the volumes that were left onsite. If you have physically obtained any of the volumes, you should remove their entries from this stanza.

- PRIMARY.VOLUMES.DESTROYED

  Contains an ADSM macro to mark primary storage pool volumes as destroyed that were onsite at the time of disaster. You can use it as a guide and execute the ADSM administrative commands as needed from a command line or graphical user interface, or optionally copy it to a file, modify it, and execute it.

  This macro is invoked by the shell script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

  In the event of a disaster, compare the primary storage pool volumes listed in this stanza with the volumes that were onsite. If you have physically obtained any of the volumes and have determined they are useable, you should remove their entries from here.

- PRIMARY.VOLUMES.REPLACEMENT.CREATE

  Contains a shell script with the commands required to recreate the ADSM server primary storage pool volumes that existed with device class DISK. You can use it as a guide and execute the commands as needed from a command line, or optionally copy it to a file, modify it, and execute it.

  This shell script is invoked by the shell script RECOVERY.SCRIPT.NORMAL.MODE.

  The SET DRMPLANVPOSTFIX character is appended to the end of the names of the original volumes listed in this stanza. This appended character serves two alternative strategies:

  – Makes it easy to find volume names that require renaming in the stanzas. Before using the volume names, change these names to new names that are valid for the device class and valid on the replacement system.
  – Automatically generate a new name that can be used by the replacement server. This strategy requires that a previously planned naming convention take into account the appended post fix character.

    **Notes:**

    1. Replacement primary volume names must be different from any other original volume name or replacement name.

    2. The ADSM server RESTORE STGPOOL command restores storage pools on a logical basis. This means that there is no one-to-one relationship between an original volume and its replacement.

    3. There will be entries for the same volumes in PRIMARY.VOLUMES.REPLACEMENTS.

- PRIMARY.VOLUMES.REPLACEMENT

  Contains an ADSM macro to define primary storage pool volumes to the ADSM server. You can use it as a guide and execute the ADSM administrative commands as needed from a command line or graphical user interface, or optionally copy it to a file, modify it, and execute it.

  This macro is invoked by the shell script RECOVERY.SCRIPT.NORMAL.MODE.

  Primary storage pool volumes that get entries in this stanza have at least one of the following three characteristics:

  1. Original volume in a storage pool whose device class was DISK.
  2. Original volume in a storage pool with MAXSCRATCH=0.
  3. Original volume in a storage pool and volume scratch attribute=no.

  The SET DRMPLANVPOSTFIX character is appended to the end of the names of the original volumes listed in this stanza. This appended character serves two alternative strategies:

  - Makes it easy to find volume names that require renaming in the stanzas. Before using the volume names, change these names to new names that are valid for the device class and on the replacement system.
  - Automatically generate a new name that can be used by the replacement server. This strategy requires that a previously planned naming convention take into account the appended post fix character.

  **Notes:**

  1. Replacement primary volume names must be different from any other original volume name or replacement name.
  2. The ADSM server RESTORE STGPOOL command restores storage pools on a logical basis. This means that there is no one-to-one relationship between an original volume and its replacement.
  3. There could be entries for the same volume in PRIMARY.VOLUMES.REPLACEMENT.CREATE and PRIMARY.VOLUMES.REPLACEMENT if the volume has a device class of DISK.

- STGPOOLS.RESTORE

  Contains an ADSM macro to restore the primary storage pools. You can use it as a guide and execute the ADSM administrative commands as needed from a command line or graphical user interface, or optionally copy it to a file, modify it, and execute it.

  This macro is invoked by the shell script RECOVERY.SCRIPT.NORMAL.MODE.

- VOLUME.HISTORY.FILE

  Contains a copy of the server volume history information that existed at the time PREPARE was run. The volume history file is very important to server recovery because the DSMSERV RESTORE DB command uses the volume history file to determine what volumes are required to restore the database. It is referenced by the shell script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

  The following rules are used to determine where the volume history file is placed at restore time:

  - If VOLUMEHISTORY entries are defined in the server options file, PREPARE uses the fully qualified file name associated with the first entry. If the specified file name does not begin with a directory specification (for example, ‘.’ or ‘/’), PREPARE adds the prefix *volhprefix*.
  - If a VOLUMEHISTORY entry is not defined in the server options file, PREPARE uses the default name *volhprefix* followed by drmvolh.txt. For example, if *volhprefix* is /usr/lpp/adsmserv/bin/ the file name used by PREPARE is /usr/lpp/adsmserv/bin/drmvolh.txt.

**Note on the volhprefix:**

The *volhprefix* is set based on the following:

- If the environmental variable DSMSERV_DIR has been defined, it is used as the *volhprefix*.
- If the environmental variable DSMSERV_DIR has not been defined, the directory where the ADSM server is started from is used as the *volhprefix*.

If a fully qualified file name was not specified for the VOLUMEHISTORY option in the server options file, PREPARE adds it to the stanza DSMSERV.OPT.FILE.

- DEVICE.CONFIGURATION.FILE

  Contains a copy of the server device configuration information that existed at the time PREPARE was run. The device configuration file is very important to server recovery because the DSMSERV RESTORE DB command uses this file to read the database backup volumes. It is referenced by the shell script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

  The following rules are used to determine where the device configuration file is placed at restore time:

  - If DEVCONFIG entries are defined in the server options file, PREPARE uses the fully qualified file name associated with the first entry. If the specified file name does not begin with a directory specification (for example, '.' or '/'), PREPARE adds the prefix *devcprefix*.
  - If a DEVCONFIG entry is not defined in the server options file, PREPARE uses the default name *devcprefix* followed by drmdevc.txt. For example, if *devcprefix* is /usr/lpp/adsmserv/bin/ the file name used by PREPARE is /usr/lpp/adsmserv/bin/drmdevc.txt.

  **Note on the devcprefix:**

  The *devcprefix* is set based on the following:

  - If the environmental variable DSMSERV_DIR has been defined, it is used as the *devcprefix*.
  - If the environmental variable DSMSERV_DIR has not been defined, the directory where the ADSM server is started from is used as the *devcprefix*.

  If a fully qualified file name was not specified for the DEVCONFIG option in the server options file, PREPARE adds it to the stanza DSMSERV.OPT.FILE.

- DSMSERV.OPT.FILE

  Contains a copy of the server options file used when the server was started. The server options file sets various server operating characteristics.

  This stanza is referenced by the shell script RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE.

Table 26 lists the recovery plan file stanzas, and indicates what type of administrative processing is required during set up of DRM, routine processing, or disaster recovery. The table also indicates whether the stanza contains a macro, a shell script, or a configuration file.

*Table 26 (Page 1 of 2). Administrative Tasks Associated with the Disaster Recovery Plan File*

| Stanza Name | Admin. Action During Setup or Periodic Updates | Recommended Admin. Action During Routine Processing | Admin. Action During Disaster Recovery |
|---|---|---|---|
| PLANFILE.DESCRIPTION | None | None | None |
| PLANFILE.TABLE.OF.CONTENTS | None | None | None |
| SERVER.REQUIREMENTS | None | None | None |
| RECOVERY.INSTRUCTIONS.GENERAL | Optionally edit source file associated with stanza | None | None |
| RECOVERY.INSTRUCTIONS.OFFSITE | Optionally edit source file associated with stanza | None | None |
| RECOVERY.INSTRUCTIONS.INSTALL | Optionally edit source file associated with stanza | None | None |
| RECOVERY.INSTRUCTIONS.DATABASE | Optionally edit source file associated with stanza | None | None |
| RECOVERY.INSTRUCTIONS.STGPOOL | Optionally edit source file associated with stanza | None | None |
| RECOVERY.VOLUMES.REQUIRED | None | MOVE DRMEDIA | None |
| RECOVERY.DEVICES.REQUIRED | None | None | None |
| RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE script | None | None | Optionally edit/execute |
| RECOVERY.SCRIPT.NORMAL.MODE script | None | None | Optionally edit/execute |
| LOGANDDB.VOLUMES.CREATE script | None | None | Optionally edit/execute |
| LOGANDDB.VOLUMES.INSTALL script | None | None | Optionally edit/execute |
| COPYSTGPOOL.VOLUMES.AVAILABLE macro | None | MOVE DRMEDIA | Optionally edit/execute |
| COPYSTGPOOL.VOLUMES.DESTROYED macro | None | MOVE DRMEDIA | Optionally edit/execute |

*Table 26 (Page 2 of 2). Administrative Tasks Associated with the Disaster Recovery Plan File*

| **Stanza Name** | **Admin. Action During Setup or Periodic Updates** | **Recommended Admin. Action During Routine Processing** | **Admin. Action During Disaster Recovery** |
|---|---|---|---|
| PRIMARY.VOLUMES.DESTROYED script | None | None | Optionally edit/execute |
| PRIMARY.VOLUMES.REPLACEMENT.CREATE script | None | None | Optionally edit/execute |
| PRIMARY.VOLUMES.REPLACEMENT macro | None | None | Optionally edit/execute |
| STGPOOLS.RESTORE macro | None | None | Optionally edit/execute |
| VOLUME.HISTORY.FILE configuration file | None | None | Optionally copy |
| DEVICE.CONFIGURATION.FILE configuration file | None | None | Optionally edit/copy |
| DSMSERV.OPT.FILE configuration file | None | None | Optionally edit/copy |

**Note:** In the column “User Action During Setup or Periodic Updates,” the action of None means that DRM automatically collects this information for the file.

## Example of a Disaster Recovery Plan File

The following is an example of a disaster recovery plan file generated by PREPARE.

```
begin PLANFILE.DESCRIPTION

Recovery Plan for ADSM Server ADSM
Created by DRM PREPARE on 08/11/1995 10:20:34
ADSM Server for AIX-RS/6000 - Version 2, Release 1, Level 0.0/0.0

end PLANFILE.DESCRIPTION

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin PLANFILE.TABLE.OF.CONTENTS

PLANFILE.DESCRIPTION
PLANFILE.TABLE.OF.CONTENTS

Server Recovery Stanzas:
  SERVER.REQUIREMENTS
  RECOVERY.INSTRUCTIONS.GENERAL
  RECOVERY.INSTRUCTIONS.OFFSITE
  RECOVERY.INSTRUCTIONS.INSTALL
  RECOVERY.INSTRUCTIONS.DATABASE
  RECOVERY.INSTRUCTIONS.STGPOOL
  RECOVERY.VOLUMES.REQUIRED
  RECOVERY.DEVICES.REQUIRED
  RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE script
  RECOVERY.SCRIPT.NORMAL.MODE script
  LOGANDDB.VOLUMES.CREATE script
  LOGANDDB.VOLUMES.INSTALL script
  COPYSTGPOOL.VOLUMES.AVAILABLE macro
  COPYSTGPOOL.VOLUMES.DESTROYED macro
  PRIMARY.VOLUMES.DESTROYED script
  PRIMARY.VOLUMES.REPLACEMENT.CREATE script
  PRIMARY.VOLUMES.REPLACEMENT macro
  STGPOOLS.RESTORE macro
  VOLUME.HISTORY.FILE
  DEVICE.CONFIGURATION.FILE
  DSMSERV.OPT.FILE

end PLANFILE.TABLE.OF.CONTENTS

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

*Figure 72 (Part 1 of 10). Example of a Disaster Recovery Plan File*

```
begin SERVER.REQUIREMENTS

Database Requirements Summary:

    Available Space (MB): 20
  Assigned Capacity (MB): 20
        Pct. Utilization: 2.2
Maximum Pct. Utilization: 2.2
        Physical Volumes: 2

Recovery Log Requirements Summary:

    Available Space (MB): 20
  Assigned Capacity (MB): 20
        Pct. Utilization: 4.4
Maximum Pct. Utilization: 4.8
        Physical Volumes: 2

end SERVER.REQUIREMENTS

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin RECOVERY.INSTRUCTIONS.GENERAL

 This ADSM server contains the backup and archive data for FileRight Company
 accounts receivable system. It also is used by various end users in the
 finance and materials distribution organizations.
 The storage administrator in charge of this server is Jane Doe 004-001-0006.
 If a disaster is declared, here is the outline of steps that must be completed.
 1. Determine the recovery site. Our alternate recovery site vendor is IBM
    BRS in Tampa, Fl, USA 213-000-0007.
 2. Get the list of required recovery volumes from this recovery plan file
    and contact our offsite vault so that they can start pulling the
    volumes for transfer to the recovery site.
 3. etc...

end RECOVERY.INSTRUCTIONS.GENERAL

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin RECOVERY.INSTRUCTIONS.OFFSITE

 Our offsite vaulting vendor is OffsiteVault Inc.
 Their telephone number is 514-555-2341. Our account rep is Joe Smith.
 Our account number is 1239992. Their address is ...
 Here is a map to their warehouse ...
 Our courier is ...

end RECOVERY.INSTRUCTIONS.OFFSITE

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

*Figure 72 (Part 2 of 10). Example of a Disaster Recovery Plan File*

```
begin RECOVERY.INSTRUCTIONS.INSTALL

 The base ADSM server system is AIX 4.1.2 running on an RS6K model 320.
 Use mksysb volume serial number svrbas to restore this system image.
 A copy of this mksysb tape is stored at the vault. There is also a copy
 in bldg 24 room 4 cabinet a. The image includes the ADSM server code.
 The system programmer responsible for this image is Fred Myers.
 Following are the instructions to do a mksysb based OS install:


end RECOVERY.INSTRUCTIONS.INSTALL

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin RECOVERY.INSTRUCTIONS.DATABASE

No recovery instructions defined.

end RECOVERY.INSTRUCTIONS.DATABASE

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin RECOVERY.INSTRUCTIONS.STGPOOL

No recovery instructions defined.

end RECOVERY.INSTRUCTIONS.STGPOOL

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin RECOVERY.VOLUMES.REQUIRED

Volumes required for data base restore

 Location =
  Device Class = LIB8MM
  Volume Name =
   TPBK08
 Location = OffsiteVault Inc.
  Device Class = LIB8MM
  Volume Name =
   TPBK06

Volumes required for storage pool restore

 Location = OffsiteVault Inc.
  Copy Storage Pool = CSTORAGEPF
  Device Class = LIB8MM
  Volume Name =
   TPBK05
   TPBK07

end RECOVERY.VOLUMES.REQUIRED

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

*Figure 72 (Part 3 of 10). Example of a Disaster Recovery Plan File*

```
begin RECOVERY.DEVICES.REQUIRED

 Purpose: Description of the devices required to read the
          volumes listed in the recovery volumes required stanza.

             Device Class Name: LIB8MM
        Device Access Strategy: Sequential
            Storage Pool Count: 2
                   Device Type: 8MM
                        Format:
         Est/Max Capacity (MB): 4.0
                   Mount Limit: 2
              Mount Wait (min):
         Mount Retention (min): 10
                  Label Prefix:
                       Library: RLLIB
                     Directory:
Last Update by (administrator): ROOT
         Last Update Date/Time: 08/11/1995 10:18:34

end RECOVERY.DEVICES.REQUIRED

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE script

#!/bin/ksh
set -x

 # Purpose: This script contains the steps required to recover the server
 #   to the point where client restore requests can be satisfied
 #   directly from available copy storage pool volumes.
 # Note: This script assumes that all volumes necessary for the restore have
 #   been retrieved from the vault and are available. This script assumes
 #   the recovery  environment is compatible (essentially the same) as the
 #   original.  Any deviations require modification to this script and the
 #   macros and shell scripts it runs.  Alternatively, you can use this
 #   script as a guide, and manually execute each step.
```

*Figure 72 (Part 4 of 10). Example of a Disaster Recovery Plan File*

```
 # Set the ADSM server working directory.
cd /usr/lpp/adsmserv/bin

 # Load the kernel extension.
/usr/lpp/adsmserv/bin/loadpkx -f pkmonx

 # Restore server options, volume history, device configuration files.
cp /prepare/DSMSERV.OPT.FILE \
    /usr/lpp/adsmserv/bin/dsmserv.optx
cp /prepare/VOLUME.HISTORY.FILE \
    /usr/lpp/adsmserv/bin/volhistory.txtx
cp /prepare/DEVICE.CONFIGURATION.FILE \
    /usr/lpp/adsmserv/bin/devconfig.txtx

export DSMSERV_CONFIG=/usr/lpp/adsmserv/bin/dsmserv.optx

export DSMSERV_DIR=/usr/lpp/adsmserv/bin

 # Create and format log and database files.
/prepare/LOGANDDB.VOLUMES.CREATE 2>&1 \
| tee /prepare/LOGANDDB.VOLUMES.CREATE.log

 # Install the log and database files.
/prepare/LOGANDDB.VOLUMES.INSTALL 2>&1 \
| tee /prepare/LOGANDDB.VOLUMES.INSTALL.log

 # Restore the ADSM server database to latest version backed up per the
 # volume history file.
/usr/lpp/adsmserv/bin/dsmserv restore db todate=08/11/1995 totime=10:20:22

 # Start the server.
nohup /usr/lpp/adsmserv/bin/dsmserv &
print Please start new ADSM server console with command dsmadmc -CONSOLE.
print Press enter to continue recovery script execution.
read pause

 # Tell ADSM Server these copy storage pool volumes are available for use.
 # Recovery Administrator: Remove from macro any volumes not obtained from vault.
dsmadmc -id=$1 -pass=$2 -serv=$3 -ITEMCOMMIT \
  -OUTFILE=/prepare/COPYSTGPOOL.VOLUMES.AVAILABLE.log \
     macro /prepare/COPYSTGPOOL.VOLUMES.AVAILABLE

 # Volumes in this macro were not marked as 'offsite' at the time
 # PREPARE ran. They were likely destroyed in the disaster.
 # Recovery Administrator: Remove from macro any volumes not destroyed.
dsmadmc -id=$1 -pass=$2 -serv=$3 -ITEMCOMMIT \
  -OUTFILE=/prepare/COPYSTGPOOL.VOLUMES.DESTROYED.log \
     macro /prepare/COPYSTGPOOL.VOLUMES.DESTROYED
```

*Figure 72 (Part 5 of 10). Example of a Disaster Recovery Plan File*

```
 # Mark primary storage pool volumes as ACCESS=DESTROYED.
 # Recovery administrator: Remove from macro any volumes not destroyed.
dsmadmc -id=$1 -pass=$2 -serv=$3 -ITEMCOMMIT \
  –OUTFILE=/prepare/PRIMARY.VOLUMES.DESTROYED.log \
     macro /prepare/PRIMARY.VOLUMES.DESTROYED

end RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE script

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin RECOVERY.SCRIPT.NORMAL.MODE script

#!/bin/ksh
set -x

 # Purpose: This script contains the steps required to recover the server
 #          primary storage pools. This mode allows you to return the
 #          copy storage pool volumes to the vault and to run the
 #          server as normal.
 # Note: This script assumes that all volumes necessary for the restore
 #   have been retrieved from the vault and are available. This script
 #   assumes the recovery  environment is compatible (essentially the
 #   same) as the original. Any deviations require modification to this
 #   script and the macros and shell scripts it runs. Alternatively,
 #   you can use this script as a guide, and manually execute each step.

 # Format replacement volumes in the primary storage pools (If any
 # are implemented as disk but not logical volume.)
 # Recovery administrator: Edit script for your replacement volumes.
/prepare/PRIMARY.VOLUMES.REPLACEMENT.CREATE 2>&1 \
| tee /prepare/PRIMARY.VOLUMES.REPLACEMENT.CREATE.log

 # Define replacement volumes in the primary storage pools. Must
 # have different name than original.
 # Recovery administrator: Edit macro for your replacement volumes.
dsmadmc -id=$1 -pass=$2 -serv=$3 -ITEMCOMMIT \
  -OUTFILE=/prepare/PRIMARY.VOLUMES.REPLACEMENT.log \
     macro /prepare/PRIMARY.VOLUMES.REPLACEMENT

 # Restore the primary storage pools from the copy storage pools.
dsmadmc -id=$1 -pass=$2 -serv=$3 -ITEMCOMMIT \
  -OUTFILE=/prepare/STGPOOLS.RESTORE.log \
     macro /prepare/STGPOOLS.RESTORE

end RECOVERY.SCRIPT.NORMAL.MODE script

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

*Figure 72 (Part 6 of 10). Example of a Disaster Recovery Plan File*

```
begin LOGANDDB.VOLUMES.CREATE script

#!/bin/ksh
set -x

 # Purpose: Create log and database volumes.
 # Recovery Administrator: Run this to format ADSM server log and database
 #  volumes.

  print Create ADSM data base volume /dev/rDSM1509db01x 12M
 mklv -y DSM1509db01x veggie2 4
   print Create ADSM data base volume /usr/lpp/adsmserv/bin/db01x 8M
 dsmfmt -m -db /usr/lpp/adsmserv/bin/db01x 8

  print Create ADSM log volume /dev/rDSM1509lg01x 12M
 mklv -y DSM1509lg01x veggie2 4

  print Create ADSM log volume /usr/lpp/adsmserv/bin/lg01x 8M
 dsmfmt -m -log /usr/lpp/adsmserv/bin/lg01x 8

end LOGANDDB.VOLUMES.CREATE script

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin LOGANDDB.VOLUMES.INSTALL script

#!/bin/ksh
set -x

 # Purpose: Install the log and database volumes.
 # Recovery Administrator: Run this to initialize an ADSM server.

 /usr/lpp/adsmserv/bin/dsmserv install \
   2 /dev/rDSM1509lg01x /usr/lpp/adsmserv/bin/lg01x \
   2 /dev/rDSM1509db01x /usr/lpp/adsmserv/bin/db01x

end LOGANDDB.VOLUMES.INSTALL script

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin COPYSTGPOOL.VOLUMES.AVAILABLE macro

 /* Purpose: Mark copy storage pool volumes as available for use in recovery. */
 /* Recovery Administrator: Remove any volumes that have not been obtained    */
 /*   from the vault or are not available for any reason.                     */
 /* Note: It is possible to use the mass update capability of the ADSM        */
 /*   UPDATE command instead of issuing an update for each volume. However,   */
 /*   the 'update by volume' technique used here allows you to select         */
 /*   a subset of volumes to be processed.                                    */

 upd vol TPBK05 acc=READW wherestg=CSTORAGEPF
 upd vol TPBK07 acc=READW wherestg=CSTORAGEPF

end COPYSTGPOOL.VOLUMES.AVAILABLE macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

*Figure 72 (Part 7 of 10). Example of a Disaster Recovery Plan File*

```
begin COPYSTGPOOL.VOLUMES.DESTROYED macro

 /* Purpose: Mark destroyed copy storage pool volumes as unavailable.      */
 /*   Volumes in this macro were not marked as 'offsite' at the time the   */
 /*   PREPARE ran. They were likely destroyed in the disaster.             */
 /* Recovery Administrator: Remove any volumes that were not destroyed.    */


end COPYSTGPOOL.VOLUMES.DESTROYED macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin PRIMARY.VOLUMES.DESTROYED macro

 /* Purpose: Mark primary storage pool volumes as ACCESS=DESTROYED.        */
 /* Recovery administrator: Delete any volumes listed here                 */
 /*   that you do not want to recover.                                     */
 /* Note: It is possible to use the mass update capability of the ADSM     */
 /*   UPDATE command instead of issuing an update for each volume. However */
 /*   the 'update by volume' technique used here allows you to select      */
 /*   a subset of volumes to be marked as destroyed.                       */

 upd vol /dev/rDSM1509bk02 acc=DESTROYED wherestg=BACKUPPOOL
 upd vol /usr/lpp/adsmserv/bin/bk01x acc=DESTROYED wherestg=BACKUPPOOL
 upd vol /usr/lpp/adsmserv/bin/bk03 acc=DESTROYED wherestg= BACKUPPOOLF
 upd vol BACK4X acc=DESTROYED wherestg=BACKUPPOOLT

end PRIMARY.VOLUMES.DESTROYED macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin PRIMARY.VOLUMES.REPLACEMENT.CREATE script

#!/bin/ksh
set -x

 # Purpose: Create replacement volumes for primary storage pools that
 #   use device class DISK.
 # Recovery administrator: Edit this section for your replacement
 #   volume names. New name must be unique, i.e. different from any
 #   original or other new name.

   print Replace /dev/rDSM1509bk02 DISK 16M in BACKUPPOOL
 mklv -y DSM1509bk02@ veggie2 4

   print Replace /usr/lpp/adsmserv/bin/bk01x DISK 5M in BACKUPPOOL
 dsmfmt -m -data /usr/lpp/adsmserv/bin/bk01x@ 5

end PRIMARY.VOLUMES.REPLACEMENT.CREATE script

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

*Figure 72 (Part 8 of 10). Example of a Disaster Recovery Plan File*

```
begin PRIMARY.VOLUMES.REPLACEMENT macro

 /* Purpose: Define replacement primary storage pool volumes for either:   */
 /*   1. Original volume in a storage pool whose device class was DISK.    */
 /*   2. Original volume in a storage pool with MAXSCRATCH=0.              */
 /*   3. Original volume in a storage pool and volume scratch=no.          */
 /* Recovery administrator: Edit this section for your replacement         */
 /*   volume names. New name must be unique, i.e. different from any       */
 /*   original or other new name.                                          */

   /* Replace /dev/rDSM1509bk02 DISK 16M in BACKUPPOOL */
 def vol BACKUPPOOL /dev/rDSM1509bk02@ acc=READW

   /* Replace /usr/lpp/adsmserv/bin/bk01x DISK 5M in BACKUPPOOL */
 def vol BACKUPPOOL /usr/lpp/adsmserv/bin/bk01x@ acc=READW

   /* Replace /usr/lpp/adsmserv/bin/bk03 FILES 4M in BACKUPPOOLF */
 def vol BACKUPPOOLF /usr/lpp/adsmserv/bin/bk03@ acc=READW

   /* Replace BACK4X COOL8MM 0M in BACKUPPOOLT */
 def vol BACKUPPOOLT BACK4X@ acc=READW

end PRIMARY.VOLUMES.REPLACEMENT macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin STGPOOLS.RESTORE macro

/* Purpose: Restore the primary storage pools from copy storage pool(s). */
/* Recovery Administrator: Delete entries for any primary storage pools  */
/*   that you do not want to restore.                                    */

 restore stgp ARCHIVEPOOL
 restore stgp BACKUPPOOL
 restore stgp BACKUPPOOLF
 restore stgp BACKUPPOOLT
 restore stgp SPACEMGPOOL

end STGPOOLS.RESTORE macro

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

*Figure 72 (Part 9 of 10). Example of a Disaster Recovery Plan File*

```
begin VOLUME.HISTORY.FILE

*****************************************************************************************************
*
*               IBM AdStar Distributed Storage Manager Sequential Volume Usage History
*                                     Updated 08/11/1995 10:20:34
*
*     Operation         Volume   Backup Backup Volume Device          Volume
*     Date/Time          Type    Series Oper.   Seq   Class Name      Name
*****************************************************************************************************
 1995/08/11 10:18:43  STGNEW          0      0      0 COOL8MM         BACK4X
 1995/08/11 10:18:43  STGNEW          0      0      0 FILES           /usr/lpp/adsmserv/bin/bk03
 1995/08/11 10:18:46  STGNEW          0      0      0 LIB8MM          TPBK05
* Location for volume TPBK06 is: 'Ironvault Inc.'
 1995/08/11 10:19:23  BACKUPFULL      1      0      1 LIB8MM          TPBK06
 1995/08/11 10:20:03  STGNEW          0      0      0 LIB8MM          TPBK07
 1995/08/11 10:20:22  BACKUPINCR      1      1      1 LIB8MM          TPBK08

end VOLUME.HISTORY.FILE

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin DEVICE.CONFIGURATION.FILE

DEFINE DEVCLASS COOL8MM DEVTYPE=8MM FORMAT=DRIVE MOUNTLIMIT=1 MOUNTWAIT=60 MOUNTRETENTION=60 PREFIX=ADSM LIBRARY=ITSML
DEFINE DEVCLASS FILES DEVTYPE=FILE MAXCAPACITY=4096K MOUNTLIMIT=2 DIRECTORY=/usr/lpp/adsmserv/bin
DEFINE DEVCLASS FILESSM DEVTYPE=FILE MAXCAPACITY=100K MOUNTLIMIT=2 DIRECTORY=/usr/lpp/adsmserv/bin
DEFINE DEVCLASS LIB8MM DEVTYPE=8MM FORMAT=DRIVE MOUNTLIMIT=1 MOUNTWAIT=60 MOUNTRETENTION=60 PREFIX=ADSM LIBRARY=RLLIB
DEFINE LIBRARY ITSML LIBTYPE=MANUAL

end DEVICE.CONFIGURATION.FILE

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*

begin DSMSERV.OPT.FILE

* Server options file located in /usr/lpp/adsmserv/bin/dsmserv.optx
TCPPort 1509
VOLUMEHISTORY /usr/lpp/adsmserv/bin/volhistory.txtx
DEVCONFIG     /usr/lpp/adsmserv/bin/devconfig.txtx

end DSMSERV.OPT.FILE

*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
```

*Figure 72 (Part 10 of 10). Example of a Disaster Recovery Plan File*

## Example: Routine Operations Using Disaster Recovery Manager

The following scenario demonstrates the use of the PREPARE command and the disaster recovery plan for disaster recovery preparation.

1. ADSM server ADSMSERV contains the backups for the FileRight Company accounts receivable application.

   For *availability management*, FileRight uses ADSM server database mirroring and copy storage pools whose volumes are kept onsite.

   For *disaster recovery management*, FileRight uses ADSM server database backup and copy storage pool volumes which are immediately moved offsite after creation.

   FileRight also uses the AIX **mksysb** function to create a bootable image of the base operating system and ADSM code.

2. To prepare for a possible disaster, the administrator records the following site specific recovery steps in the RECOVERY.INSTRUCTIONS stanza source files:

- Software license numbers
- Sources of replacement hardware
- The FileRight Company's specific recovery steps

3. ADSM storage pool backup and database backup processing is performed nightly using ADSM central scheduling.

4. At 8 a.m. the ADSM tape administrator issues the following command to create a list of volumes to be given to the courier:

```
query drmedia * wherestate=mountable
```

   These volumes were created last night by the ADSM server storage pool backup and database backup processing.

5. To check the volumes out of the library, the administrator issues the following command:

```
move drmedia * wherestate=mountable
```

   This command ejects the volumes from the library and marks them unavailable to ADSM.

6. The administrator then packages the volumes so that they can be given to the courier.

7. The courier arrives to pick up today's backup tapes. The administrator records that the volumes will be given to the courier by issuing the following command:

```
move drmedia * wherestate=notmountable
```

8. To generate a recovery plan file with the latest volume information, the administrator issues the following command:

```
prepare
```

   The administrator copies the recovery plan file to a diskette which will be given to the courier.

9. Several weeks ago during routine ADSM server processing copy storage pool volume CSP01 was reclaimed and its ADSM volume status was changed to PENDING. The volume is physically located at the offsite vault.

10. Last night the PENDING window passed for volume CSP01 and its ADSM volume status is now EMPTY. The volume no longer contains valid backup data and should be brought back onsite for reuse or disposal.

11. To determine if any tapes need to be returned from the vault, the administrator generates a list with the following command:

```
query drmedia * wherestate=vaultretrieve
```

These tapes no longer have valid backup data on them. Volume CSP01 is included in the list.

12. The administrator gives the database backup and copy storage pool tapes, the recovery plan file diskette, and the list of volumes to be returned from the vault to the courier.

13. The courier gives the tapes that were on the previous day's return from the vault list to the administrator. To update the state of these tapes as returned to onsite, the administrator issues the following command:

```
move drmedia * wherestate=courierretrieve
```

14. The courier drives to the vault with today's database backup and copy storage pool tapes, the recovery plan diskette, and the volumes to return from the vault list.

15. At 4 p.m. the ADSM tape administrator calls the vault and verifies that today's backup database and storage pool tapes arrived and are secure. To set the location of these volumes to VAULT, the administrator issues the following command:

```
move drmedia * wherestate=courier
```

16. The vault also tells the administrator that the volumes on today's return from the vault list have been given to the courier. The administrator issues the following command:

```
move drmedia * wherestate=vaultretrieve
```

This command changes the status for this volume to COURIERRETRIEVE.

17. Later that week, an audit team from headquarters arrives unannounced and asks the administrator for a copy of the disaster recovery plan for this server. The administrator gives the auditors a copy of the recovery plan file generated two hours earlier with up-to-date information that includes the volumes required for recovery, their location, and the commands required to use them to restore the server. The auditors are impressed by the plan's timeliness and completeness.

## Storage of Client Recovery Information

DRM allows you to store recovery information for client machines backed up by the ADSM server.

| Task | Required Privilege Class |
|---|---|
| Defining machine information<br>Associating client nodes with machines<br>Defining and tracking machine recovery media<br>Associating recovery media with machines | System |

## Defining Machine Information

Machine information is used to store details about the machine on which a client node resides. In the event of a disaster, this information can help you identify what you need to rebuild or restore the replacement machines.

To assist with the recovery of an ADSM client machine, define the following information in the ADSM database:

- Machine location and business priority
- The ADSM client nodes associated with a machine
- Machine characteristics
- Machine recovery instructions

**Note:** The machine characteristics and machine recovery instructions do not have to be defined during the set up process. You can return to this step later.

1. To store information about the machine that contains the client, issue the DEFINE MACHINE command and specify the client's location and business priority.

   The following example defines machine mach22 in building 021, 2nd floor, in room 2929, and has a priority value of 1:

```
define machine mach22 building=021 floor=2 room=2929 priority=1
```

2. To associate one or more ADSM client nodes with a machine, issue the DEFINE MACHNODEASSOCIATION command.

   During disaster recovery, this association information is used to determine what ADSM client nodes resided on machines that have been destroyed. The file spaces associated with these client nodes should be restored.

The following example associates node CAMPBELL with machine mach22:

```
define machnodeassociation mach22 campbell
```

You can query your machine definitions by issuing the QUERY MACHINE command. For an example, see the query machine output in “Example: Recovering ADSM Clients” on page 399.

3. To insert machine characteristics and recovery instructions into the ADSM database, issue the INSERT MACHINE command. You must insert machine characteristics or recovery instructions line by line; therefore, you may want to create an awk script to do this process for you, see Figure 73 on page 393 for an example.

   The following two examples display how to insert machine characteristics and recovery instructions using a line-by-line method, and using an awk script. You must first use an operating system query command or utility to identify the characteristics for your client machine.

   - **INSERT MACHINE Command from an ADSM Prompt**

     The following shows partial output from a query on an AIX client machine operating system. For our example, we want to save the information from lines 1 and 4 with the INSERT MACHINE command.

     ```
     --1  Host Name: mach22 with 8 MB Memory Card
     ---    16 MB Memory Card
     ---
     --4  Operating System: AIX Version 3 Release 2
     ---
     ---  Hardware Address: 10:00:5x:a8:6a:46
     ```

     – The following example inserts the text “Host Name: mach22 with 8 MB Memory Card” as line 1 and “Operating System: AIX Version 3 Release 2” as line 2 into the ADSM database for machine mach22.

```
insert machine mach22 1 characteristics="Host Name: mach22 with 8 MB Memory Card"
insert machine mach22 2 characteristics="Operating System: AIX Version 3 Release 2"
```

  - To specify recovery instructions for your client machine, use this same command but with the RECOVERYINSTRUCTIONS parameter. Characteristics and recovery instructions cannot be specified on the same command.

```
insert machine mach22 1 -
  recoveryinstructions="Recover this machine for accounts receivable dept."
```

- **INSERT MACHINE Command Using an Awk Script**

  The following is an example procedure to show how you can write a local procedure to insert machine characteristics.

  - The output from the AIX operating system commands *lsdev, lsvg,* and *df* are written to the file clientinfo.txt. These commands will list the devices, logical volumes by volume group, and file systems.

    The file, clientinfo.txt, is then processed by the awk script, which builds an ADSM macro of INSERT commands (one INSERT command for each line in clientinfo.txt).

    The macro is then executed to load the data into the ADSM database.

    From an AIX prompt, the following commands are issued:

```
echo "devices"                           > clientinfo.txt
lsdev -C | sort -d -f                   >> clientinfo.txt
echo "logical volumes by volume group"  >> clientinfo.txt
lsvg -o | lsvg -i -l                    >> clientinfo.txt
echo "file systems"                     >> clientinfo.txt
df                                      >> clientinfo.txt
```

    Figure 73 is an example procedure named *machchar* to insert machine characteristics.

    The machchar.awk.smp script is shipped with the DRM feature and is located in the /usr/lpp/adsmserv/bin directory.

```
# Read machine characteristics from a file and build ADSM macro commands
# to insert the information into the machine characteristics table.
# Invoke with:
#   awk -f machchar.awk -v machine=acctrcv filewithinfo
BEGIN {
      print "delete machine "machine" type=characteri"
      }
      {
      print "insert machine "machine" "NR" characteri=\""$0"\""
      }
END   {
      }
```

*Figure 73. Example of Awk Script File to Insert Machine Characteristics*

The machchar.awk script is then executed from an AIX prompt as follows:

```
awk -f machchar.awk -v machine=acctrcv clientinfo.txt > clientinfo.mac
```

- To insert the machine characteristics, start an administrative client and execute the macro. For example:

```
dsmadmc -id=xxx -pw=xxx macro clientinfo.mac
```

  You can view your machine characteristics by issuing the QUERY MACHINE command with FORMAT=CHARACTERISTICS parameter.

- To specify recovery instructions for your client machine, you can use this same awk script process but with the RECOVERYINSTRUCTIONS parameter.

Your client recovery information is now saved in the ADSM database.

## Defining and Tracking Recovery Media

Use the following commands to save a description of the bootable media required to reinitialize or reinstall an operating system on a client machine, and associate one or more machines with this media. You can also use these commands to associate non-executable media such as application user guides with client machines.

1. Define your boot media needed for recovering one or more machines by issuing the DEFINE RECOVERYMEDIA command. In the following example, the boot recovery media name is tellerwrkstnimage, the volume list includes aix001, aix002, and aix003, for product AIX 4.1. The location of the recovery media is Building 21.

```
define recoverymedia tellerwrkstnimage volumenames=aix001,aix002,aix003
  type=boot product="AIX 4.1" location="Building 21"
```

   This command is usually only needed when a client machine configuration changes. For example, if you install a new level of AIX and create a bootable image with **mksysb**, issue the DEFINE RECOVERYMEDIA command to create a new recovery media definition that can be used to track the new mksysb volumes.

   To query your recovery media definitions, issue the QUERY RECOVERYMEDIA command with the FORMAT=DETAILED parameter.

2. Use the DEFINE RECMEDMACHASSOCIATION command to associate one or more machines with a recovery media. Before you associate a machine with a recovery media, the specified machine must exist and the recovery media must exist.

   During disaster recovery, this association information can be used to determine what boot media to use in the replacement machines.

The following example associates machine MACH255 with recovery media tellerwrkstnimage:

```
define recmedmachassociation tellerwrkstnimage mach255
```

3. When the boot media is moved offsite, update the location with the UPDATE RECOVERYMEDIA command.

   The following example updates the location of boot media tellerwrkstnimage to Ironvault:

```
update recoverymedia tellerwrkstnimage location=ironvault
```

In a *recovery* scenario, you may want to have softcopy manuals that are on a CD-ROM. You can define this to DRM with the DEFINE RECOVERYMEDIA command.

The following example defines the AIX 4.1 manuals, a volume identifier of cd0001, and a type of OTHER because this is a manual:

```
define recoverymedia aix41manuals description="AIX 4.1 Bookshelf" -
  volumes=cd0001 type=other
```

## Recovering the Server

The following list is a guideline to recovering your ADSM server using DRM's disaster recovery plan file.

- Obtain the latest disaster recovery plan file
- Break out the disaster recovery plan file to view, update, print, or execute as ADSM macros or shell scripts
- Obtain the backup volumes from the vault
- Locate a suitable replacement machine
- Restore an AIX image to your replacement machine
- Review the RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE RECOVERY.SCRIPT.NORMAL.MODE shell scripts because they are important for restoring the server to a point where clients can be recovered (see "About the Recovery Plan File Stanzas" on page 369).

### Example: Recovering the ADSM Server

The following scenario demonstrates how an administrator uses the recovery plan file to recover the ADSM server.

1. A disaster is declared for the building that houses the distributed systems server facility at the FileRight Company. Complete recovery of the ADSM server is required.

2. The recovery administrator views the latest recovery plan file for the ADSM server. The recovery plan file stored all the required recovery information in one place. Following the predefined notes in the RECOVERY.INSTRUCTIONS.GENERAL stanza, the administrator reviews the sequence of steps required to recover the server.

3. Step one is to begin the process of obtaining the backup tapes for the server. Fortunately, the backup tapes were stored offsite.

4. The administrator views the RECOVERY.INSTRUCTIONS.OFFSITE stanza for the name and telephone number of the courier the company uses to move tapes between the data center and the offsite vault.

5. The administrator uses a locally written procedure to break out the recovery plan file stanzas into multiple files. (See "About the Disaster Recovery Plan File" on page 369). These files can be optionally viewed, updated, printed, or executed as ADSM macros or shell scripts.

6. The administrator prints out the RECOVERY.VOLUMES.REQUIRED file. The printout is handed to the courier who goes to the offsite vault to obtain the backup volumes.

7. In the meantime, the administrator must find a suitable replacement machine. Stanza RECOVERY.DEVICES.REQUIRED specifies the required tape drive type that will be needed to read the backup tapes. Stanza SERVER.REQUIREMENTS summarizes the required amount of disk space.

8. The administrator restores an AIX image to the replacement machine using a mksysb tape. This tape is created whenever software updates or configuration

changes are made to the AIX system. It includes the ADSM server software. This tape and its location were specified in the RECOVERY.INSTRUCTIONS.INSTALL stanza.

Restoring from the mksysb tapes include:

- Recreating the root volume group.
- Recreating the file system that the database, recovery log, storage pool and disk volumes are located.

9. The administrator reviews the ADSM macros contained in the recovery plan. At the time of the disaster, the courier had not picked up the database and storage pool incremental backup volumes created the previous night. However, they were not destroyed by the water. The administrator removes the entry for the storage pool backup volume from the COPYSTGPOOL.VOLUMES.DESTROYED file.
10. The courier returns with the required volumes. Somehow, the vault could not find one of the copy storage pool volumes. There is not enough time to wait for the vault location to find the lost volume. The administrator removes the entry for that volume from the COPYSTGPOOL.VOLUMES.AVAILABLE file.
11. All of the server's primary volumes were destroyed. The administrator decides there are no changes required to the PRIMARY.VOLUMES shell script and ADSM macro files.
12. To restore the server database to a point where clients can be recovered, the administrator invokes the RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE shell script file by entering the shell script file name at the command prompt.

    **Note:** Alternatively the administrator could have used the steps in the recovery script as a guide, and manually executed each step.

    The following are the steps executed in this recovery shell script:

    a. Copy the ADSM server options file from the DSMSERV.OPT file to its original location.

    b. Copy the volume history file required by ADSM DSMSERV RESTORE DB processing from the VOLUME.HISTORY.FILE file to its original location.

       **Note:** Use this copy of the volume history file unless you have a more recent copy (after the disaster occurred).

    c. Copy the device configuration file required by ADSM DSMSERV RESTORE DB processing from the DEVICE.CONFIGURATION.FILE file to its original location.

    d. Create the ADSM server recovery log and database volumes using DSMFMT.

    e. Issue DSMSERV INSTALL for the recovery log and database files.

    f. Issue the DSMSERV RESTORE DB command.

    g. Start the server in the background.
13. The administrator invokes the RECOVERY.SCRIPT.NORMAL.MODE shell script file to restore the server primary storage pools.

**Note:** This action is optional at this time because ADSM can access the copy storage pool volumes directly to restore client data. Using this feature, the administrator can minimize client recovery time because server primary storage pools do not have to be restored first. However, in this scenario, the client machines were not damaged, so the focus of the administrator is to restore full ADSM server operation.

If client machines are damaged, you may want to delay this action until after all clients are recovered.

Alternatively, the administrator could have used the steps in the recovery script as a guide, and manually executed each step.

The steps executed in this recovery shell script are as follows:

a. Create replacement primary volumes.

b. Define the replacement primary volumes to ADSM.

c. Restore the primary storage pools.

14. The administrator collects the database backup and copy storage pool volumes used in the recovery so that they can be returned to the vault. For these backup volumes to be returned to the vault using the routine MOVE DRMEDIA process, the administrator executes the following ADSM administrative commands:

    ```
    update volhist TPBK50 devcl=lib8mm ormstate=mountable
    update volhist TPBK51 devcl=lib8mm ormstate=mountable
    ```

    The copy storage pool volumes used in the recovery already have the correct ORMSTATE.

15. The administrator then runs the BACKUP DB command to back up the newly restored database.

16. The administrator issues the MOVE DRMEDIA * WHERESTATE=MOUNTABLE command to check the volumes out of the library.

17. To create a list of the volumes to be given to the courier, the administrator issues the QUERY DRMEDIA * WHERESTATE=NOTMOUNTABLE.

18. After the administrator packages the volumes and gives them to the courier, the MOVE DRMEDIA * WHERESTATE=NOTMOUNTABLE command is issued.

19. The administrator issues PREPARE and celebrates the successful disaster recovery.

## Recovering the Clients

To recover ADSM clients, you must have the following information:

- Client machines that have been defined to ADSM, along with their location and restore priority value.
- The boot recovery media.
- Specific recovery instructions for the machine.
- Hardware requirements for the machine.

## Example: Recovering ADSM Clients

The following scenario demonstrates how to use DRM's query commands to guide an administrator through the recovery of ADSM clients.

1. A week after the ADSM server was recovered, another water pipe burst in the building that houses distributed systems applications. Many machines that were backed up using ADSM clients are destroyed. A disaster is declared.

2. To view a list of client machines that were lost in building 21 and their restore priority, the administrator issues the following command:

```
query machine building=021 format=detailed
```

ADSM displays information similar to the following:

```
         Machine Name: POLARIS
     Machine Priority: 1
             Building: 21
                Floor: 2
                 Room: 1
         ADSM Server?: No
          Description: Payroll
            Node Name: POLARIS
  Recovery Media Name: MKSYSB1
     Characteristics?: Yes
Recovery Instructions?: Yes
```

3. For *each* machine, the administrator issues the following commands:

   a. To determine the location of the boot media, the administrator issues the QUERY RECOVERYMEDIA command. For example:

```
query recoverymedia mksysb1
```

   ADSM displays the following information:

```
Recovery Media Name  Volume Names    Location    Machine Name
-------------------- -----------     ----------  ----------------
MKSYSB1              vol1 vol2       IRONVAULT    POLARIS
                      vol3
```

   b. To determine the machine specific recovery instructions for the POLARIS machine, the administrator issues:

```
query machine polaris format=recoveryinstructions
```

   ADSM displays the following:

```
Recovery Instructions for Polaris.
Primary Contact:
   Jane Smith (wk 520-000-0000 hm 520-001-0001)
Secondary Contact:
   John Adams (wk 520-000-0001 hm 520-002-0002)
```

   c. To determine the machine hardware requirements for POLARIS, the administrator issues:

```
query machine polaris format=characteristics
```

ADSM displays information similar to the following:

```
devices
aio0         Defined                Asynchronous I/O
bus0         Available 00-00        Microchannel Bus
fd0          Available 00-00-0D-00  Diskette Drive
fda0         Available 00-00-0D     Standard I/O Diskette Adapter
fpa0         Available 00-00        Floating Point Processor
gda0         Available 00-04        Color Graphics Display Adapter
hd1          Defined                Logical volume
hd2          Defined                Logical volume
hd3          Defined                Logical volume
hdisk0       Available 00-01-00-00  400 MB SCSI Disk Drive
hdisk1       Available 00-01-00-40  Other SCSI Disk Drive
hft0         Available              High Function Terminal Subsystem
inet0        Available              Internet Network Extension
ioplanar0    Available 00-00        I/O Planar
kbd0         Defined   00-00-0K-00  United States keyboard
lb0          Available 00-02-00-20  ADSM Library
lo0          Available              Loopback Network Interface
loglv00      Defined                Logical volume
lp0          Available 00-00-0P-00  IBM 4201 Model 3 Proprinter III
lv03         Defined                Logical volume
lv04         Defined                Logical volume
lvdd         Available              N/A
mem0         Available 00-0B        8 MB Memory Card
mem1         Available 00-0C        16 MB Memory Card
mous0        Defined   00-00-0M-00  3 button mouse
mt0          Available 00-02-00-40  ADSM Tape Drive
ppa0         Available 00-00-0P     Standard I/O Parallel Port Adapter
pty0         Available              Asynchronous Pseudo-Terminal
rootvg       Defined                Volume group
sa0          Available 00-00-S1     Standard I/O Serial Port 1
sa1          Available 00-00-S2     Standard I/O Serial Port 2
scsi0        Available 00-01        SCSI I/O Controller
scsi1        Available 00-02        SCSI I/O Controller
sio0         Available 00-00        Standard I/O Planar
siokb0       Available 00-00-0K     Keyboard Adapter
sioms0       Available 00-00-0M     Mouse Adapter
siotb0       Available 00-00-0T     Tablet Adapter
sys0         Available 00-00        System Object
sysplanar0   Available 00-00        CPU Planar
sysunit0     Available 00-00        System Unit
tok0         Available 00-03        Token-Ring High-Performance Adapter
tr0          Available              Token Ring Network Interface
tty0         Available 00-00-S1-00  Asynchronous Terminal
tty1         Available 00-00-S2-00  Asynchronous Terminal
usrvice      Defined                Logical volume
veggie2      Defined                Volume group
logical volumes by volume group
veggie2:
LV NAME             TYPE       LPs   PPs  PVs  LV STATE      MOUNT POINT
hd2                 jfs        103   103  1    open/syncd    /usr
hd1                 jfs        1     1    1    open/syncd    /home
hd3                 jfs        3     3    1    open/syncd    /tmp
hd9var              jfs        1     1    1    open/syncd    /var
file systems
Filesystem    Total KB    free %used   iused %iused Mounted on
/dev/hd4          8192     420   94%     909    44% /
/dev/hd9var       4096    2972   27%      87     8% /var
/dev/hd2        421888   10964   97%   17435    16% /usr
/dev/hd3         12288   11588    5%      49     1% /tmp
/dev/hd1          4096    3896    4%      26     2% /home
```

d. With the necessary recovery information now available, the administrator successfully restores each client machine.

## Customizing Disaster Recovery Manager

You can customize DRM with SET commands to specify the management of the following:

- Storage pools
- Path name prefixes where the recovery plan instructions and disaster recovery plan files should reside
- Replacement volume identifier
- Offsite recovery media

You can also customize site specific recovery instructions. The site specific recovery instructions are flat files that are manually edited using predetermined file names (for example, RECOVERY.INSTRUCTIONS.GENERAL). The site specific recovery instructions are used by the PREPARE command when a new disaster recovery plan is generated.

| Task | Required Privilege Class |
|---|---|
| Specify copy storage pools to be managed<br>Specify primary storage pools to be managed<br>Specify the character ID for replacement volume names<br>Specify the prefix portion of the path name for recovery plan instructions<br>Specify the prefix portion of the path name for recovery plan files | System |

## Customizing the Management of Storage Pools, Path Name Prefixes, and Replacement Volume Identifiers

This section describes the SET commands to configure DRM. For more information, refer to *ADSM Administrator's Reference*.

### Copy Storage Pools

Issue the SET DRMCOPYSTGPOOL command to specify the copy storage pools to be managed by DRM. Specify the name of the copy storage pools used for backing up the primary storage pools defined with the SET DRMPRIMSTGPOOL command. These copy storage pools will be processed by the PREPARE, MOVE DRMEDIA, and QUERY DRMEDIA commands. You can specify a list of copy storage names or a null string (“”) to indicate that all copy storage pools defined to the server are eligible for processing. At installation, all primary storage pools defined to the server are eligible for processing.

Copy storage pools that you may not want DRM to manage can include onsite copy storage pools used for recovery from media failures.

The following example specifies that copy storage pools with the pattern-matching expression of COPY are to be managed by DRM:

```
set drmcopystgpool copy*
```

You can override this SET command using the COPYSTGPOOL parameter with the PREPARE command, for more information refer to *ADSM Administrator’s Reference*.

## Primary Storage Pools

You can specify the primary storage pools that you want to restore. Eligible primary storage pool volumes defined to these storage pools are included in the plan file stanzas generated by the PREPARE command.

Use the SET DRMPRIMSTGPOOL command to specify which primary storage pools should be processed by the PREPARE command. You can specify a list of primary storage pool names or a null string (“”) to indicate that all primary storage pools defined to the server are eligible for processing. At installation, all the primary storage pools defined to the server are eligible for processing.

The following example specifies that primary storage pools PRIM1 and PRIM2 are to be managed by DRM:

```
set drmprimstgpool prim1,prim2
```

You can override this setting using the PRIMSTGPOOL parameter with the PREPARE command, for more information refer to *ADSM Administrator’s Reference*.

## Character Identification for Replacement Volume Names

Issue the SET DRMPLANVPOSTFIX command to specify one character to be added to the end of the replacement volumes names in the disaster recovery plan. At installation, the character is set to @. After installation, use this command to change the character.

This command can be used to make the replacement primary storage pool volumes easy to find in the recovery plan stanzas, or to automatically generate replacement volume names.

The following example defines the character identification as r:

```
set drmplanvpostfix r
```

## Prefix for Recovery Instructions

Issue the SET DRMINSTRPREFIX command to specify the prefix portion of the path name for the recovery instructions source files.

The following example specifies the prefix as /u/recovery/plans/rpp:

```
set drminstrprefix /u/recovery/plans/rpp
```

PREPARE processing will include the information from the following files in the disaster recovery plan file:

```
/u/recovery/plans/rpp.RECOVERY.INSTRUCTIONS.GENERAL
/u/recovery/plans/rpp.RECOVERY.INSTRUCTIONS.OFFSITE
/u/recovery/plans/rpp.RECOVERY.INSTRUCTIONS.INSTALL
/u/recovery/plans/rpp.RECOVERY.INSTRUCTIONS.DATABASE
/u/recovery/plans/rpp.RECOVERY.INSTRUCTIONS.STGPOOL
```

You can override this SET command using the INSTRPREFIX parameter with the PREPARE command, for more information refer to *ADSM Administrator's Reference*.

## Prefix for Recovery Plan File

Issue the SET DRMPLANPREFIX command to specify the prefix portion of the path name for the generated recovery plan file.

This prefix is used by ADSM to identify the location of the recovery plan file. The plan prefix is also used to generate the ADSM macros and shell script file names included in the RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE and RECOVERY.SCRIPT.NORMAL.MODE.

The following example specifies the prefix as /u/server/recoveryplans/:

```
set drmplanprefix /u/server/recoveryplans/
```

The disaster recovery plan file name created by PREPARE processing will be in the following format:

```
/u/server/recoveryplans/19950603.013030
```

You can override this SET command using the PLANPREFIX parameter with the PREPARE command, for more information refer to *ADSM Administrator's Reference*.

## Customizing the Management of Offsite Recovery Media

This section describes the SET commands to configure DRM with information necessary for offsite recovery media management. For more information, refer to *ADSM Administrator’s Reference*.

| Task | Required Privilege Class |
|---|---|
| Specify the copy storage pools to be managed | System |
| Specify the name of the courier | |
| Specify if ADSM should read sequential media labels of volumes checked out with MOVE DRMEDIA | |
| Specify expiration of a database backup series | |
| Specify processing of backup volumes | |
| Specify the name of the vault where volumes are stored | |

### Copy Storage Pools

To specify the copy storage pools to be managed, see “Copy Storage Pools” on page 402.

### Courier Name

Issue the SET DRMCOURIERNAME command to specify the courier name. At installation, the name of the courier is set to COURIER. After installation, this command can be used to modify the name of the courier. The courier name is used by the MOVE DRMEDIA command to set the location of volumes that are changing from the NOTMOUNTABLE state to the COURIER state.

The following example specifies the courier name as Joe’s Courier Service:

```
set drmcouriername "Joe’s Courier Service"
```

### Sequential Media Labels for Checked Out Volumes

Issue the SET DRMCHECKLABEL command to specify whether ADSM should read sequential media labels of volumes checked out with the MOVE DRMEDIA command. At installation, the value is set to YES. After installation, use this command to modify the value.

The following example specifies that DRM should not read the volume labels:

```
set drmchecklabel no
```

## Expiration of a Database Series

Issue the SET DRMDBBACKUPEXPIREDAYS command to specify the number of days before a database backup series is expired. At installation, the number of days before expiration is set to 60. After installation, use this command to modify the number of days that must elapse before a database is expired. A volume is considered eligible for expiration if all of the following conditions are true:

- The last volume of the series exceeds the expiration value specified with SET DRMDBBACKUPEXPIREDAYS. The expiration value specifies the number of days that must elapse since the volume was used by database backup.
- The volume's state is VAULT.
- The volume is not part of the most recent series (DRM will not expire the most recent database backup series).

The following example specifies that 30 days should pass before a database is expired:

```
set drmdbbackupexpiredays 30
```

## Processing of Backup Volumes

Issue the SET DRMFILEPROCESS command to specify whether the MOVE DRMEDIA or QUERY DRMEDIA commands should process database backup volumes and copy storage pool volumes that are associated with a device class with a DEVTYPE=FILE. This command is useful for testing of the DRM environment. At installation, the value is set to No. After installation, you can modify this value.

```
set drmfileprocess yes
```

## Specify the Vault Name

Issue the SET DRMVAULTNAME command to specify the vault name where volumes are stored. At installation, the name of the vault is set to VAULT. After installation, you can modify this value.

The vault name is used by the MOVE DRMEDIA command to set the location of volumes that are transitioning from the COURIER state to the VAULT state.

The following example specifies the vault name as Ironvault with a contact name of D. Lastname, at telephone number 1-000-000-0000:

```
set drmvaultname "Ironvault, D. Lastname, 1-000-000-0000"
```

## Querying the Disaster Recovery Manager System Parameters

To query the settings defined for DRM, issue the QUERY DRMSTATUS command. For example:

```
query drmstatus
```

ADSM displays information similar to the following:

```
           Recovery Plan Prefix: /u/recovery/plans/rpp
        Plan Instructions Prefix: /u/recovery/plans/source/
      Replacement Volume Postfix: @
           Primary Storage Pools: PRIM1 PRIM2
              Copy Storage Pools: COPY*
                    Courier Name: Joe’s Courier Service
                 Vault Site Name: Ironvault, D. Lastname, 1-000-000-0000
DB Backup Series Expiration Days: 30 Day(s)
                     Check Label: Yes
                   Process Files: No
```

## Customizing the Site Specific RECOVERY.INSTRUCTIONS

The PREPARE command includes site specific recovery instructions as stanzas in the disaster recovery plan.

Using the following file names, you can create and edit files with specific recovery instructions for your site:

- instructionsprefixRECOVERY.INSTRUCTIONS.GENERAL
- instructionsprefixRECOVERY.INSTRUCTIONS.OFFSITE
- instructionsprefixRECOVERY.INSTRUCTIONS.INSTALL
- instructionsprefixRECOVERY.INSTRUCTIONS.DATABASE
- instructionsprefixRECOVERY.INSTRUCTIONS.STGPOOL

When you create and edit these files and then issue the PREPARE command, the information in these files is included in the disaster recovery plan as stanzas.

The following examples show sample entries for these files.

**instructionsprefixRECOVERY.INSTRUCTIONS.GENERAL**

Include information such as administrator names, telephone numbers, location of passwords, and so on.

The following is example text for this file:

```
Recovery Instructions for ADSM Server ACMESRV on system ZEUS.
Joe Smith  (wk 002-000-1111 hm 002-003-0000)is the primary system programmer.
Salley Doe (wk 002-000-1112 hm 002-005-0000) is primary recovery administrator.
Jane Smith (wk 002-000-1113 hm 002-004-0000) is the responsible manager.
Security Considerations:
 Joe Smith has the password for the Admin ID ACMEADM.  If Joe is unavailable,
 you will need to either issue SET AUTHENTICATION OFF or define a new
 administrative user ID at the replacement ADSM server console.
```

**instructionsprefixRECOVERY.INSTRUCTIONS.OFFSITE**

Include information such as the offsite vault location, courier's name, and telephone numbers.

The following is example text for this file:

```
Our offsite vault location is Ironvault, Safetown, Az.
The phone number is 1-800-000-0008. You need to contact them directly
to authorize release of the tapes to the courier.
Our courier's name is Fred Harvey.  You can contact him at 1-800-444-0000.
Since our vault is so far away, be sure to give the courier a list
of both the database backup and copy storage pool volumes required. Fred
is committed to returning these volumes to us in less than 12 hours.
```

**instructionsprefixRECOVERY.INSTRUCTIONS.INSTALL**

Include information about how to restore the base server system from boot media or if boot media is unavailable, how to install the ADSM server and where the installation volumes and license number are located.

The disaster recovery plan file issues commands using the ADSM administrative client, for example, dsmadmc. Ensure the proper path to the administrative client is established prior to executing the scripts RECOVERY.SCRIPT.DISASTER.RECOVERY.MODE and RECOVERY.SCRIPT.NORMAL.MODE. For example, set the shell variable PATH or update the scripts with the appropriate path specification to find the ADSM administrative client.

The following is example text for this file:

```
Most likely you will not need to reinstall the ADSM server and
administrative clients because we use
mksysb to backup the rootvg volume group, and the ADSM server code and
configuration files exist in this group.
However, if you cannot do a mksysb restore of the base server system,
and instead have to start with a fresh AIX build, you may need
to add ADSM server code to that AIX system.
The install volume for the ADSM server is INS001. If that is lost, you
will need to contact Copy4You Software, at 1-800-000-0000, and obtain
a new copy. Another possibility is the local IBM Branch office at 555-7777.
Our license number is 0000000000.0
```

**instructionsprefixRECOVERY.INSTRUCTIONS.DATABASE**
Include information about how to recover the database along with how much hardware space is needed.

The following is example text for this file:

```
You will need to find replacement disk space for the server database.  We have
an agreement with Joe Replace that in the event of a disaster, he will
provide us with disk space.
```

**instructionsprefixRECOVERY.INSTRUCTIONS.STGPOOL**
Include information on primary storage pool recovery instructions.

The following is example text for this file:

```
Do not worry about the archive storage pools during this disaster recovery.
Focus on migration and backup storage pools.
The most important storage pool is XYZZZZ.
```

## Using an Awk Script to Break Out a Disaster Recovery Plan File

*Awk* is an AIX utility that can process a text file.  In the case of DRM, this text file is the disaster recovery plan file.

If you want to restore the ADSM server, you can use an awk script procedure or an editor to break out the stanzas in the disaster recovery plan file into individual files as appropriate.

An example procedure, *planexpl.awk.smp*, is shipped with the DRM feature and shows how you can write a local procedure to process your disaster recovery plan file.

## Summarized Example of Disaster Recovery Manager Usage

This section is an example outline to show how you use DRM in normal routine processing and during a disaster recovery procedure.

*Setup*

1. Enable DRM by registering the license
2. Ensure the device configuration and volume history information files exist
3. Backup your storage pools and database
4. Define site specific ADSM server recovery instructions
5. Describe priority ADSM client machines

*Daily Operations*

Day 1

- Back up client files
- Backup ADSM server storage pools
- Backup ADSM server database (full backup)
- Determine what backup volumes have been created
- Eject the volumes from the library
- Hand the volumes to the courier
- Generate the disaster recovery plan with PREPARE command

Day 2

- Back up client files
- Back up ADSM server storage pools
- Back up ADSM server database (full backup)
- Move the new backup volumes offsite
- Acknowledge receipt of previously sent volumes at vault (from Day 1)
- Generate the disaster recovery plan with PREPARE command

Day 3

- Automatic storage pool reclamation processing occurs
- Back up ADSM server database (incremental)
- Move the new backup volumes offsite
- Acknowledge receipt of previously sent volumes at vault (from Day 2)
- Give courier a list of empty volumes to be returned from the vault.
- Generate the disaster recovery plan with PREPARE command

*Disaster Occurs*

Day 4

- The ADSM server machine and the client machines have been destroyed in the disaster.

*Disaster Recovery*

Day 4 (continued)

1. Restore ADSM server using the latest recovery plan
2. Identify the top priority client node in the building where disaster occurred
3. Restore client machine files from ADSM server copy storage pools
4. Restore ADSM server primary storage pools
5. Move database backup and copy storage pool volumes back to the vault

*Daily Operations*

Day 5

- Back up client files
- Backup ADSM server storage pools
- Backup ADSM server database (full backup)
- Determine what backup volumes have been created
- Eject the volumes from the library
- Hand the volumes to the courier
- Generate the disaster recovery plan with PREPARE command

# Appendix A. External Media Management Interface Description

This appendix contains General-use Programming Interface and Associated Guidance Information about the interface that ADSM provides to external media management programs. To use the interface, you must first define an EXTERNAL library. For information on this library type, see “External Libraries” on page 206.

The interface consists of request description strings that ADSM sends and response strings that the external program sends.

The details of the request types and the required processing are described in the sections that follow. The request types are:

- Initialization of the external program
- Volume mount
- Volume dismount
- Volume release (return to scratch)

## Processing during ADSM Server Initialization

Ensure that the external media management program cooperates with the ADSM server during the server’s initialization. For each external library defined to the ADSM server, the following must occur during server initialization:

1. The ADSM server loads the external program in a newly created process.
2. The server sends an initialization request description string, in text form, into the standard input (**stdin**) stream of the external program. The server waits for the response.
3. When the external process completes the request, the process must write an initialization response string, in text form, into its standard output (**stdout**) stream.
4. The external program must end by calling the stdlib **exit** routine.

## Processing for Volume Mount, Dismount, and Release Requests

Ensure that the external media management program responds to ADSM server requests, other than initialization, according to the following process:

1. If the request is for a volume mount or release, the external program must be initialized:
    a. The server loads the external program in a newly created process.
    b. The server sends an initialization request description string, in text form, into the standard input (**stdin**) stream of the external program. The ADSM server waits for the response.
    c. When the external process completes the request, the process must write an initialization response string, in text form, into its standard output (**stdout**) stream.

2. The server sends the request description string, in text form, into the standard input (**stdin**) stream of the external program. The ADSM server waits for the response.
3. When the external process completes the request, the process must write a response string, in text form, into its standard output (**stdout**) stream.
4. If the request was for a volume mount, the external program must remain active and ready to accept the volume dismount request from the ADSM server. Otherwise, the program must end by using the stdlib **exit** routine.

## Initialization Requests

When the ADSM server is started, the server sends an initialization request to the external media management program for each EXTERNAL library. The external program must process this request to ensure that the external program is present, functional, and ready to process ADSM requests. If the initialization request is successful, ADSM informs its operators that the external program reported its readiness for ADSM operations. Otherwise, ADSM reports a failure to its operators.

ADSM does not attempt any other type of operation with that library until an initialization request has succeeded. For a mount or release operation, the ADSM server sends an initialization request first. If the initialization is successful, the request is sent. If the initialization is not successful, the request fails. The external media management program can detect whether the initialization request is being sent by itself or with another request by detecting end-of-file on the **stdin** stream. When end-of-file is detected, the external program must end by using the stdlib **exit** routine (not the **return** call).

When a valid response is sent by the external program, the external program must end by using the **exit** routine.

**Format of the ADSM request:**

```
INITIALIZE libraryname
```

where *libraryname* is the name of the EXTERNAL library as defined to ADSM.

**Format of the external program response:**

```
INITIALIZE libraryname COMPLETE, RESULT=resultcode
```

where:

*libraryname*
Specifies the name of the EXTERNAL library as defined to ADSM.

*resultcode*
One of the following:

- SUCCESS
- NOT_READY
- INTERNAL_ERROR

## Volume Mount Requests

When the ADSM server requires a volume mount, the server starts the external media management program, issues a request to initialize, then issues a request to mount a volume. The external program is responsible for verifying that this request is coming from ADSM and not from an unauthorized system.

The volume mounted by the external media management program must be a tape with a standard IBM label. When the external program completes the mount request, the program must send a response. If the mount was successful, the external program must remain active. If the mount failed, the external program must end immediately by using the **exit** routine.

**Format of the ADSM request:**

MOUNT *libraryname volname accessmode devicetypes timelimit userid volumenumber*

where:

*libraryname*
: Specifies the name of the EXTERNAL library as defined to ADSM.

*volname*
: Either SCRTCH if the request is for a scratch mount, or the actual volume name if the request is for an existing volume.

*accessmode*
: Specifies the access mode required for the volume. Possible values are READONLY and READWRITE.

*devicetypes*
: Specifies a list of possible device types and formats that can be used to satisfy the request for the volume. The most preferred device type and format is first in the list. Items are separated by commas, with no intervening spaces. Possible values are:

- 3480
- 3480XF
- 3490E
- 3590
- 4MM_DDS1
- 4MM_DDS1C
- 4MM_DDS2
- 4MM_DDS2C
- 8MM_8200
- 8MM_8205
- 8MM_8500
- 8MM_8500C
- DLT_2000
- DLT_4000
- OPT_RW_1300MB
- OPT_RW_650MB

- OPT_WORM_1300MB
- OPT_WORM_650MB
- QIC_IBM1000
- QIC_525

*timelimit*
Specifies the maximum number of minutes that the server waits for the volume to be mounted. If the mount request is not completed within this time, the external manager responds with the result code TIMED_OUT.

*userid*
Specifies the user ID of the process that needs access to the drive.

*volumenumber*
For non-optical media, the *volumenumber* is 1. For optical media, the *volumenumber* is 1 for side A, 2 for side B.

**Format of the external program response:**

```
MOUNT libraryname volname COMPLETE ON specialfile, RESULT=resultcode
```

where:

*libraryname*
Specifies the name of the EXTERNAL library as defined to ADSM.

*volname*
Specifies the name of the volume mounted for the request.

*specialfile*
The fully qualified path name of the device special file for the drive in which the volume was mounted. If the mount request fails, the value should be set to /dev/null.

The external program must ensure that the special file is closed before the response is returned to the ADSM server.

*resultcode*
One of the following:

- SUCCESS
- DRIVE_ERROR
- LIBRARY_ERROR
- VOLUME_UNKNOWN
- VOLUME_UNAVAILABLE
- CANCELLED
- TIMED_OUT
- INTERNAL_ERROR

## Volume Dismount Requests

When a successful mount operation completes, the external process must wait for a request to dismount the volume. When the dismount operation completes, the external program must send a response to the ADSM server.

After the dismount response is sent, the external process ends immediately by using the **exit** routine.

**Format of the ADSM request:**

```
DISMOUNT libraryname volname
```

where:

*libraryname*
Specifies the name of the EXTERNAL library as defined to ADSM.

*volname*
Specifies the name of the volume to be dismounted.

**Format of the external program response:**

```
DISMOUNT libraryname volname COMPLETE, RESULT=resultcode
```

where:

*libraryname*
Specifies the name of the EXTERNAL library as defined to ADSM.

*volname*
Specifies the name of the volume dismounted.

*resultcode*
One of the following:

- SUCCESS
- DRIVE_ERROR
- LIBRARY_ERROR
- INTERNAL_ERROR

## Volume Release Requests

When the ADSM server returns a volume to scratch status, the server starts the external media management program, issues a request to initialize, then issues a request to release a volume.

The external program must send a response to the release request. No matter what response is received from the external program, ADSM returns the volume to scratch. For this reason, ADSM and the external program can have conflicting information on which volumes are scratch. If an error occurs, the external program logs the failure so that the external library inventory can be synchronized later with ADSM. The synchronization can be a manual operation.

**Format of the ADSM request:**

```
RELEASE libraryname volname
```

where:

*libraryname*
: Specifies the name of the EXTERNAL library as defined to ADSM.

*volname*
: Specifies the name of the volume to be returned to scratch (released).

**Format of the external program response:**

```
RELEASE libraryname volname COMPLETE, RESULT=resultcode
```

where:

*libraryname*
: Specifies the name of the EXTERNAL library as defined to ADSM.

*volname*
: Specifies the name of the volume returned to scratch (released).

*resultcode*
: One of the following:

- SUCCESS
- VOLUME_UNKNOWN
- VOLUME_UNAVAILABLE
- INTERNAL_ERROR

# 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 data storage.

**authentication**. The process of checking a user's password before allowing that user access to the server. Authentication can be turned on or off by an administrator with system privilege.

**autochanger**. A small multislot tape device that has a mechanism that automatically puts tape cartridges into the tape drive or drives. Also called *medium* or *media changer*, or a *library*.

| **availability management**. Managing recovery from relatively common computer system outages such as a disk drive head crash. Recovery is often accomplished by using disk mirroring and other forms of RAID technology, or by maintaining onsite backup copies of data.

**available space**. The amount of space, in megabytes, that is available to the database and recovery log. This space can be used to extend the capacity of the database or recovery log, or to provide sufficient free space before a volume is deleted from the database or recovery log.

**awk**. In AIX, a pattern-matching program for processing text files. With the DRM feature, you can use an awk script to break up the disaster recovery plan file into usable parts.

# B

**background process**. A server process that runs in the background, allowing the administrative client to be used for other work.

**backup**. The process of copying information for safekeeping. ADSM has processes for backing up user files, the database, and storage pools. For example, users can back up one or more files to a storage pool to ensure against loss of data. Contrast with *restore*. See also *database backup series* and *incremental backup*.

**backup-archive client**. A program that runs on a workstation or file server and provides a means for users to back up, archive, restore, and retrieve files. Contrast with *administrative client*.

**backup copy**. A user file that has been backed up to an ADSM storage pool.

**backup copy group**. A policy object containing attributes that control the generation, destination, and expiration of backup files. A backup copy group belongs to a management class.

**BACKUPPOOL**. A disk storage pool defined by ADSM at installation. It can be the destination for client files that are backed up to the server. See *storage pool*.

**backup retention grace period**. The number of days ADSM retains a backup version after the server is unable to rebind the file to an appropriate management class.

**backup series**. See *database backup series*.

**backup version**. A file, directory, or file space that a user has backed up, which resides in ADSM's data storage. There may be more than one backup version of a file in the storage pool, but at most only one is an active backup version. See *active version* and *inactive version*.

**binding**. The process of associating a file with a management class name. See *rebinding*.

**boot media**. Media that contains operating system and other files essential to running a workstation or server.

**buffer**. Storage used to compensate for differences in the data rate flow, when transferring data from one device to another.

**buffer pool**. Temporary space used by the server to hold database or recovery log pages. See *database buffer pool* and *recovery log buffer pool*.

**buffer pool size**. The size of an area in memory used to store database or recovery log pages.

**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 data 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 files that are migrated via ADSM space management. 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. 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 file will be backed up, archived, or migrated. At installation, ADSM provides storage destinations named BACKUPPOOL, ARCHIVEPOOL, and SPACEMGPOOL.

**device class**. A named group of storage devices. Each device class has a unique name and represents a device type of disk, file, optical disk, or tape.

**device configuration file**. A file that contains information about defined device classes, and, on 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.

**disaster recovery**. Recovery from catastrophic interruptions of computer systems, such as loss of the system location because of natural events. Backup data is kept offsite to protect against such catastrophes.

**Disaster Recovery Manager (DRM)**. An ADSM feature that assists in preparing and later using a disaster recovery plan for the ADSM server.

**disaster recovery plan**. A document that contains information about how to recover computer systems if a disaster occurs. With DRM, the plan is a file that contains information about the software and hardware used by the ADSM server, and the location of recovery media.

**DISK**. A device class that is defined by ADSM at installation. It is used to categorize disk drives, such as 3390 DASD or 3380 DASD.

**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.

**DRM**. Disaster Recovery Manager.

**dsm.opt file**. See *client user options file*.

**dsmserv.opt**. See *server options file*.

**dsm.sys file**. See *client system options file*.

**dynamic**. A copy group serialization value that specifies that ADSM accepts the first attempt to back up or archive a file regardless of whether the file is modified during the backup or archive process. See also *serialization*. Contrast with *shared dynamic*, *shared static*, and *static*.

**dynamic link library**. A file containing executable code and data bound to a program at load time or run time, rather than during linking. The code and data in a dynamic link library can be shared by several applications simultaneously.

# E

**ECCST**. Enhanced capacity cartridge system tape.

**enhanced capacity cartridge system tape (ECCST)**. Cartridge system tape with increased capacity that can only be used with 3490E tape subsystems. Contrast with *cartridge system tape*.

**error log**. A character file written on random access media that contains information about errors detected by the server or client.

**estimated capacity**. The available space, in megabytes, of a storage pool.

**Ethernet**. A data link protocol and LAN that interconnects personal computers and workstations via coaxial cable.

**event**. Administrative commands or client operations that are scheduled to be executed at a particular time.

**event record**. A database record that describes actual status and results for events.

**exclude**. The process of identifying files or directories in an include-exclude list to prevent these objects from being backed up whenever a user or schedule issues an incremental or selective backup operation, or to prevent these objects from being migrated off the client node via ADSM space management.

**exclude-include list**. See *include-exclude list*.

**exit**. To execute an instruction within a portion of a computer program in order to terminate the execution of that portion.

**exit machine**. On a VM server, a virtual machine that runs the mount and deletion installation-wide exits on VM systems.

**expiration**. The process by which files are identified for deletion because their expiration date or retention period has passed. Backed up or archived files are marked expired by ADSM based on the criteria defined in the backup or archive copy group.

**expiration date**. On MVS, VM, and VSE servers, a device class attribute used to notify tape management systems of the date when ADSM no longer needs a tape volume. The date is placed in the tape label so that the tape management system does not overwrite the information on the tape volume before the expiration date.

**export**. The process of copying administrator definitions, client node definitions, policy definitions, server control information or file data to external media.

**export/import facility**. See *import/export facility*.

**extend**. The process of increasing the portion of available space that can be used to store database or recovery log information. Contrast with *reduce*.

# F

**file data**. File space definitions, authorization rules, backed up files, and archive copies. 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's data storage. ADSM does not necessarily store all the files from a single file space together, but can identify all the files in its data 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 data storage.

**incremental backup**. (1) A function that allows users to back up files or directories from a client domain that are not excluded in the include-exclude list and that meet the requirements for frequency, mode, and serialization as defined in the backup copy group of the management class assigned to the files. Contrast with *selective backup*. (2) An ADSM function that copies only the pages in the database that are new or changed since the last full or incremental backup. Contrast with *full backup*. See *database backup series*.

**internal mounting facility**. On a VM server, a VM facility that allows the server to request tape mounts by sending a message to a mount operator. The message is repeated until the tape is mounted or until the mount wait time is exceeded.

**inter-user communication vehicle (IUCV) facility**. On a VM server, a VM communication method used to pass data between virtual machines and VM components.

**IPX/SPX**. Internetwork Packet Exchange/Sequenced Packet Exchange. IPX/SPX is Novell NetWare's communication protocol.

**IUCV**. Inter-user communication vehicle.

# K

**KB**. Kilobyte.

**kernel**. The part of an operating system that performs basic functions such as allocating hardware resources.

**kernel extension**. A program that modifies parts of the kernel that can be customized to provide additional services and calls. See *kernel*.

**kilobyte (KB)**. 1024 bytes.

# L

**LAN**. Local area network.

**length**. A device class attribute that specifies the length of cartridge tape by specifying one of the following media types: CST for standard length tape or ECCST for double length tape.

**library**. (1) A repository for demountable recorded media, such as magnetic tapes. (2) In ADSM, a collection of one or more drives, and possibly robotic devices (depending on the library type), which can be used to access storage volumes. (3) In the AS/400 system, a system object that serves as a directory to other objects. A library groups related objects, and allows the user to find objects by name.

**linear data set**. A type of MVS data set that ADSM uses for the database, the recovery log, and storage pools. The data set must be preallocated using VSAM IDCAMS and formatted by ADSM for its use. See *minidisk*.

**load**. See *mount*.

**local area network (LAN)**. A network in which a set of devices are connected to one another for communication and that can be connected to a larger network.

**log pool size**. The size of an area in memory used to store recovery log pages.

**logical volume**. The combined space from all volumes defined to either the database or the recovery log. In ADSM, the database is one logical volume and the recovery log is one logical volume.

**low migration threshold**. A percentage of the storage pool capacity that specifies when ADSM can stop the migration of files to the next storage pool. Contrast with *high migration threshold*. See *server migration*.

# M

**machine information**. Details about the machine on which a client node resides.

**macro file**. An optional file that contains one or more administrative commands and is invoked from an administrative client.

**management class**. A policy object that users can bind to each file to specify how the server manages the file. The management class can contain a backup copy group, an archive copy group, and space management attributes. The copy groups determine how the ADSM server manages backup versions or archive copies of files. The space management attributes determine whether files are eligible for migration from client nodes to ADSM storage, and under what conditions. See also *copy group*, *binding* and *rebinding*.

**mask**. A pattern of characters that controls the keeping, deleting, or testing of positions of another pattern of characters or bits.

**maximum extension**. Specifies the maximum amount of storage space, in megabytes, that you can extend the database or recovery log.

**maximum reduction**. Specifies the maximum amount of storage space, in megabytes, that you can reduce the database or recovery log.

**maximum utilization**. The highest percentage of assigned capacity used by the database or recovery log.

**MB**. Megabyte.

**megabyte (MB)**. (1) For processor storage and real and virtual memory, $2^{20}$ or 1 048 576 bytes. (2) For disk storage capacity and transmission rates, 1 000 000 bytes.

**migrate**. (1) To move data from one storage pool to the storage pool specified as the next pool in the hierarchy. The process is controlled by the high and low migration thresholds for the first storage pool. See *high migration threshold* and *low migration threshold*. (2) To copy a file from a client node to ADSM storage. ADSM replaces the file with a stub file on the client node. The process is controlled by the include-exclude list and by space management attributes in management classes.

**migration**. The process of moving data from one storage location to another. See *client migration* and *server migration*.

**minidisk**. A logical subdivision of a VM physical disk that provides storage on contiguous cylinders of DASD. On a VM server, a minidisk can be defined as a disk volume that can be used by the database, recovery log, or a storage pool. See also *linear data set*.

**mirroring**. A feature that protects against data loss within the database or recovery log by writing the same data to multiple disks at the same time. Mirroring supports up to three exact copies of each database or recovery log volume. 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 recovery media**. Media that is kept at a different location to ensure its safety if a disaster occurs at the primary location of the computer system. The media contains data necessary to recover the ADSM server and clients. The offsite recovery media manager, which is part of DRM, identifies recovery media to be moved offsite and back onsite, and tracks media status.

**offsite volume**. A removable media volume that is at a location where it cannot be mounted for use.

**OpenEdition MVS**. MVS/ESA services that support an environment within which operating systems, servers, distributed systems, and workstations share common interfaces. OpenEdition MVS supports standard application development across multivendor systems and is required to create and use applications that conform to the POSIX standard.

**open registration**. A registration process in which users can register their own workstations as client nodes with the server. Contrast with *closed registration*.

**Operating System/2 (OS/2)**. An operating system used in IBM PC AT, PS/2, and compatible computers. OS/2 is one of the supported client environments and one of the supported server environments.

**operator privilege class**. An administrative privilege class that allows an administrator to issue commands that control the operation of the server. This privilege class allows disabling or halting the server to perform maintenance, enabling the server, canceling server processes, and managing tape.

**optical 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.

**recovery media**. Media that contains data necessary to recover the ADSM server and clients.

**reduce**. The process of freeing up enough space to allow you to delete a volume from the database or recovery log. Contrast with *extend*.

**REEL**. On a VM server, a device class that is defined by ADSM at installation. It is used with VM servers 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 files or directories from a client domain 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.

**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 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 data 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
  *See* cartridge
349X library 203
3590 drive support 204
  device class, defining 221
  enabling, procedure 204

## A

absolute mode, description of 146
access mode 286
accounting record
  description of 72
  determining for storage pool 263, 266
  monitoring 72
ACTIVATE POLICYSET command 42, 150
activity log
  adjusting the size 70
  description of 70
  monitoring 70
  querying 71
  setting the retention period 71
administrative client
  description of 1
  options 29
  viewing information after IMPORT or EXPORT 314
administrative command line interface 27
administrative commands
  AUDIT LIBVOLUME 216, 217
  AUDIT LICENSE 38, 104
  AUDIT VOLUME 50, 296
  BACKUP DB 55, 342
  BACKUP DEVCONFIG 55
  BACKUP STGPOOL 49, 350
  BACKUP VOLHISTORY 54
  CHECKIN LIBVOLUME 213
  CHECKOUT LIBVOLUME 215
  DEFINE DEVCLASS 48, 231
  DEFINE DOMAIN 41
  DEFINE DRIVE 211
  DEFINE LIBRARY 203, 209
  DEFINE SCHEDULE 167
  DEFINE STGPOOL 49, 264, 265
  DEFINE VOLUME 208, 290
administrative commands *(continued)*
  DELETE DEVCLASS 48, 235
  DELETE DRIVE 213
  DELETE LIBRARY 210
  DELETE LOGVOLUME 93
  DELETE VOLHISTORY 339, 340
  DELETE VOLUME 304, 305
  DISMOUNT VOLUME 218
  DSMFMT 36, 285
  DSMLABEL 288
  EXTEND LOG 345
  GRANT AUTHORITY 106
  HELP 73
  MOVE DATA 301
  QUERY ACTLOG 71
  QUERY DB 90
  QUERY DBBACKUPTRIGGER 343
  QUERY DEVCLASS 48, 233
  QUERY DRIVE 212
  QUERY LIBRARY 210
  QUERY LICENSE 104
  QUERY MOUNT 218
  QUERY OCCUPANCY 277, 278, 279
  QUERY OPTION 70
  QUERY PROCESS 303
  QUERY REQUEST 218
  QUERY STGPOOL 49, 271, 273, 322
  QUERY VOLHISTORY 339
  QUERY VOLUME 292, 293, 303
  REGISTER ADMIN 105
  REGISTER LICENSE 100, 103
  RENAME ADMIN 113
  RESET DBMAXUTILIZATION 82
  RESET LOGMAXUTILIZATION 82
  RESTORE STGPOOL 56, 351, 353
  RESTORE VOLUME 51, 354
  SET ACCOUNTING 72
  SET ACTLOGRETENTION 71
  SET AUTHENTICATION 104
  SET LICENSEAUDITPERIOD 104
  SET LOGMODE 55
  SET PASSEXP 105
  SET SCHEDMODES 162
  SET SERVERNAME 69
  UPDATE ADMIN 105
  UPDATE DBBACKUPTRIGGER 344

administrative commands *(continued)*
- UPDATE DEVCLASS 48, 234
- UPDATE DRIVE 212
- UPDATE LIBRARY 210
- UPDATE LIBVOLUME 208, 216
- UPDATE SCHEDULE 167
- UPDATE VOLUME 290

administrative privilege class
- analyst 110
- description of 106
- granting authority 106
- operator 110
- policy 108
- revoking all 111
- storage 109
- system 107

administrator
- authorizing to manage a policy domain 106
- description of 1
- locking 114
- managing registration 99
- querying 115
- registering 105
- removing 114
- renaming 113
- unlocking 114
- updating 105, 106
- viewing information about 115

analyst privilege class
- description of 110
- granting 110
- revoking 111

API
- compression option 124
- deletion option 125
- description of 1, 7, 31

application client, registering 124

application programming interface
- description of 1, 31
- information about xx
- registering to server 124
- using 7

archive
- amount of space used 279
- defining criteria 139
- description of 4
- processing 136

archive copy group
- defining 149, 150
- deleting 155

archive copy group *(continued)*
- description of 130

archive file management 128

archiving a file 128, 136

ASSIGN DEFMGMTCLASS command 42, 150

assigned capacity 82, 87

association
- defining 172
- deleting 173
- description of 167
- querying 173
- viewing information about 173

audit license 104, 107

AUDIT LICENSE command 104

AUDIT VOLUME command 294, 296

authentication, client/server 104

authority
- granting to administrators 106
- revoking 110

# B

backup
- a file 128, 135, 136
- amount of space used 279
- database 342, 343
- defining criteria 139
- description of 4
- file management 128
- incremental 128, 135
- selective 128, 136
- storage pool 349
- when to perform for database 336

backup copy group
- defining 145, 148
- deleting 155
- description of 130
- frequency 135
- mode 135
- serialization 135

BACKUP DB command 55, 342

BACKUP DEVCONFIG command 55, 341

backup period, specifying for incremental 163

BACKUP STGPOOL command 49, 350

BACKUP VOLHISTORY command 54, 339

backup-archive client
- description of 1
- registering 118

batch mode 28

binding
- description of 133
- file to a management class 133

buffer pool 94
BUFPOOLSIZE option 95

## C

cache
- description of 5
- enabling for disk storage pools 252, 263
- monitoring utilization on disk 276

CANCEL PROCESS command 34, 49, 68, 275
CANCEL SESSION command 34, 65
capacity, assigned 82, 87
cartridge 221
category, 349X library 204
central scheduling
- controlling the workload 163
- coordinating 163
- description of 4, 159

characteristics, machine 392
class, administrator privilege
- analyst 110
- description of 106
- granting authority 106
- operator 110
- policy 108
- revoking all 111
- storage 109
- system 107

class, device
- 3590 219, 221
- 4MM 219, 221
- 8MM 219, 221
- amount of space used 278
- CARTRIDGE 221
- defining for backup 338
- description of 4
- DISK 219
- DLT 219, 221
- FILE 219, 231
- OPTICAL 219, 230
- QIC 219, 221
- selecting for import and export 311
- WORM 219, 230

class, policy privilege
- description of restricted 108
- description of unrestricted 108
- granting restricted 108

class, policy privilege *(continued)*
- granting unrestricted 108
- reducing unrestricted to restricted 112
- revoking 111

class, storage privilege
- description of restricted 109
- description of unrestricted 109
- granting restricted 109
- granting unrestricted 109
- reducing unrestricted to restricted 112
- revoking 111

client
- administrative 1
- application 31
- backup-archive 4
- space management 1

client migration 136, 137
client node
- amount of space used 278
- locking 120
- managing registration 99, 115
- querying 120
- registering 118, 124
- removing 124
- renaming 119
- setting password authentication 104
- setting scheduling mode 163
- unlocking 120
- updating 119
- viewing information about 120

client queries to the server, setting the frequency 165
client session
- canceling 65
- managing 64
- querying 64
- recycling 63
- viewing information about 64

client system options file 125
client-polling scheduling 160, 165
client/server, description of 1
closed registration
- description of 117
- setting 116

collocation
- definition 253, 263, 266
- description of 5
- determining whether to use collocation 253, 263, 266
- enabling for sequential storage pool 253, 263, 266
- how it affects reclamation 259

collocation *(continued)*
how the server selects volumes when disabled 256
how the server selects volumes when enabled 255
migration thresholds 250
command retry attempts
setting the amount of time between 166
setting the number of 166
command-line interface, using 7
commands, administrative
AUDIT LIBVOLUME 216, 217
AUDIT LICENSE 38, 104
AUDIT VOLUME 50, 296
BACKUP DB 55, 342
BACKUP DEVCONFIG 55
BACKUP STGPOOL 49, 350
BACKUP VOLHISTORY 54
CHECKIN LIBVOLUME 213
CHECKOUT LIBVOLUME 215
DEFINE DEVCLASS 48, 231
DEFINE DOMAIN 41
DEFINE DRIVE 211
DEFINE LIBRARY 203, 209
DEFINE SCHEDULE 167
DEFINE STGPOOL 49, 264, 265
DEFINE VOLUME 208, 290
DELETE DEVCLASS 48, 235
DELETE DRIVE 213
DELETE LIBRARY 210
DELETE LOGVOLUME 93
DELETE VOLHISTORY 339, 340
DELETE VOLUME 304, 305
DISMOUNT VOLUME 218
DSMFMT 36, 285
DSMLABEL 288
EXTEND LOG 345
GRANT AUTHORITY 106
HELP 73
MOVE DATA 301
QUERY ACTLOG 71
QUERY DB 90
QUERY DBBACKUPTRIGGER 343
QUERY DEVCLASS 48, 233
QUERY DRIVE 212
QUERY LIBRARY 210
QUERY LICENSE 104
QUERY MOUNT 218
QUERY OCCUPANCY 277, 278, 279
QUERY OPTION 70
QUERY PROCESS 303
QUERY REQUEST 218
commands, administrative *(continued)*
QUERY STGPOOL 49, 271, 273, 322
QUERY VOLHISTORY 339
QUERY VOLUME 292, 293, 303
REGISTER ADMIN 105
REGISTER LICENSE 100, 103
RENAME ADMIN 113
RESET DBMAXUTILIZATION 82
RESET LOGMAXUTILIZATION 82
RESTORE STGPOOL 56, 351, 353
RESTORE VOLUME 51, 354
SET ACCOUNTING 72
SET ACTLOGRETENTION 71
SET AUTHENTICATION 104
SET LICENSEAUDITPERIOD 104
SET LOGMODE 55
SET PASSEXP 105
SET SCHEDMODES 162
SET SERVERNAME 69
UPDATE ADMIN 105
UPDATE DBBACKUPTRIGGER 344
UPDATE DEVCLASS 48, 234
UPDATE DRIVE 212
UPDATE LIBRARY 210
UPDATE LIBVOLUME 208, 216
UPDATE SCHEDULE 167
UPDATE VOLUME 290
COMMIT command 33
compression 116, 117, 124
compression option, API 124
configuration, device
files
backup 337, 340
dsmserv.opt 337, 340
information 337, 340
console mode 27, 314
contextual help menu
accessing 23
description of 20
COPY DOMAIN command 41, 143
copy group
deleting 155
description of 4
COPY MGMTCLASS command 41, 144
COPY POLICYSET command 41, 143
COPY SCHEDULE command 45, 171

# D

data
- considering user needs for recovering 190
- exporting 309
- importing 309

data movement, querying about the process 303

data storage
- client files, process for storing 177
- concepts overview 177
- considering user needs for recovering 190
- deleting files from 304
- evaluating 189
- managing 177
- monitoring 294
- tailoring definitions 324

database
- adding space to 85
- allocating space 78
- available space 80
- backup 342
- backup trigger 343
- buffer pool 94, 95
- committing data to 94
- defining a volume 85
- defining mirrored volumes 331
- deleting a volume 93
- deleting space 89
- description of 3, 75
- determining how much space is allocated 79
- ensuring integrity of 6
- estimating the amount of space needed 78
- logical volume 79
- managing 75
- mirroring 330
- monitoring space 82
- monitoring the buffer 95
- optimizing performance 94
- querying the buffer pool 95
- recovering 346
- reducing capacity 92
- resetting buffer pool statistics 96
- restoring 345
- storage pool size effect 77
- transactions 75, 76, 77
- viewing information about 95
- volume placement 83

database backup and recovery
- defining device classes 338
- full backup 336

database backup and recovery *(continued)*
- incremental backup 336
- point-in-time 334, 335
- roll-forward 334, 335
- to most current state 334
- trigger 336

database backup trigger and roll-forward mode 334

database recovery
- example recovery procedures 356
- general strategy 329
- methods 329
- offline utilities 330
- providing 329
- salvage utilities 355
- when to backup 336

day of the week, description of 167

default management class, description of 129

default management classes 132

DEFINE ASSOCIATION command 45, 172

DEFINE COPYGROUP command 41, 145, 148, 149, 150

DEFINE DBBACKUPTRIGGER 55, 338, 343

DEFINE DBCOPY command 54, 331

DEFINE DBVOLUME command 36, 85

DEFINE DOMAIN command 41

DEFINE LOGCOPY command 54, 331

DEFINE LOGVOLUME command 36, 85

DEFINE MACHINE command 391

DEFINE MACHNODEASSOCIATION command 391

DEFINE MGMTCLASS command 41, 144

DEFINE POLICYSET command 41, 143

DEFINE RECMEDMACHASSOCIATION command 395

DEFINE RECOVERYMEDIA command 394

DEFINE SCHEDULE command 44, 167

DEFINE STGPOOL command 264, 265

DEFINE VOLUME command 50, 290

delete
- empty volume 305, 340
- files 304
- scratch volume 244, 340
- storage volume 304
- volume with residual data 305

DELETE ASSOCIATION command 45, 173

DELETE COPYGROUP command 43, 155

DELETE DBBACKUPTRIGGER 55, 343

DELETE DBVOLUME command 37, 93

DELETE DOMAIN command 43, 157

DELETE EVENT command 45, 175

DELETE FILESPACE command 40, 123

DELETE LOGVOLUME command 37, 93
DELETE MGMTCLASS command 43, 156
DELETE POLICYSET command 43, 156
DELETE SCHEDULE command 45, 172
DELETE STGPOOL command 49, 280
DELETE VOLHISTORY command 54, 339, 340
DELETE VOLUME command 51, 304, 305
DEVCONFIG option 340
device class
  3590 219, 221
  4MM 219, 221
  8MM 219, 221
  amount of space used 278
  CARTRIDGE 221
  defining for backup 338
  description of 4
  DISK 219
  DLT 219, 221
  FILE 219, 231
  OPTICAL 219, 230
  QIC 219, 221
  selecting for import and export 311
  WORM 219, 230
device configuration
  files
    backup 337, 340
    dsmserv.opt 337, 340
  information 337, 340
device configuration files, recreating 341
device type
  3590 221
  4MM 221
  8MM 221
  CARTRIDGE 221
  DISK 219
  DLT 221
  FILE 231
  OPTICAL 230
  QIC 221
  WORM 230
DISABLE command 34, 66
disaster recovery
  example recovery procedures 356
  general strategy 329
  methods 329
  offline utilities 330
  providing 329
  salvage utilities 355
  when to backup 336
disaster recovery manager
  awk script 409
  client recovery information 361
  creating a disaster recovery plan 368
  customizing 402
  enabling 363
  features 360
  moving volumes back onsite 366
  offsite recovery media management 361
  overview of set up 362
  recovery media 394
  saving machine characteristics 391
  stanzas, recovery instructions 407
disk device class, defined 219
disk storage pool
  estimating space 260
  estimating space for archived files 261
  estimating space for backed up files 261
documentation, user xvii
drive
  defining 211
  deleting 213
  querying 212
  updating 212
DSMFMT utility 285, 287
DSMLABEL utility
  identifying drives 288
  insert category 289, 290
  labeling sequential storage pool volumes 288
  overwriting existing volume labels 288
  using a library device 288
  volume labeling examples 289
DSMSERV AUDITDB utility 56, 355
DSMSERV DUMPDB utility 56, 355
DSMSERV INSTALL utility 56, 355
DSMSERV LOADDB utility 56, 355
DSMSERV RESTORE DB command 55, 346, 348
dsmserv.opt 337, 340
dynamic serialization, description of 145, 149

# E

edit, description of menu choices 14
ENABLE command 34, 66
ENABLE3590LIBRARY parameter 204
event record
  deleting 175
  description of 159
  removing from the database 175

event retention period 175
event, description of 159
EXPINTERVAL option 134, 152
expiration date, setting 168
expiration processing 134, 152
EXPIRE INVENTORY command 134, 152
export
  labelling tapes 312
  monitoring 312
  planning for sequential media 311
  PREVIEW parameter 310
  querying about a process 312
  querying the activity log 314
  using scratch media 311
  viewing information about a process 312
EXPORT ADMIN command 52, 317
EXPORT commands 313, 314
EXPORT NODE command 52, 318
EXPORT POLICY command 52, 319
EXPORT SERVER command 52, 311, 317
exporting
  administrator data 317
  client node data 318
  data to tape 315
  description of 5, 309
  policy data 319
  server data 317
EXTEND DB command 36, 87
EXTEND LOG command 36, 87
external media management
  initialization requests 414
  interface description 413
  processing during server initialization 413
  volume dismount requests 417
  volume mount requests 415
  volume release requests 417

F
file data, importing 309
FILE device type
  defining device class 219
  deleting scratch volumes 340
file size, determining maximum for storage pool 263
filespace
  deleting 123
  description of 4, 120
  querying 122
  viewing information about 122
formatting
  database volume 84
  recovery log volume 84
  storage pool volume 287
frequency, description of 146

G
general help menu
  accessing 21
  description of 20
GRANT AUTHORITY command 38
graphical user interface
  accessing contextual help 23
  accessing general help 21
  accessing help from windows 21
  accessing task help 22
  administration window 13
  completing tasks when using 15
  ending a client session 19
  starting from a window 13
  starting from an icon 13
  using 7
  using hypertext links 24
  using menu choices 14
  using online help 20
  using the search facility 25

H
HALT command 33, 62
halting the server 62
help
  description of 21
  description of menu choices 14
HELP command 35
help index menu, description of 20
hierarchical storage management
  archive policy, relationship to 137
  backup policy, relationship to 137
  description 128
  files, destination for 144
  migration of client files
    description 129
    eligibility 136
  policy for, setting 144
  premigration 129
  recall of migrated files 129
  reconciliation between client and server 129
  selective migration 129

hierarchical storage management *(continued)*
  setting policy for   137, 144
  stub file   129
hierarchy, storage
  defining in reverse order   265
  establishing   245
HSM
  *See* hierarchical storage management

# I

import
  monitoring   312
  PREVIEW parameter   310, 320
  querying about a process   312
  querying the activity log   314
  recovering from an error   328
  viewing information about a process   312
IMPORT ADMIN command   52, 319
IMPORT commands   313, 314
IMPORT NODE command   52, 319, 326
IMPORT POLICY command   52, 319
IMPORT SERVER command   52, 319, 326
importing
  data   319
  data storage definitions   322, 324
  date of creation   326
  description of   5, 309
  directing messages to an output file   323
  duplicate file spaces   325
  file data   325
  policy definitions   323
  server control data   324
  subsets of information   327
include-exclude file
  description of   4
  for policy environment   139
incremental backup
  file eligibility for   135
incremental backup frequency, specifying   163
initial start date, description of   167
initial start time, description of   167
interactive mode   29
interface, application programming
  description of   1, 31
  information about   xx
  registering to server   124
  using   7
interface, command-line   7
interface, graphical user
  accessing contextual help   23
  accessing general help   21
  accessing help from windows   21
  accessing task help   22
  administration window   13
  completing tasks when using   15
  ending a client session   19
  starting from a window   13
  starting from an icon   13
  using   7
  using hypertext links   24
  using menu choices   14
  using online help   20
  using the search facility   25

# L

labels
  overwriting existing labels   288
  sequential storage pools   288
  volume examples   289
  volumes using a library device   288
libraries
  349X   203
  auditing volume inventory   216
  automated   213
  defining   209
  deleting   210
  external   206
  managing   201, 209
  manual   202
  querying   210
  SCSI   203, 217
  single drive   207
  type   202
  updating   210
license
  compliance   103
  example   103
  features
    for additional clients   100
    for clients other than AIX   101
    for device module support   102
  monitoring   104
  trial   104
  using   99
link recycling   63
LOCK ADMIN command   39, 114

LOCK NODE command 39, 120
log mode
  normal 343, 345
  roll-forward 343, 345
  setting 345
LOGPOOLSIZE option 94

M
machine characteristics 392
machine recovery information 393
MACRO command 29
macros 29
management class
  assigning a default 150
  associating a file with 133
  binding a file to 133
  controlling user access 131
  copying 138, 144
  default 132
  defining 144
  deleting 156
  description of 4, 129, 130, 131
  rebinding a file 134
  updating 138, 144
management class configuration 131
MANUAL libraries 202
maximum extension 86
media loss, recovery from 354
menu bar choices 14
menu choices 14
migrating a file 128, 136
migration
  automatic, for HSM client
    demand 129
    threshold 129
  canceling the server process 275
  defining threshold for disk storage pool 249
  defining threshold for tape storage pool 250
  description of 5
  description, server process 248
  monitoring thresholds for storage pools 273
  premigration for HSM client 129
  providing additional space for server process 275
  providing users with immediate access to files on disk 250
  reconciliation 129
  selective, for HSM client 129
  stub file on HSM client 129
  threshold for a storage pool
    high 248
migration *(continued)*
  threshold for a storage pool *(continued)*
    low 248
mirrored volume
  description of 332
  querying 332
  viewing information about 332
mirroring
  advantages 330
  database 330
  defining volumes 331
  description of 6
  recovery log 330, 331
  recovery procedure 333
mode
  description of 146
  scheduling 162
modified mode, description of 146
mount
  library 230
  limit, for tapes 221
  query 47
  retention period 222
  wait period 222
mount mode 28
mount operations 217
MOVE DATA command 50, 301
MOVE DRMEDIA command 367

O
occupancy, querying 277
offsite recovery media (for DRM)
  about 361
  volumes
    moving back onsite 366
    sending offsite 364
    states 364, 366
online help
  accessing from windows 21
  finding information 24
  using 20
  using hypertext links 24
  using the search facility 25
open registration
  description of 116
  setting 116
operator privilege class
  description of 110
  granting 110

operator privilege class *(continued)*
- revoking 111

optical device class
- defined 219
- mount limit 230
- recording format 231

option
- DEVCONFIG 340
- VOLUMEHISTORY 339

options file, server
- *See* dsmserv.opt

options, querying
- BUFPOOLSIZE 95
- LOGPOOLSIZE 94
- VIRTUALMOUNTPOINT 120

## P

page, description of 94

password
- setting authentication for a client 104
- setting expiration 105

period, specifying for an incremental backup 163

policies, managing ADSM 127

policy definitions, importing 323

policy domain
- creating 142
- deleting 157
- description of 129, 130
- updating 138, 142

policy objects
- deleting 155
- description of 129
- querying 152

policy operations 128

policy privilege class
- description of restricted 108
- description of unrestricted 108
- granting restricted 108
- granting unrestricted 108
- reducing unrestricted to restricted 112
- revoking 111

policy set
- activating 150
- copying 138, 143
- defining 143
- deleting 156
- description of 129, 130
- updating 143
- validating 150, 152

pool, storage
- amount of space used 278
- backup and recovery 349
- copy 238, 239
- creating a hierarchy 245
- defining 262
- defining for disk 264
- defining for tape 264
- deleting 280
- description of 238
- determining access mode 263, 266
- determining maximum file size 263
- determining whether to use collocation 253, 263, 266
- enabling cache for disk 252, 263
- estimating space for archived files on disk 261
- estimating space for backed up files on disk 261
- estimating space for disk 260
- estimating space for sequential 262
- estimating space in multiple 245
- managing 237
- monitoring 271
- moving files 301
- moving files between 301
- overview 183
- primary 238
- querying 271
- random access 238
- recovery log, effect on 77
- restore 351, 353
- sequential access 238
- updating 262
- updating for disk 265
- using cache on disk 252, 263
- viewing information about 271

prefix, for recovery instructions 404

prefix, for recovery plan file 404

premigration 129

PREPARE command 368

PREVIEW parameter 310, 320

priority 168

private category 204

privilege class, administrator
- analyst 110
- description of 106
- granting authority 106
- operator 110
- policy 108
- revoking all 111
- storage 109

privilege class, administrator *(continued)*
- system 107

privilege class, policy
- description of restricted 108
- description of unrestricted 108
- granting restricted 108
- granting unrestricted 108
- reducing unrestricted to restricted 112
- revoking 111

process canceling 68
product information menu, description of 20
programming interface notice xiii
publications xvii

# Q

QUERY ACTLOG command 35, 71, 314
QUERY ADMIN command 39, 115
QUERY ASSOCIATION command 45, 173
QUERY CONTENT command 50, 298
QUERY COPYGROUP command 42, 153, 324
QUERY DB command 37, 90, 95
QUERY DBBACKUPTRIGGER command 55, 343
QUERY DBVOLUME command 36, 54, 89, 332
QUERY DEVCLASS command 311
QUERY DOMAIN command 42, 155
QUERY EVENT command 45, 174
QUERY FILESPACE command 40, 122
QUERY LICENSE command 38, 104
QUERY LOG command 37, 97
QUERY LOGVOLUME command 37, 54, 89, 332
QUERY MGMTCLASS command 42, 153
QUERY NODE command 40, 120
QUERY OCCUPANCY 278, 279
QUERY OCCUPANCY command 49, 277
QUERY OPTION command 35
QUERY POLICYSET command 42, 154
QUERY PROCESS command 34, 50, 52, 67, 275, 303, 312
QUERY SCHEDULE command 45, 172
QUERY SESSION command 34, 64
QUERY STATUS 35, 68
QUERY STGPOOL command 49, 271, 273, 276
QUERY VOLHISTORY command 54
QUERY VOLUME command 50, 292, 293, 303
querying for general information 292
querying policy objects 152
querying storage volumes 293
QUIT command 29

# R

randomize, description of 164
read-only access mode 286
read/write access mode 286
rebinding
- description of 134
- file to a management class 134

recalling a file
- selective 129
- transparent 129

reclamation
- affect of collocation on 259
- description of 5
- setting a threshold for sequential storage pool 257, 264, 266
- threshold 5

reclamation threshold, setting for sequential storage pool 257, 264, 266
recovering storage pools 349
recovering the database 346
recovery from disaster
- *See* disaster recovery

recovery log
- adding space to 85
- allocating space 78
- available space 80
- buffer pool 96
- consistent database image 75
- defining a volume 85
- defining mirrored volumes 331
- deleting a volume 93
- deleting space 89
- description of 3, 75
- determining how much space is allocated 79
- estimating the amount of space needed 79
- logical volume 79
- managing 75
- mirroring 330, 331
- monitoring space 79
- monitoring the buffer pool 96
- optimizing performance 94
- querying the buffer pool 97
- reducing capacity 92
- size of 345
- storage pool size effect 77
- viewing information about 97
- volume placement 83
- when to backup 336

recovery log mode

recovery log mode *(continued)*
normal 343, 345
roll-forward 343, 345
setting 345
recovery plan file
creating 368
example 379
prefix 404
stanzas 369
recovery, disaster
client 399
example recovery procedures 356
general strategy 329
media 394
methods 329
offline utilities 330
providing 329
salvage utilities 355
server 396
when to backup 336
recycling links 63
REDUCE DB command 37, 92
REDUCE LOG command 37, 92
REGISTER ADMIN command 38, 105
REGISTER LICENSE command 38, 100, 103
REGISTER NODE command 39, 40, 118
registering a workstation 124
registration
closed 117
description of 116
managing client node 115
managing for a client node 99
managing for an administrator 99
open 116
setting for a client node 116
REMOVE ADMIN command 39, 114
REMOVE NODE command 40, 124
RENAME ADMIN command 39, 113
RENAME FILESPACE command 53
RENAME NODE command 39, 119
renaming an administrator ID 113
RESET BUFPOOL command 37, 96
RESET DBMAXUTILIZATION command 36, 82
RESET LOGMAXUTILIZATION command 36, 82
restarting the server 62
RESTORE STGPOOL command 49, 351, 353
RESTORE VOLUME command 51, 354
restoring a file 128
restoring the database
point-in-time 346
restoring the database *(continued)*
to its most current state 348
restricted policy privilege
granting 108
reducing 113
restricted storage privilege
granting 109
reducing 113
retain extra versions, description of 148
retain only version, description of 148
retention grace period
description of archive 142
description of backup 142
using archive 142
using backup 142
retrieving a file 128
REVOKE AUTHORITY command 38, 110
roll-forward recovery
database backup trigger 334
mirroring recovery log 334
recovery log 334
ROLLBACK command 33
root user tasks
deactivate SNASVCMG mode sessions 63
define compression 116
free unused SNA LU6.2 links 63
registering your workstation 124

# S

schedule
associating client node 172
client options to use 170
coordinating 163
copying 171
day of the week 167
defining 167
deleting 172
description of 159
expiration date 168
files to process 169
frequency of service 168
initial start date 167
initial time 167
managing associations 167
priority 168
querying 172
startup window 164, 168
type of action 170
updating 167

schedule *(continued)*
viewing information about 172
schedule event
managing 174
querying 174
viewing information about 174
scheduled operations, setting the maximum 163
scheduler workload, controlling 163
scheduling mode
client-polling 160
description of 159
selecting 162
server-prompted 161
setting on a client node 163
setting on the server 162
scheduling, central
controlling the workload 163
coordinating 163
description of 4, 159
scratch category, 349X library 204
scratch volume
deleting 244, 340
number allowed in a storage pool 263, 266
using in storage pools 208, 244
SCSI libraries 203
search facility, using 25
selected, description of menu choices 14
selective backup 128, 136
selective recall 129
sequential storage pool
auditing multiple volumes in 297
collocation 243
estimating space 262
reclamation 257
server
canceling process 68
console session 30
description of 1
disabling 66
disabling access 66
enabling 66
enabling access 66
halting 62
managing operation 61
managing processes 66
querying about processes 67
querying options 70
querying status 68
restarting 62
scheduling mode 44
server *(continued)*
setting the name 69
starting 61
stopping 62
viewing information about 68
viewing information about processes 67
server console, description of 105
server option
DEVCONFIG 340
ENABLE3590LIBRARY 204
VOLUMEHISTORY 339
server options file
*See* dsmserv.opt
server-prompted scheduling 161
session
canceling 65
setting the maximum percentage for scheduled operations 163
SET ACCOUNTING command 35, 72
SET ACTLOGRETENTION command 35, 71
SET AUTHENTICATION command 38, 104
SET DRMCHECKLABEL command 405
SET DRMCOPYSTGPOOL command 402
SET DRMCOURIERNAME command 405
SET DRMDBBACKUPEXPIREDAYS command 406
SET DRMFILEPROCESS command 406
SET DRMINSTPREFIX command 404
SET DRMPLANPOSTFIX command 403
SET DRMPLANPREFIX command 404
SET DRMPRIMSTGPOOL command 403
SET DRMVAULTNAME command 406
SET EVENTRETENTION command 45, 175
SET LICENSEAUDITPERIOD command 38
SET LOGMODE command 55, 344
SET MAXCMDRETRIES command 44, 166
SET MAXSCHEDSESSIONS command 163
SET PASSEXP command 38, 105
SET QUERYSCHEDPERIOD command 44, 165
SET RANDOMIZE command 44, 164
SET REGISTRATION command 39, 116
SET RETRYPERIOD command 44, 166
SET SCHEDMODES command 162
SET SERVERNAME command 35
setting a password 124
setting compression 124
shared dynamic serialization, description of 145, 149
shared static serialization, description of 145, 149
single drive library 207
SNA LU6.2 63

SNASVCMG session 63
space
  planning how much to allocate 83
  unusable 80
  usable 80
space management
  *See* hierarchical storage management
special file names 207
standard management class, copying 144
standard storage management policies, using 137
start time, randomizing for a schedule 164
starting the server 61
startup window, description of 164
static serialization, description of 145, 149
stopping the server 62
storage devices
  IBM 3590 tape drives 219, 220
  quarter-inch cartridge tape drives 219, 220
storage hierarchy
  defining in reverse order 265
  establishing 245
storage management policies
  description of 4, 129
  managing 127
  tailoring 138
  using standard 137
storage occupancy, querying 277
storage pool
  amount of space used 278
  backup and recovery 349
  copy 238, 239
  creating a hierarchy 245
  defining 262
  defining for disk 264
  defining for tape 264
  deleting 280
  description of 238
  determining access mode 263, 266
  determining maximum file size 263
  determining whether to use collocation 253, 263, 266
  enabling cache for disk 252, 263
  estimating space for archived files on disk 261
  estimating space for backed up files on disk 261
  estimating space for disk 260
  estimating space for sequential 262
  estimating space in multiple 245
  managing 237
  monitoring 271
  moving files 301
storage pool *(continued)*
  moving files between 301
  overview 183
  primary 238
  querying 271
  random access 238
  recovery log, effect on 77
  restore 351, 353
  sequential access 238
  updating 262
  updating for disk 265
  using cache on disk 252, 263
  viewing information about 271
storage pool backup
  full 350
  incremental 350
storage privilege class
  description of restricted 109
  description of unrestricted 109
  granting restricted 109
  granting unrestricted 109
  reducing unrestricted to restricted 112
  revoking 111
storage volume
  auditing 294
  contents 298
  formatting random access 287
  information about 292
  labeling sequential access 288
  monitoring use 292
  overview 183
  quarter-inch tape cartridge 285
stub file 129
system privilege class
  description of 107
  granting 107
  reducing 111
  revoking 111

# T

tape
  exporting data 315
  label prefix 223
  planning for exporting data 311
  recording format 223
  scratch, determining use 244, 263, 266
  setting mount retention period 222
task help menu
  accessing 22

task help menu *(continued)*
  description of 20
trademarks xv
transactions, database 75, 76, 77
transparent recall 129
type, device
  3590 221
  4MM 221
  8MM 221
  CARTRIDGE 221
  DISK 219
  DLT 221
  FILE 231
  OPTICAL 230
  QIC 221
  WORM 230

## U

unavailable access mode 286
UNLOCK ADMIN command 39, 114
UNLOCK NODE command 39, 120
unplanned shutdown 62
unrestricted policy privilege
  granting 108
  reducing to restricted 112
unrestricted storage privilege
  granting 109
  reducing to restricted 112
unusable space 80
UPDATE ADMIN command 38, 106
UPDATE COPYGROUP command 42, 145, 149
UPDATE DBBACKUPTRIGGER command 55
UPDATE DOMAIN command 41, 143
UPDATE MGMTCLASS command 41, 144
UPDATE NODE command 39, 119
UPDATE POLICYSET command 41, 143
UPDATE RECOVERYMEDIA command 395
UPDATE SCHEDULE command 44, 167
UPDATE STGPOOL command 49
UPDATE VOLUME command 50, 290
usable space 80
user documentation xvii
using help menu, description of 20
utilization
  description of 82
  monitoring 82

## V

VALIDATE POLICYSET command 42, 150
VARY command 35, 68
varying volumes on or off line 68
versions data deleted, description of 147
versions data exists, description of 146
view, description of menu choices 14
VIRTUALMOUNTPOINT option 120
volume
  allocating space for disk 287
  auditing 216, 294
  auditing considerations 296
  defining for database 85
  defining for recovery log 85
  defining to storage pools 290
  deleting 304, 305, 340
  detailed report 299
  determining which are mounted 218, 312
  disk storage 290
  disk storage pool 296
  dismounting 218
  managing 213
  monitoring movement of data 303
  monitoring use 292
  mount retention time 222
  moving files between 300
  new 213
  preparing for data storage 285
  private 208
  querying 293
  querying contents 298
  querying for general information 292
  random access storage pools 238, 244
  recovery using mirroring 333
  removing 215
  scratch category 208
  scratch, using 244
  sequential 290
  sequential storage pools 288
  setting access mode 286
  standard report 299
  status codes 208
  updating 216
  updating to storage pools 290
  varying 68
volume capacity 227
volume copy
  allocating to separate disks 331
  description of 330

volume history backup
  and point-in-time 337
volume history files, establishing 339
VOLUMEHISTORY option 337, 339

W
Windows
  GUI administration 13
  using menu choices 14
workstation, registering 124
WORM device class
  defined 219
  mount limit 230
  recording format 231

# Communicating Your Comments to IBM

ADSTAR Distributed Storage Manager
for AIX
Administrator's Guide
Version 2

Publication No. SH35-0134-01

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 AIX**
**Administrator’s Guide**
**Version 2**

**Publication No. SH35-0134-01**

### Overall, how satisfied are you with the information in this book?

| | Very Satisfied | Satisfied | Neutral | Dissatisfied | Very Dissatisfied |
|---|---|---|---|---|---|
| Overall satisfaction | □ | □ | □ | □ | □ |

### How satisfied are you that the information in this book is:

| | Very Satisfied | Satisfied | Neutral | Dissatisfied | Very Dissatisfied |
|---|---|---|---|---|---|
| Accurate | □ | □ | □ | □ | □ |
| Complete | □ | □ | □ | □ | □ |
| Easy to find | □ | □ | □ | □ | □ |
| Easy to understand | □ | □ | □ | □ | □ |
| Well organized | □ | □ | □ | □ | □ |
| Applicable to your tasks | □ | □ | □ | □ | □ |

### Please tell us how we can improve this book:

Thank you for your responses. May we contact you? □ Yes □ No

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

Name

Address

Company or Organization

Phone No.

Cut or Fold
Along Line

**Readers' Comments — We'd Like to Hear from You**
SH35-0134-01

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

# BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Information Development
Department 61C
9000 South Rita Road
TUCSON AZ 85775-4401

Fold and Tape **Please do not staple** Fold and Tape

Cut or Fold
Along Line

Program Number: 5765-564

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SH35-0134-01

*Spine information:*

IBM

ADSTAR Distributed Storage Manager
for AIX

Administrator's Guide

*Version 2*