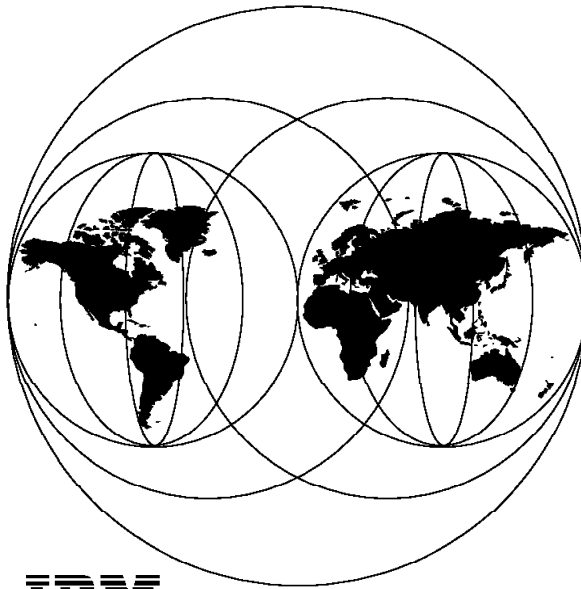


SG24-4705-00

AIX Tape Management

August 1996



IBM

**International Technical Support Organization
San Jose Center**

SG24-4705-00

International Technical Support Organization

AIX Tape Management

August 1996



Take Note!

Before using this information and the products it supports, be sure to read the general information in Appendix J, "Special Notices" on page 253.

First Edition (August 1996)

This edition applies to Versions 1 and 2 of ADSTAR Distributed Storage Manager (ADSM) for AIX, Program Numbers 5765-203 and 5765-564, Network Tape Access and Control System for AIX (NetTAPE), Program Number 5765-637, NetTAPE Tape Library Connection, Program Number 5765-643, and Client Input Output/Sockets (CLIO/S), Program Number 5648-129.

Comments may be addressed to:

IBM Corporation, International Technical Support Organization
Dept. 471 Building 80-E2
650 Harry Road
San Jose, California 95120-6099

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

© **Copyright International Business Machines Corporation 1996. All rights reserved.**

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

Contents

Figures	xi
Tables	xiii
Preface	xv
How This Redbook Is Organized	xvii
The Team That Wrote This Redbook	xxiii
Comments Welcome	xxiv
Chapter 1. Tape Management Overview	1
1.1 UNIX Tape Environments and Applications	2
1.2 Tape Usage Considerations	8
1.2.1 Media Type and Reliability	9
1.2.2 Recording Density	9
1.2.3 Tape Capacity	9
1.2.4 Performance	10
1.2.5 Manual and Automated Tape	10
1.2.6 Device Connectivity	11
1.2.7 Media Movement between Storage Locations	11
1.2.8 Media Operation	12
1.2.9 Application Usage	12
1.2.10 Business Benefits	12
1.3 How AIX Supports Tapes	13
1.4 ADSTM Tape Management	14
1.4.1 Objectives	14
1.4.2 Functions	14
1.4.3 Benefits	14
1.5 NetTAPE Tape Management	15
1.5.1 Objectives	16
1.5.2 Functions	16
1.5.3 NetTAPE with ADSTM	17
1.5.4 CLIO/S	17
Chapter 2. AIX Tape Management	19
2.1 Tape Support in UNIX	19
2.1.1 How Tape Devices Are Known to UNIX	19
2.1.2 Device Drivers	20
2.1.3 Writing to and Reading from Files on Tape	21
2.1.4 AIX Tape Commands and Programming Interfaces	22

2.1.5	Hardware Tape Drivers	24
2.1.6	ADSM Tape Drivers	25
2.1.7	Tape Drivers Summary	26
2.2	Tape Technology	27
2.2.1	Recording Techniques	27
2.2.2	Device Connections and Protocols	29
2.3	Tape Library Technology	31
2.3.1	What Is a Tape Library?	31
2.3.2	Types of Tape Libraries	32
2.3.3	Library Device Drivers	33
2.4	Tape Device and Library Sharing	34
2.4.1	Tape Device Sharing	35
2.4.2	Tape Library Sharing and Partitioning	37
2.5	Tape Management Requirements and Trends	38
Chapter 3. ADSM Tape Management		41
3.1	ADSM Structure	41
3.2	ADSM Tape Device Support	46
3.2.1	Manual Tape Devices	46
3.2.2	Small SCSI Tape Libraries	46
3.2.3	Magstar 3590-B11 Libraries	46
3.2.4	3494 and 3495 Libraries	47
3.2.5	ADSM Tape Device Driver	47
3.2.6	ADSM Device Definitions	47
3.3	ADSM Tape Media Management	49
3.3.1	Types of ADSM Tape Volumes	49
3.3.2	Scratch and Private Volumes	50
3.3.3	Preparing Volumes for ADSM	51
3.3.4	Managing Volumes in Automated Libraries	53
3.3.5	Barcode Support	56
3.3.6	Retiring and Replacing Volumes	57
3.3.7	Cleaning Tapes	60
3.3.8	Managing Offsite Volumes	61
3.4	ADSM Tape Operation	62
3.4.1	Direct Client Backup, Archive, and Migration to Tape	62
3.4.2	Storage Pool Migration	63
3.4.3	Tape Reclamation	64
3.4.4	Collocation	67
3.4.5	Backing Up ADSM	69
3.4.6	Export and Import	74
3.4.7	Tape Drive Preemption	75
3.5	ADSM Device Sharing	76
3.5.1	Without the External Library Manager	76
3.5.2	Sharing with Other Applications on the Same System	76

3.5.3	Sharing with Other Systems	77
3.5.4	Library Sharing or Partitioning	77
3.6	ADSM Tape Performance	78
3.6.1	Data Movement Batches	80
3.6.2	Management Classes for Directories	80
Chapter 4.	NetTAPE Overview	83
4.1	Objectives	83
4.2	Functions	83
4.2.1	What NetTAPE Supports	84
4.2.2	What NetTAPE Does Not Support	87
4.3	When Would You Use NetTAPE?	88
4.4	Technical Details	88
4.4.1	NetTAPE Components	89
4.4.2	End-User Sample Scenario	91
4.4.3	Central Operations	93
4.4.4	Device Pools	95
4.4.5	Tape Device Allocation	96
4.4.6	Tape Libraries	97
4.4.7	Mount Queues	97
4.4.8	Remote Tape Access	98
4.4.9	Tape Labels	99
4.4.10	Record Formats	100
4.4.11	End-User Interfaces	100
4.4.12	Support for the ADSM External Library	101
4.4.13	Volume Information (VFILE)	101
4.5	Platforms Supported	101
4.5.1	Workstation Hardware	102
4.5.2	Workstation Software	102
4.5.3	Network Support	102
4.5.4	Tape Device Support	102
4.5.5	Tape Library Support	104
Chapter 5.	NetTAPE Installation and Configuration	105
5.1	Planning	105
5.2	Installation	107
5.3	NetTAPE Server User IDs	108
5.4	Configuration	108
5.4.1	Daemons	109
5.4.2	Configuration Files	111
5.4.3	Client User IDs	113
5.4.4	Updating NetTAPE Code	113
5.4.5	Sample Configuration Files	113
5.5	Additional Steps to Configure 3494, 3495 and STK Tape Libraries	120

5.5.1 Overall Steps	120
5.5.2 Configuring the nettape.adm.config File	124
Chapter 6. NetTAPE Administrator Operations and End-User and Application Use	127
6.1 NetTAPE Administrator Operations	127
6.1.1 How to Start and Stop the NetTAPE Daemon	127
6.1.2 Starting the NetTAPE GUI	128
6.1.3 Typical Operator Tasks	129
6.1.4 Occasional Operator Tasks	134
6.1.5 Error Handling	136
6.2 End-User and Application NetTAPE Use	137
6.2.1 Determining the Available Device Pools for Your Allocation Request	137
6.2.2 Allocating a Tape Device or Sending a Tape Mount Request	138
6.2.3 Unallocating a Tape Device	142
6.2.4 Obtaining Information about Tape Volumes Used	143
6.2.5 Using the Volume Information File	144
6.2.6 Using MultiVolume Data Sets	145
6.2.7 Using Tape Labels on the 3480, 3490, and 3590	147
6.2.8 Setting an Application Block Size	149
6.2.9 Using ACL or ACF 3480, 3490, and 3590 Tape Libraries	150
6.2.10 Writing Special Applications Using the NetTAPE API	150
Chapter 7. ADSM Tape Management with NetTAPE	153
7.1 The ADSM External Library and NetTAPE ADSM External Manager	153
7.1.1 How Does an ADSM External Library Work?	153
7.1.2 The NetTAPE External Manager	154
7.1.3 Advantages of External Libraries	154
7.1.4 Sample Configurations	155
7.2 Configuring the ADSM External Library for Use with the NetTAPE ADSM External Manager	158
7.2.1 Running the ADSM Server Under Its Own User ID	158
7.2.2 Defining the ADSM Server to NetTAPE	160
7.2.3 Defining the External Library and External Manager	160
7.2.4 Migrating an Existing ADSM Server	161
7.3 Managing an ADSM NetTAPE Environment	166
7.3.1 Which Do You Start First—NetTAPE or ADSM?	166
7.3.2 Labeling ADSM Tapes with NetTAPE	166
7.3.3 Mounts	167
7.3.4 Deadlocks	169
7.3.5 ADSM Database Restore	170
7.3.6 What If NetTAPE Assigns ADSM a Remote Tape Drive?	171
7.3.7 Stopping ADSM While a Volume Is Mounted	172

7.3.8 Managing Library Volumes for ADSM	172
Chapter 8. CLIO/S Tape Management	177
8.1 Overview	177
8.2 Benefits	178
8.2.1 Flexibility	178
8.2.2 Performance	179
8.2.3 Transparency	179
8.3 Functions	179
8.3.1 The Tape Server	180
8.3.2 Fast Data Transfer	180
8.4 Typical CLIO/S Environments	181
8.4.1 Accessing Tapes with the Tape Server	181
8.4.2 Accessing Tapes with the Tape Pseudo Device Driver	183
8.5 Supported Hardware and Software	185
8.5.1 Mainframe to RISC System/6000 Workstation Data Transfer	185
8.5.2 RISC System/6000 to RISC System/6000 Data Transfer	187
8.6 Configuring CLIO/S for AIX Workstations	188
8.6.1 How CLIO/S Works	188
8.6.2 Basic Configuration	189
8.6.3 Creating and Customizing CLIO/S Configuration Files	190
8.6.4 The CLIO/S PARSRV Server	192
8.6.5 Ensuring Password Security	195
8.7 Configuring and Managing the CLIO/S Tape Pseudo Device Driver	195
8.7.1 Configuring and Starting the CLIO/S Tape Mount Daemon	196
8.7.2 Monitoring Tape Mount Daemon Activity	199
8.7.3 Stopping the CLIO/S Tape Mount Daemon	200
8.7.4 Changing the CLIO/S Tape Pseudo Device Driver	200
8.7.5 Reconfiguring the Tape Pseudo Device Driver and Restarting the CLIO/S Tape Mount Daemon	201
8.7.6 Removing the CLIO/S Tape Pseudo Device Driver	202
8.8 Administrative Commands	202
8.8.1 cltclntd—Starting the CLIO/S Tape Mount Daemon	202
8.8.2 cltterm—Stopping the CLIO/S Tape Mount Daemon	203
8.9 End-User Commands	203
8.9.1 Accessing CLIO/S Tapes with AIX Commands and Subroutines	203
8.9.2 cltalloc—Allocating a CLIO/S Tape Device	205
8.9.3 cltunall—Unallocating a CLIO/S Tape Device	207
8.10 CLIO/S Positioning	207
Appendix A. ADSM Supported Tape Drives and Libraries	209
Appendix B. Records at the Beginning of a Labeled Tape	219

Appendix C. Single Drive Reclamation Process	221
Appendix D. NetTape Supported Tape Drives and Libraries	235
Appendix E. ADSM for AIX StorageTek Tape Library Support Services Solution	237
Appendix F. NetTAPE Hints and Tips	239
F.1 Using tar with Labeled Tapes	239
F.2 Using tar, cpio, dd, and backup/restore with Multivolume Data Sets	240
F.2.1 Without a Labeled Tape (NL)	240
F.2.2 With a Labeled Tape (AL or SL)	240
F.3 Using the backup and restore Commands	241
F.4 Using the cpio Command	242
F.5 Using DB2 PE Database backup and restore Commands	242
Appendix G. Scripts	243
G.1 Automatic Labeling of a Library	243
G.2 Automatic Checkin of Volumes into a Library	244
Appendix H. IBM Tape Drive and Library Attachment	245
H.1 Internal Tape Drives	245
H.2 External Tape Drives and Libraries	245
H.2.1 SCSI Attachment	245
H.2.2 Parallel Channel and ESCON Attachments	248
H.2.3 7331 and 3590-B11 Tape Library Considerations	249
Appendix I. Product Levels	251
I.1 AIX	251
I.2 ADSM	251
I.3 NetTAPE	251
Appendix J. Special Notices	253
Appendix K. Related Publications	255
K.1 International Technical Support Organization Publications	255
K.2 Other Publications	256
How To Get ITSO Redbooks	261
How IBM Employees Can Get ITSO Redbooks	261
How Customers Can Get ITSO Redbooks	262
IBM Redbook Order Form	263
List of Abbreviations	265

Index 267

Figures

1.	Single AIX System with SCSI Tape Device	3
2.	Single AIX System with Several SCSI Tape Devices	4
3.	Single SCSI Tape Device Connected to Several AIX Systems	4
4.	Multiple UNIX Systems with Multiple SCSI Tape Devices	5
5.	Mixed MVS and AIX Tape Environment	6
6.	List All Defined Tape Drives SMIT Panel: Tape Devices and Drivers Panel	26
7.	ADSM and AIX Command Use of Tape Drivers	27
8.	SCSI Library Components	32
9.	Library Device Drivers	34
10.	Two Hosts Accessing One Tape Device by Sharing the SCSI Bus	35
11.	Two Hosts Accessing One Twin-Tailed Tape Drive	36
12.	Sharing Two Twin-Tailed Drives between Two Hosts	37
13.	ADSM Server Structure	43
14.	Damaged Storage Pool Volume Recovery	59
15.	Single Drive Reclamation Process	65
16.	Sample NetTAPE Configuration	90
17.	NetTAPE GUI Mount Message	92
18.	NetTAPE Pool Example	96
19.	Mount Queue Management Window with Device Pools	97
20.	Managing a Mount Queue for Device Pool DEV3490	98
21.	NetTAPE Daemons and Their Associated Files and Logs	110
22.	Sample NetTAPE Configuration	114
23.	NetTAPE Operator GUI Mount Message Area	130
24.	NetTAPE Operator GUI for Deleting a Device	135
25.	Drive Sharing with ADSM and NetTAPE: Two Hosts	156
26.	Drive Sharing with ADSM and NetTAPE: with an SP2	157
27.	Migrating an Existing ADSM Server to Use an External Library	164
28.	NetTAPE GUI after an ADSM Mount through the External Manager	168
29.	Accessing MVS Tapes from AIX with the Tape Server	182
30.	Accessing AIX Tapes from AIX or MVS with the Tape Server	183
31.	Accessing MVS Tapes from AIX with the Tape Pseudo Device Driver	184
32.	Accessing AIX Tapes from AIX with the Tape Pseudo Device Driver	185
33.	Sample CLIO/S Configuration	190
34.	Adding a Remote Tape for CLIO/S	197

Tables

1. Tape Drive Special File Characteristics	22
2. NetTAPE Components and Code Packages	108
3. When to Use cltvols	144
4. Configuration Files Used by CLIO/S	190
5. Programs for Starting CLIO/S	193

Preface

This redbook provides a comprehensive guide to AIX tape management. Until recently in UNIX environments, magnetic tapes were used primarily for product installation and image backups of the system. Today, UNIX systems are used more and more for mission-critical applications, and more and more important data is stored these systems. Tapes, drives, and libraries, and tools for managing them, are required to handle the data.

Powerful facilities for managing tapes have been developed over the years in the mainframe environment. These facilities are now finding their way to the UNIX environment.

We describe the facilities provided in the native AIX operating system and the additional facilities provided by ADSM, NetTAPE, and CLIO/S. ADSM provides facilities such as data management, tape library support, and ADSM tape volume inventory management. NetTAPE provides facilities such as remote tape access, centralized operator interfaces, and tape drive and library sharing among applications and systems. CLIO/S provides facilities such as transparent sharing of tape data among applications running on an MVS platform and multiple AIX systems.

This redbook is written for system, storage, and UNIX administrators who want to learn more about managing tapes in an AIX environment.

How This Redbook Is Organized

This book is organized as follows:

- Chapter 1, "Tape Management Overview"

In this chapter we discuss how and why customers use tapes in an AIX environment. We provide an overview of:

- UNIX tape environments and applications
- Tape usage considerations
- How AIX supports tapes
- How ADSM manages tapes
- The functions that the new NetTAPE product provides
- The functions that CLIO/S provides

- Chapter 2, "AIX Tape Management"

In this chapter we look at tape support in UNIX in general. We describe how tape devices are known, accessed, and used in a standard UNIX system without any special tape management software. We then discuss the different types of tape drives and tape libraries that are available today and compare device sharing techniques. We also look at how people use tapes in UNIX and explain the trends that are triggered by the move of mission-critical applications to UNIX, advances in tape technology, and problems presented by distributed computing environments.

- Chapter 3, "ADSM Tape Management"

In this chapter we look at how ADSM manages tape drives, libraries, and volumes. We discuss how best to manage an ADSM server with respect to tape operations. We also discuss how drives and libraries can be shared between ADSM and other tape applications running on the same or different hosts.

The topics covered in this chapter are:

- ADSM structure
- ADSM tape device support
- ADSM tape media management
- ADSM tape operation
- ADSM device sharing
- ADSM tape performance

We describe the tape usage considerations that will enable you to maximize ADSM. We do not duplicate the information that is available in the formal ADSM manuals, so please refer to them for command syntax and general functional descriptions.

- Chapter 4, “NetTAPE Overview”

In this chapter we provide an overview of the NetTAPE product. The topics we cover are:

- Objectives
- NetTAPE functions, including what is supported, what is not supported, and when you should use NetTAPE
- Technical details
- Hardware and software platforms supported

- Chapter 5, “NetTAPE Installation and Configuration”

In this chapter we describe NetTAPE planning, installation, and configuration. We provide a checklist of the steps to take and decisions to make when planning your NetTAPE environment. We give an overview of NetTAPE installation; we do not try to duplicate the formal product install books. We provide an overview of the NetTAPE daemons and their associated files and logs. We describe configuring NetTAPE, in particular, the four configuration files. We conclude with the additional steps required to install and configure tape libraries.

- Chapter 6, “NetTAPE Administrator Operations and End-User and Application Use”

In this chapter we describe some of the common administrator operations that use the NetTAPE command line interface and GUI. We also describe how an end user and application would use NetTAPE from a command line.

- Chapter 7, “ADSM Tape Management with NetTAPE”

In this chapter we discuss how to integrate ADSM with NetTAPE by using the new External Library facility in ADSM. We explain how the External Library works, its advantages, and what you must do to make it work.

The topics we cover are:

- The ADSM External Library and NetTAPE ADSM External Manager
- Configuring the ADSM External Library for use with the NetTAPE ADSM External Manager
- Managing an ADSM NetTAPE environment

- Chapter 8, “CLIO/S Tape Management”

In this chapter we describe CLIO/S support for tape management. Several books document how AIX workstations can use CLIO/S to access tape devices on MVS. Therefore in this book, although we provide an overview of all of the CLIO/S functions including MVS tape access, we discuss only AIX-to-AIX tape access in detail. Note that NetTAPE replaces CLIO/S as the product of choice for AIX-to-AIX environments.

The CLIO/S topics we cover are:

- Overview
- Benefits
- Functions
- Typical CLIO/S environments
- Supported hardware and software
- Configuring CLIO/S for AIX workstations
- Configuring and managing the tape pseudo device driver
- Administrative commands
- End-user commands
- CLIO/S positioning

- Appendix A, “ADSM Supported Tape Drives and Libraries”

The ADSM for AIX server supports a wide range of tape drives and libraries. We include the latest list at the time of writing. Please check the ADSM server README file, contact your IBM representative, or access the ADSM home page on the World Wide Web (<http://www.storage.ibm.com/adsm/>) for the most current product information.

- Appendix B, “Records at the Beginning of a Labeled Tape”

We describe the three EBCDIC-encoded 80-byte records at the beginning of a labeled tape. We also describe each field for IBM standard labels.

- Appendix C, “Single Drive Reclamation Process”

We provide a copy of a script that performs single drive reclamation. It is provided as-is with no support. It can also be found on the ADSM ftp server as SDRECLM AIXBIN. You can download the file in binary to a file named sdreclaim.

- Appendix D, “NetTape Supported Tape Drives and Libraries”

We provide a list of the tape drives and libraries supported by NetTAPE when they are connected to an AIX workstation.

- Appendix E, “ADSM for AIX StorageTek Tape Library Support Services Solution”

In this appendix we describe the StorageTek solution that enables you to extend the use of automated tape operations to the local area network through ADSM and the StorageTek tape libraries. Tape libraries can be effectively shared among RISC System/6000 or SP2 servers, or with an MVS host. The tape drivers provided support StorageTek tape drives in native mode and do not emulate IBM 3490 tape drives. Support of StorageTek tape drives in native mode allows the use of mixed tape media in the same library and the accurate reporting of drive types in reporting systems.

The software includes a set of utilities and features that interact with StorageTek’s ACSLS server or Library Station software on an MVS host. The utilities query the status of volumes, drives, and scratch pools and allow for the mounting and dismounting of tape volumes, the definition of scratch pools and the assignment of tape volumes to scratch pools. The *auto checkin* feature enables you to check large numbers of tape volumes into the library at one time.

- Appendix F, “NetTAPE Hints and Tips”

In this appendix we provide hints and tips for using NetTAPE with the tar, cpio, dd, and backup and restore AIX backup commands.

For specific information about using NetTAPE, please see the *IBM Network Tape Access and Control System for AIX* and *IBM NetTAPE Tape Library Connection User’s Guide and Reference* in the */usr/lpp/nettape/doc* directory.

The topics we cover are:

- Using tar with labeled tapes
- Using tar, cpio, dd, and backup and restore with multivolume data set
- Using the backup and restore commands
- Using the cpio command
- Using DB2 PE database backup and restore commands

- Appendix G, “Scripts”

In this appendix we present two scripts. The first script provides automatic volume labeling using an automatic library. The second script provides automatic checkin of volumes into a library.

- Appendix H, “IBM Tape Drive and Library Attachment”

In this appendix we include some details about attaching tape drives and libraries to a RISC System/6000. We look at internal and external tape drives and external tape libraries. Note that not all RISC System/6000 models support all of the devices we describe.

- Appendix I, “Product Levels”

In this appendix we list the AIX, ADSM, and NetTAPE product levels used in our environment.

The Team That Wrote This Redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization (ITSO), San Jose Center.

Cyndie Behrens is a Distributed Storage Software Specialist at IBM's ITSO, San Jose Center. She writes and teaches IBM classes worldwide on all areas of ADSM. Cyndie has written extensively on ADSM Version 2, ADSM backup of databases (such as Oracle, Sybase, DB2, Informix, and Lotus Notes), and OS/2 LAN and Warp Servers. Before joining the ITSO in 1993, Cyndie worked in software marketing and technical support in Los Angeles, California as a client/server specialist. She holds degrees in mathematics and psychology from the State University of New York at Binghamton.

Nick Higham is currently an AIX specialist at IBM UK. He holds a degree in computer science and electronics from Swansea University in Wales and has been with IBM for the past eight years. Nick has worked in a variety of areas including networking, PCs, and AS/400 in a branch office and AIX country technical support. For the past two years he has been working at the ITSO in Austin with responsibilities for storage products in general, and ADSM in particular. His areas of expertise also include programming and graphics.

Charles Silvan holds a degree in physics from the Swiss Federal Institute of Technology in Zurich, Switzerland, and a Masters Degree in Computer Science from Union College, Schenectady, NY. Charles has been with IBM for 26 years, including 3 years at the World Trade Systems Center in Poughkeepsie. He has worked in a variety of areas including MVS, TPNS, VM, and document processing. Charles is presently working in services in Geneva where he is responsible for storage projects at CERN, and for ADSM installations, mostly under AIX, in the French part of Switzerland.

Lilian Friedrichkeit is an AIX specialist at IBM Austria. She has studied computer science at the Technical University of Vienna and has been with IBM for the past 6 years. Lilian has worked in a variety of areas including networking, VM, marketing, and AIX country technical support. Her areas of expertise include SP2, ADSM, HACMP, networking, and storage in general.

Geoff Buss is currently a storage systems management specialist at IBM UK. He is studying for a degree in social sciences, majoring in psychology. Geoff has been with IBM for the past 18 years and has worked in a variety of jobs including production application support, capacity planning, storage management, and 3 years with IBM's Storage System Division in the U.S. in

information development. Since 1991, Geoff has been providing storage management services ranging from planning and strategy to design and implementation for IBM UK customers. His areas of specialization include tape management, storage administration process management, and storage management planning and design.

Jean-Claude Delft is a Advisory Systems Engineer at IBM Belgium. He has 29 years of experience in IBM, including 6 years specializing in AIX. Jean-Claude graduated from Mons University with a degree in mechanical engineering. He has worked in a variety of jobs, such as a hardware and software customer engineer and an MVS, VM, and networking technical professional. Jean-Claude's areas of expertise include design, setup, and performance of the RISC System/6000 and its associated storage devices and software.

Thanks to the following people for their invaluable contributions to this project:

Steve Atkins
IBM UK

Bob Colao
IBM Poughkeepsie

Rong Lee
IBM Poughkeepsie

Dave Minch
IBM Tucson

Greg Van Hise
IBM Tucson

Ellen Waldo
IBM Tucson

Andy Warton
IBM UK

Comments Welcome

We want our redbooks to be as helpful as possible. Should you have any comments about this or other redbooks, please send us a note at the following address:

redbook@vnet.ibm.com

Your comments are important to us!

Chapter 1. Tape Management Overview

Magnetic tapes are difficult to use. In comparison to magnetic disks, they are inconvenient for these reasons:

- Only one user or application can access a tape drive at a time.
- When you are reading a tape, you must know its characteristics (for example, block size, data format, number of files) to make sense out of it.
- Tape volumes come in two sizes: too small and too large! When tapes are too small, you must manage volume switching; when tapes are too large, you waste space.
- It is very easy to overwrite someone else's tape, so someone could very easily overwrite your tape!
- Someone must physically store tape volumes somewhere and ensure that the correct volume can be found again when needed.
- Tape drives are complicated mechanical devices that tend to require frequent maintenance.
- There are many types of cartridges. Unless your installation has many types of drives, you cannot easily exchange cartridges with other people.

Storing data on tape is cheap, however. It has always been about 10 to 100 times cheaper to store data on tape than on disk, and there is no reason why this will change in the future.

Magnetic tapes are also very good at transmitting data from place to place. A box of cartridges sent by overnight express mail provides a much higher bandwidth than any network you can buy, and at a fraction of the cost.

Tapes are also ideal for storing data that you access infrequently. Why keep data spinning on your disks if it is going to be read by someone only once a year?

People use magnetic tapes for all of these reasons, and will continue to do so because there is no other media on the horizon that can really replace it.

Until recently in traditional UNIX environments, magnetic tapes were used primarily for product installation and image backups of the system. Today, UNIX systems are used more and more for mission-critical applications, and more and more important data is stored on them. Data that was previously on mainframes is now being moved to UNIX systems. This data has to be

backed up, space on disk must be managed so that the inactive data is migrated to tape, and applications have to process large amounts of data. Tapes, drives, and libraries, and tools for managing them, are required for all of those purposes.

Powerful facilities for managing tapes have been developed over the years in the mainframe environment. Such facilities include tape catalogs, system functions for allocating drives, and operator interfaces. If you want to have the same level of service under AIX, you need the same kind of facilities.

It would seem at first sight that all you have to do is convert the mainframe facilities and methodologies to UNIX, because they already exist. However, it is not quite that simple. Unlike mainframes, UNIX systems are typically used in a networked environment—which presents a whole new set of challenges for both connectivity and management.

We discuss in this chapter:

- UNIX tape environments and applications
- Tape usage considerations
- How AIX supports tapes
- How ADSM manages tapes
- What functions the new NetTAPE product provides
- What functions are provided by CLIO/S

1.1 UNIX Tape Environments and Applications

UNIX systems, high-performance personal workstations a few years ago, are now used more and more as collections of connected systems responsible for providing enterprisewide services.

Enterprisewide services are often provided on several, different UNIX systems, usually from several suppliers. Configurations are very dynamic, with systems being added regularly to provide new applications or additional capacity.

All of these systems and applications require magnetic tapes, some more than others. There are two basic approaches to meeting tape requirements in a distributed systems environment:

- Install tape drives on every system
- Provide centralized tape facilities

Installing tape drives on every system is an appropriate approach as long as the number of systems involved is reasonably small and tape use is not too heavy. When you start having hundreds, or even only tens, of small tape drives on as many systems, however, with everyone mounting and storing data on tapes, this approach breaks down because of cost, reliability, and productivity factors.

Providing centralized tape facilities allows for economies of scale. The centralized facilities include hardware, because the tape drives and libraries are now a global resource pool, as well as human resources, because operations and management can be organized much more efficiently.

Below we review and illustrate some common tape environments. Figure 1 shows a simple arrangement consisting of a single UNIX system, designated as a tape server, with a SCSI tape device attached. The tape device is attached locally, and it is either an internal or external stand-alone tape drive or a tape library containing a robotics accessor mechanism (that is, a single drive with storage slots for 10 or more tape cartridges).

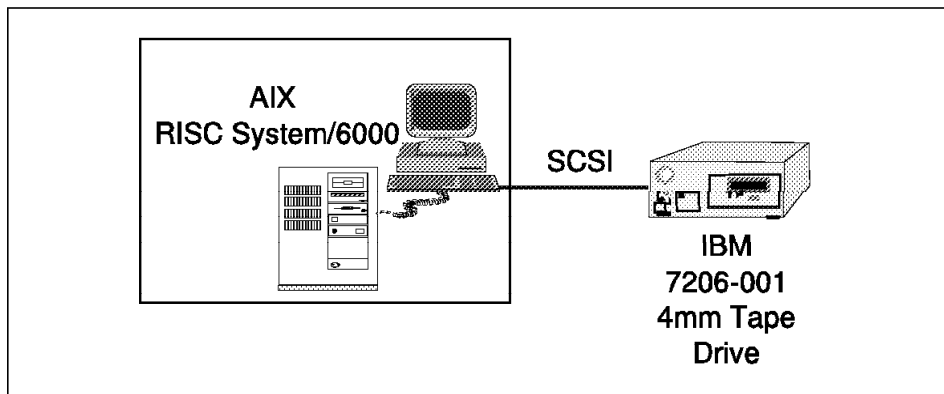


Figure 1. Single AIX System with SCSI Tape Device

Figure 2 on page 4 depicts a single system acting as a tape server, with several SCSI tape devices attached. The tape devices could be a combination of external tape drives or a tape library with several drives.

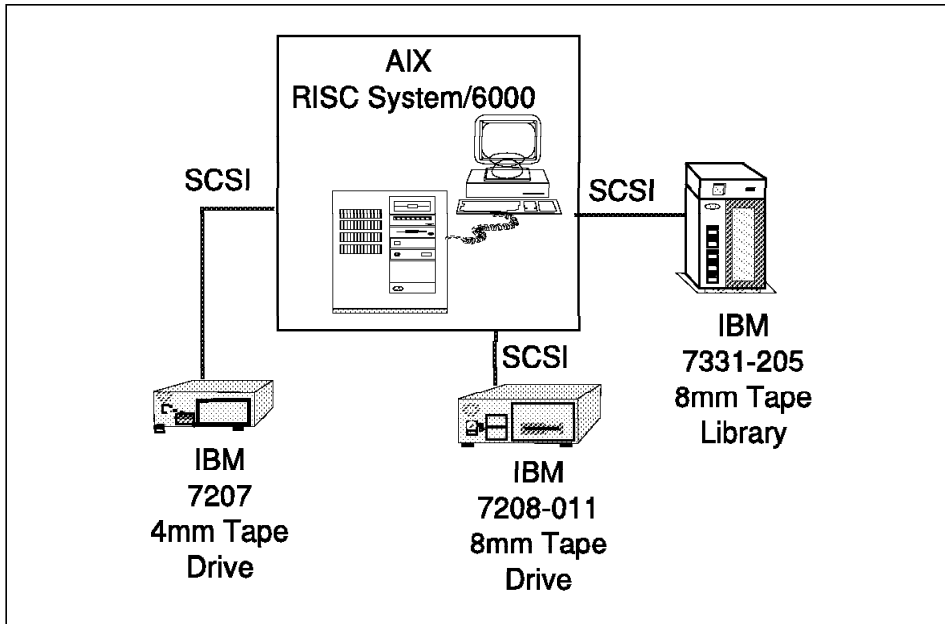


Figure 2. Single AIX System with Several SCSI Tape Devices

Figure 3 shows a single tape device connected to several systems, one of which is designated as a tape server. The tape device is local to two of the systems and therefore remote to the others. The device is most probably a tape library.

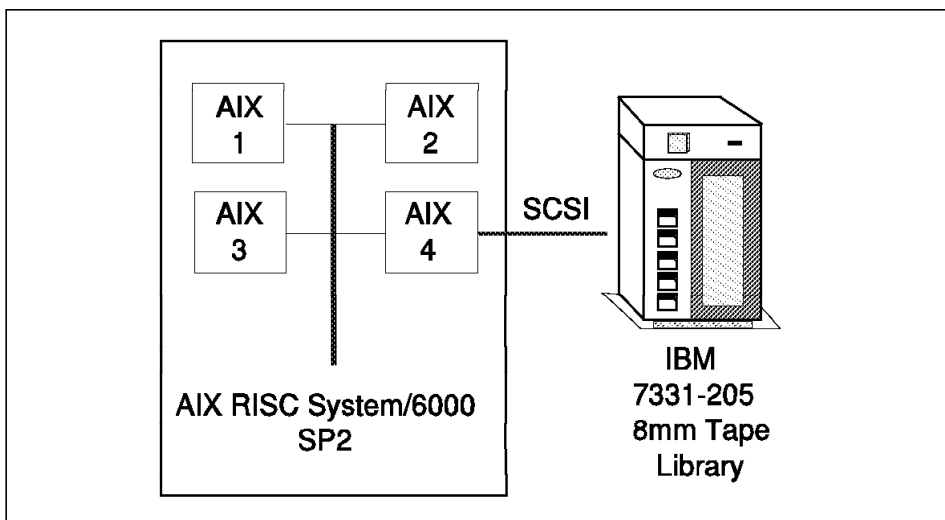


Figure 3. Single SCSI Tape Device Connected to Several AIX Systems

Figure 4 on page 5 depicts multiple systems, some of which are designated as tape servers, with multiple SCSI tape devices that are locally attached. The devices are most probably a combination of external tape drives and one or more tape libraries.

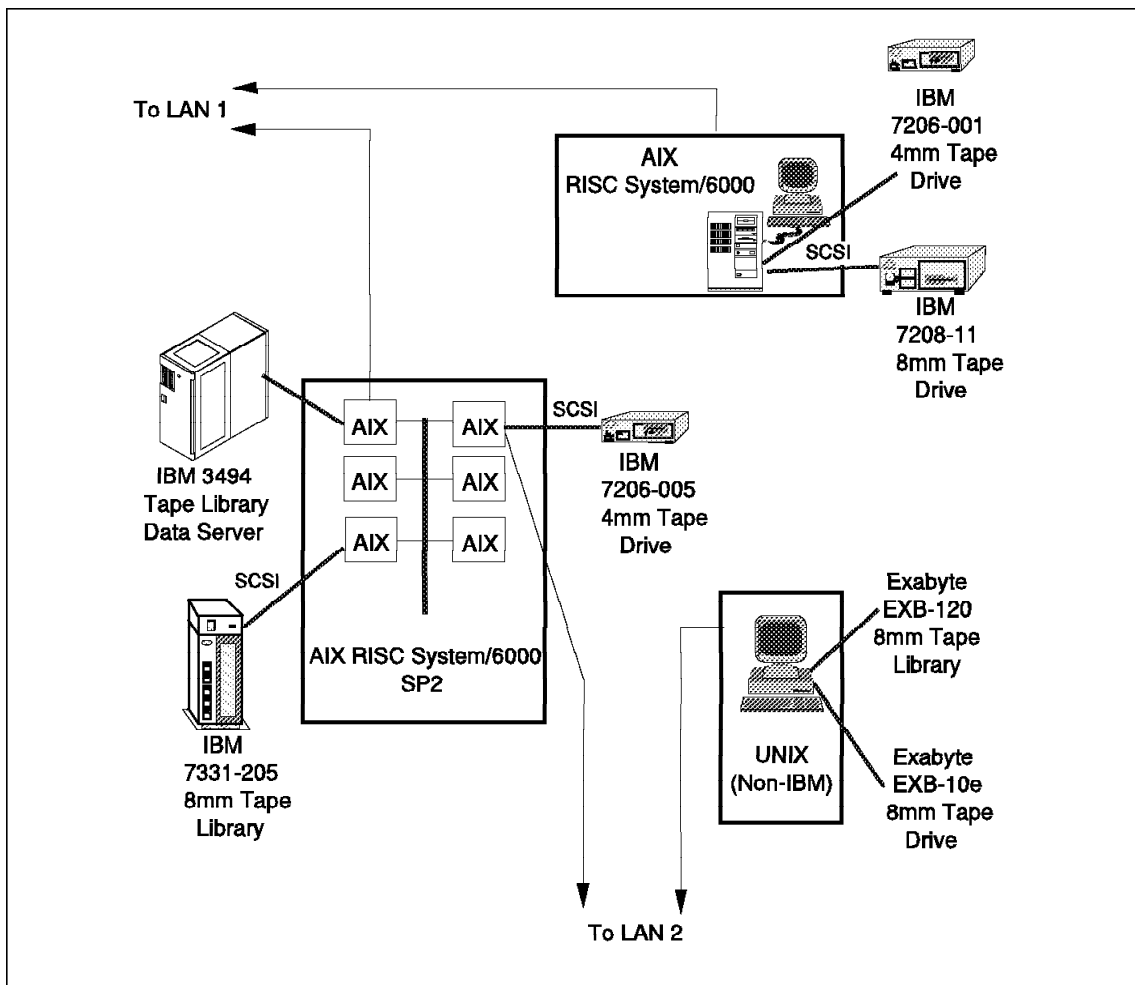


Figure 4. Multiple UNIX Systems with Multiple SCSI Tape Devices

Figure 5 on page 6 shows an MVS mainframe with 3480, 3490, and 3590 tape drives. The 3490 devices might be in a 3494 or 3495 Automated Tape Library Dataserver (ATLDS). There are one or more AIX systems, some of which are designated as tape servers, with one or more SCSI tape devices locally and remotely attached.

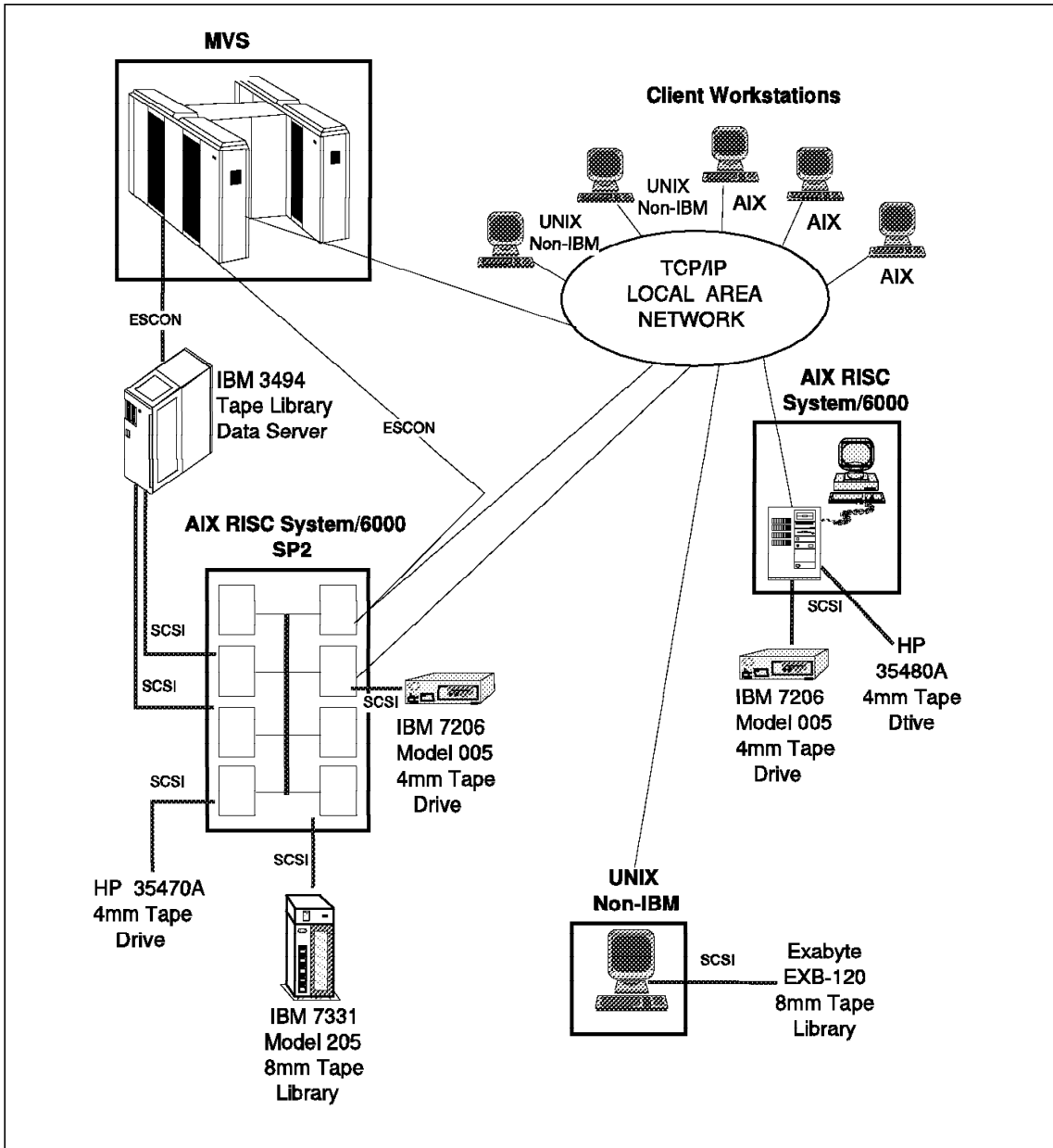


Figure 5. Mixed MVS and AIX Tape Environment

Some of the 3490 drives in the 3494 ATLDS might also be connected to one or more of the AIX systems. Perhaps some of the tape data is accessed from both AIX and MVS applications.

Centralization of tape resources, however, is possible only if the system or application software provides the necessary functions. Such functions include:

- Tape volume control (inventory, ownership, and access rights)
- Allocation of tape drives
- Centralized (manual) mount interface for operators
- Remote access to tape drives or tape data

All of the above functions are missing in UNIX operating systems and therefore must be provided by additional software products, such as NetTAPE, which are just starting to appear in the marketplace.

Let us now look at the main types of tape applications found on mainframes as well as on distributed systems:

- Backup

Backup tapes are used to store copies of files so that you can restore the files if the original copy is corrupted or erased by mistake or the disk on which they reside becomes unavailable.

It is safe to say that everyone has to do backups.

- Archive

Archive tape storage is used as long-term storage for files that have been deleted from disk or for which you must keep copies from different dates.

Archive, like backup, is almost a universal need.

- Batch

Batch tape storage is used as input or output for batch application jobs and is typically used in scientific environments.

- Data interchange

Data interchange tape storage is used for storing data on tape cartridges to be sent to another location. It may also include encoding so that tapes can be read on a different platform when they are moved. For example, encoding can be used when moving from an ASCII to an EBCDIC platform, or, for binary data, when moving from a big-endian to a small-endian platform. Data interchange tape storage is also a universal need.

- Hierarchical storage management

Hierarchical storage management (HSM) tape storage is used as the last level in a storage hierarchy; files that are infrequently accessed are migrated from disk to tape.

HSM is a growing market because of the increasing amounts of data that must be managed. Web servers are also a good opportunity for HSM implementations. HSM is also sometimes referred to as *space management*.

- Data acquisition

Data acquisition tape storage is used for storing data coming directly from measurement equipment for later analysis (for example, satellite data). This tape storage is typical in scientific environments.

- Digital libraries

Digital libraries tape storage uses sophisticated techniques to exploit very large quantities of data, for example, data mining. The data can be textual data, images, or streams of video or audio. The tape storage is used as the last level in the storage hierarchy and/or as archive and backup.

Digital libraries storage is a fast growing market that will likely drive the future of tape technology.

1.2 Tape Usage Considerations

There are many factors to consider when using tape devices:

- Media type and reliability
- Recording density
- Tape capacity
- Performance
- Manual and automated tape devices
- Device connectivity
- Media movement between storage locations
- Media operation
- Application usage
- Business benefits

1.2.1 Media Type and Reliability

In a UNIX environment there are a variety of tape devices with different service lives and reliability. For example, the recommended service life of an 8 mm tape cartridge is approximately 400 passes over any one area on the tape media, whereas the life expectancy of a 4 mm cartridge is approximately 500 passes. However, this lifespan can be shortened by leaving the tape in a drive for periods longer than 7 hours. Therefore, always ensure that tapes are replaced regularly, by backing up the data to a new cartridge, and that cartridges are dismounted and stored properly after use.

The drives themselves also have recommendations for use. For example, you should clean your 8 mm tape drives at least once a month or every 30 hours of read and write activity, whichever occurs first. If you use an inappropriate cleaning cartridge, such as a video cleaning cartridge, a drive can be permanently damaged.

1.2.2 Recording Density

The tapes you use in your UNIX environment have different recording densities. The length of the tape media and the density at which data is recorded on the tape media decide the capacity of the tape cartridge. Different tape devices support different densities and hence different capacities. The same device may provide the capability to record in several densities, usually for backward compatibility with earlier models.

Let us say you have an old tape cartridge that needs replacing. You want to copy the data from the old tape to a new tape cartridge with a larger capacity than the cartridge you are replacing, so you specify a higher density setting for the new tape. You must ensure that the new tape can be used on existing devices and is still compatible with the application that is to use the tape data.

1.2.3 Tape Capacity

Tape cartridges have different capacities. Even within the same family of tape devices there can be variations in capacity. For example, a 4 mm digital data system (DDS) “four-stripe” tape can hold 2 GB of data, whereas a 4 mm DDS2 tape has a capacity of 4 GB. Therefore, when you write large files to a 4 mm device, it is more advantageous to use the DDS2 tape than the DDS four-stripe tape. If you write the files to the 2 GB tape, you might need a second cartridge, and additional time would be needed to dismount the first cartridge and mount the second cartridge. We suggest that, whenever possible, you try to match tape devices to the most appropriate and effective use of their capacity.

Hardware compression, provided on some tape devices, enables you to make better use of tape capacity by storing data in a compressed format. Applications are saved from having to do the compression because the hardware does the compression.

1.2.4 Performance

The performance of your tape devices can be affected by the size of the blocks of data written to the tape. Every tape device has a maximum or peak data transfer rate that defines the maximum speed that data can be written to the physical media. It is important to note that this rate is the peak rate for data being streamed to the device. Therefore, if an application using a tape device writes to the tape in small blocks of data, the average data transfer rate is probably going to be slower than the maximum capability of that device.

The block size used can affect tape device performance in other ways too. The block size defined to the operating system for a tape device will be the size of a block of data sent to the device driver for the device on a write operation or read from the device driver on a read operation. The tape drive itself, however, may use a completely different block size. Some 8 mm tape devices physically write data to the media in blocks of 1024 bytes, for example. If the block size set by the application or operating system is not a multiple of 1024 bytes, however, data will have to be fragmented and reassembled during the read and write processes, thereby reducing performance.

Another performance factor is tape drive technology (see 2.2, “Tape Technology” on page 27), which includes functions on high-end tape drives (for example, the IBM 3590), such as a serpentine architecture and high-speed seeks to specific blocks.

1.2.5 Manual and Automated Tape

You probably have manual, stand-alone, internal and external tape devices as well as automated tape libraries in your UNIX environment. Single, manual drives require tape cartridges to be loaded and unloaded one at a time by hand. The tape libraries have one or more drives, contain a number of tape cartridges stored in slots in a removable cassette, and use a robotics accessor mechanism to mount (load) and dismount (unload) tapes in the drives.

Unless there are attended tape operations overnight, manual tape devices are far more likely to be used during working hours when people are available to load and unload tapes. Overnight, therefore, they are unlikely to be used unless a tape is preloaded and used with a job started by cron or

at, for example. However, if a manual tape drive is used overnight, the tape may remain in the drive, before or after use, for more than 7 hours, which, as we note in 1.2.1, “Media Type and Reliability” on page 9, can shorten the service life of the tape. Also, an unauthorized person can take the tape from the drive while it is waiting to be unloaded. Manual devices are slower to use (their performance may be good, but this is countered by the manual loading and unloading of tapes), are not well utilized, and are a security risk if they are not located in a secure area.

Automated tape libraries clearly have a distinct advantage over manual tape devices. To begin with, you can leave automated tape libraries unattended for long periods of time because their robotics load and unload tapes in the drives. In addition you can physically lock the access door, making the tapes relatively secure. You can use an automated tape library for more than one application at a time, because there is usually more than one drive in the library. Automated tape libraries are well suited for use with an application like ADSM.

1.2.6 Device Connectivity

You will want to provide alternative access to your tape devices when some systems are unavailable. In certain situations, however, connectivity to your tape devices requires careful consideration, in particular, when sharing devices between systems or sharing devices between applications on the same system. The configuration of the drives, the device drivers, SCSI cable connections, and internal buses and ports govern the level of sharing and the type of access you can achieve. For a detailed discussion of the various configurations involved in device sharing, see 2.4, “Tape Device and Library Sharing” on page 34.

1.2.7 Media Movement between Storage Locations

In many cases, tape is used for storing an application backup file set that is used for disaster recovery purposes or vital records. The implication is that the tape media is physically moved from the tape device to a firesafe, usually kept onsite, or an offsite vault. Sometimes the tape media may be moved to a firesafe first, until another version of the backup file set is created, before it is moved to the offsite vault, where it is kept for a period of time. Later, when the tape returns onsite, it is added to a “scratch” pool for reuse. Currently, you have to have a manual logging procedure to track the movement of tape media. Until tape media management tools become available for the UNIX environment, there is no simple or cheap solution to better track the movement of tape media.

The Disaster Recovery Manager (DRM) feature of ADSM tracks the movement of ADSM tape media. See 3.4.5, “Backing Up ADSM” on page 69 for more details.

1.2.8 Media Operation

Manual, stand-alone tape devices require someone to load and unload tapes in the drive. If your UNIX environment has a large number of manual drives, you need a person to continually load, unload, and clean the devices. From the point of view of scheduling or use of a person’s time, this is not very productive or practical. The alternative is to have quite a few people (generally users) “operating” the manual devices. Again, this is not a very productive use of your users’ time.

As discussed in 1.2.5, “Manual and Automated Tape” on page 10, automated tape libraries require little manual intervention for their operation. For small tape libraries, for example, the IBM 7331-205, tapes are loaded and unloaded from the library in bulk by means of the cassette or magazine in which the tape cartridges are stored while inside the library. For example, the IBM 7331-205 has two 10-slot cassettes, and so tapes can be loaded and unloaded in groups of 10 or 20 each time.

For the large automated tape libraries, the IBM 3495 ATLDS, for example, tape cartridges are loaded and unloaded through one of several cartridge input/output stations, depending on the quantity of cartridges and the type of operation (loading or unloading). Cartridges can be loaded or unloaded in bulk in quantities of about 200 at each operation.

1.2.9 Application Usage

Applications in your UNIX environment use tape devices in different ways, as noted in sections 1.2.4, “Performance” on page 10 and 1.2.6, “Device Connectivity” on page 11, for example. Consider how best to configure and utilize your tape devices while ensuring that they continue to remain compatible with the applications in your computing environment. In the end, your configurations will most probably be a balance between the needs of your applications and the most effective use of your tape devices.

1.2.10 Business Benefits

Important among your considerations for tape usage are the business benefits that a particular type of tape device provides. This is the best balance available between the cost of the device and the benefits it offers to your company’s business. The main cost elements consist of:

- Purchase price of the device
- Maintenance overhead, such as cleaning the drives, and regular servicing and repair
- Environmentals, such as power consumption and “footprint” requirements
- Amount of a person’s time and effort required to operate the device

The main business benefits are:

- Productivity improvements
- Improvement in processing throughput and performance
- The contribution the device makes toward increased business revenue and profit

1.3 How AIX Supports Tapes

AIX (similar to all of the other UNIX systems) enables you to define and address tape devices but does not provide any tape management. That is, once a drive is available, someone can walk to the drive, insert a cartridge, and issue some AIX commands that will read or write data on this cartridge. This process is fine if you have a tape drive for your own private use, but it is far from sufficient if several tape drives are being used by different users and applications.

The following tape management support is missing in AIX:

- A way of controlling who is given access to a drive at a given time, that is, device allocation
- A tool for knowing who is using which drive
- An interface to tell the operators which cartridge must be mounted where
- Protection against someone writing on a cartridge owned by someone else

AIX provides only very basic device support, which we review in detail in Chapter 2, “AIX Tape Management” on page 19. You will have to install additional products to use tape drives and libraries in a more automated production environment. ADSM and NetTAPE are two such products.

1.4 ADSM Tape Management

In this section we present an overview of the ADSM objectives, functions, and benefits, focusing on issues related to AIX tape management.

1.4.1 Objectives

ADSM, as its name implies, manages distributed storage. Using a client/server architecture, ADSM enables a variety of client nodes to send data to a central server machine, the ADSM server. The ADSM server is responsible for managing the storage devices to which the client data is sent and stored, and it uses a database to keep track of the location of every piece of data. Clients can get their data back when they ask for it.

1.4.2 Functions

Because UNIX operating systems do not provide the necessary functions for managing tapes, the AIX ADSM server provides those functions for its own use.

If you are in a situation where ADSM can satisfy your needs for data storage, you can use ADSM to manage your entire tape and automated library environment to provide your distributed clients with:

- User home directory backup, archive, and space management
- Backup and archive of distributed databases
- Lotus Notes backup and archive
- Backup and archive for specialized applications, such as image and print-on-demand

Some of the functions for which ADSM is not appropriate are data acquisition and data interchange. These applications, if they are relevant to you, have to be processed outside ADSM.

1.4.3 Benefits

Using ADSM to support tapes enables you to get away from a model where a user owns a tape volume. When you use ADSM, user data is managed in a way that makes tape volumes transparent to the end user. Using ADSM has the following advantages:

- Improved tape volume utilization

Studies have shown that tape volumes in most installations are very poorly utilized. Because a large proportion of the cartridges are filled at a very low percentage, large numbers of volumes are purchased, stored, cataloged, and mounted for storing very little data. ADSM tape

volume utilization will remain consistently high because tape volumes are filled to full capacity, and, when data is deleted from these volumes, the space is reclaimed by consolidating several partially filled volumes onto a full volume.

If you are using ADSM's collocation feature on large capacity tapes for small clients, those tapes may not be well utilized. See 3.4.4, "Collocation" on page 67, for details on how to use collocation appropriately.

- Improved tape drive utilization with disk caching

ADSM provides the ability to use disk space as a write cache in front of your tape drives. Disk caching avoids the problem of users holding on to drives for many hours for little usage and therefore provides better utilization of your tape drives.

- Management of the tape volume inventory

ADSM manages its tape volume inventory so you do not have to manage the inventory yourself.

- Improved data recovery

ADSM enables you to selectively create duplicate copies of your backed up data (whether the backed up data is stored on tape or disk) and automatically uses the duplicate copies if the physical volume containing the primary copies is lost or damaged.

- Automated library support

Because ADSM supports and manages automated libraries, it is easy for you to implement tape automation.

- Disaster recovery

ADSM includes functions for disaster recovery that enable you to reconstruct your entire ADSM production environment at a recovery site.

The DRM feature can manage an offsite rotation strategy for ADSM tapes, automatically create a disaster recovery plan for the ADSM server, and store client recovery information in the ADSM database.

1.5 NetTAPE Tape Management

NetTAPE, a recently announced and available IBM product, provides enhanced tape capabilities in the AIX environment. NetTAPE offers a single-system image of tape drives and libraries for operations and end-user access.

1.5.1 Objectives

The main objectives of NetTAPE are to provide:

- Centralized tape operations management
- Transparent end-user and application access to any tape device in a networked environment
- Tape drive sharing among applications, including ADSM
- Facilities for exchanging tape volumes with mainframes

All of the tape and library devices that NetTAPE manages appear as a set of resource pools to operators, applications, and end users, thus providing a single-system image. Please note that, at the time of writing, NetTAPE provides library support for high-end libraries; stand-alone drive support is provided for low-end environments.

1.5.2 Functions

The main NetTAPE functions are:

- Central operation
Tape drives can be managed from one or more operator stations through command lines or a Motif-based graphical interface.
- Remote tape access
An application can access a tape drive connected to a remote system.
- Device allocation
NetTAPE manages the selection and allocation of devices to requesting applications and end users. Thus tape drives can be shared between applications running on either the same or different systems.
- Exchange of tape volumes
NetTAPE's support of tape labels enables you to exchange tape volume data between AIX and mainframe applications.
- ADSM support
NetTAPE provides an interface to ADSM so that tape and library devices can be managed by NetTAPE instead of being dedicated to ADSM. Thus drives can be shared between ADSM servers and/or other applications. ADSM cannot use NetTAPE to access remote devices.

1.5.2.1 Attaching Devices

The tape and library devices that NetTAPE supports can be attached to several hosts yet be accessible from anywhere in the network. Accessibility of devices from anywhere in the network provides a lot of configuration

flexibility. You do not have to provide locally attached devices to all of the systems that may need tape drive access, often for very limited periods of time.

1.5.2.2 Configuring Devices

NetTAPE device configuration support adds another level of management beyond that of the standard AIX configuration functions. NetTAPE provides grouping of devices of the same characteristics into pools. Users and applications can request tape services on a pool basis, rather than having to know and select specific devices characteristics, host attachments, and addresses.

Tape devices are also grouped into domains for distributing tape operations, for example, on a geographical basis.

1.5.2.3 Accessing and Using Devices

Users and applications can request tape volumes to be mounted on any device in the network and then use standard UNIX utilities or tape applications to access these devices as if it were directly attached to the user and application workstations.

1.5.2.4 Managing Devices

The Motif graphical interface provides global device management capabilities that facilitate an optimal utilization of the tape and libraries.

1.5.3 NetTAPE with ADSM

The ADSM External Library, in conjunction with NetTAPE's ADSM External Manager, offers additional functions:

- Drive sharing

Tape drives are not dedicated to ADSM and therefore can be used by other applications running on either the same system or another system (sharing the drives by multiple attachments). In particular, you can share drives between two ADSM servers.

- STK Silo support

NetTAPE supports STK silos and thus allows ADSM to use these tape libraries, which are otherwise not supported in native ADSM.

1.5.4 CLIO/S

CLIO/S provides transparent sharing of tape data among applications running on an MVS platform and multiple AIX systems. This function is important to those customers who share large volumes of tape data and

require reliable, high-speed communications in a distributed computing environment.

Devices and tape operation for the mixed MVS and AIX enterprise are controlled and administered from the MVS platform. Thus customers gain productivity benefits, and management of the tape environment is better centralized.

For an AIX environment without an MVS platform, CLIO/S can still provide sharing of local and remote tape devices between AIX systems. However, communication between AIX workstations is supported only over TCP/IP network connections and therefore performs at the speed of TCP/IP. CLIO/S does not provide dynamic allocation of tape devices when used for AIX-to-AIX workstation configurations.

In fact, for an all AIX environment, NetTAPE should be used instead of CLIO/S because it provides similar tape sharing services at a substantially lower cost and can coexist with CLIO/S in a mixed environment.

CLIO/S is best suited for large tape environments that include an MVS platform which wants to share tape and data with AIX systems. High-speed access over ESCON or block multiplex links is required.

Chapter 2. AIX Tape Management

In this chapter we look at tape support in UNIX in general. We describe how tape devices are known, accessed, and used in a standard UNIX system without any special tape management software. We then discuss the different types of tape drives and tape libraries that are available today and compare device sharing techniques. We also look at how people use tapes in UNIX and explain the trends that are triggered by the move of mission-critical applications to UNIX, advances in tape technology, and problems presented by distributed computing environments.

2.1 Tape Support in UNIX

In this section we explain:

- How tape devices are known to UNIX
- Device drivers
- Writing to and reading from files on tape
- AIX tape commands and programming interfaces
- Hardware tape drivers
- ADSM tape drivers

We conclude the section by summarizing tape device drivers through illustrative examples.

2.1.1 How Tape Devices Are Known to UNIX

Tape devices are known to the AIX system as device special files (also known as character special files), for example, `/dev/rmt0.x`, where `x` has different values that are used to control options (such as *rewind on close*). A tape application performs an *Open* to the device, followed by read or write commands, either by directly using a command (for instance, `tar`), or from a C or Fortran program. Tapes are read or written to as if they were UNIX files.

In UNIX there is:

- No mechanism for controlling who is allowed to access the drive, other than the UNIX file permissions
- No function for checking, in advance, whether the drive is ready or busy
- No support for verifying that the correct tape volume is mounted
- No notion of generic devices

In other words, the user has complete control over and responsibility for the drive and volume, even a volume that may already be mounted on the device.

This basic level of support is fine when dealing with one drive that is used essentially for product installation and system backup. However, it is totally inadequate for production use of several tape devices with multiple applications.

Given the shortcomings of UNIX tape support, the pragmatic approach used in most cases is to dedicate drives to different applications, for example, one drive could be dedicated to database logging and other drives to ADSM. Each application handles the tape and drive management at the application level. ADSM is an example of an application that takes control of its drives and libraries and allocates them for its processes as needed. ADSM uses a client/server model that indirectly gives tape access to remote systems and thus can be used very effectively to replace a number of small tape drives on each system with a centralized solution on the ADSM server.

Application-controlled tape management is not a total system solution, however. In particular, it does not enable you to effectively share tape devices among several applications, and it does not provide you with any centralized tape operation support.

Products such as NetTAPE are now appearing to provide the functions needed for a system-managed tape environment that can be used by all of the tape applications in the distributed network. NetTAPE provides central management of tapes drives and volumes through operations control and resource pooling.

2.1.2 Device Drivers

Device drivers are kernel extensions that control and manage specific devices used by the operating system. The I/O subsystem, in conjunction with the device drivers, enables processes to communicate with peripheral devices such as terminals, printers, disks, tape units, and networks. Device drivers can be installed into the kernel to support a class of devices (such as tapes) or a particular type of device (such as a specific tape drive model). Device drivers shield the operating system from device-specific details and provide a common I/O model for accessing the devices for which they provide support. The system interface to devices, which is supported by device drivers, is through the file system.

Each device that is accessible to a user-mode application has a file name and can be accessed as if it were an ordinary file. By convention, this device file name is found in the /dev directory in the root file system. The

device file name and its associated i-node are known as a device special file.

Device drivers can be characterized by the class of I/O subsystem interfaces that they provide. Device driver routines can be further characterized by the roles that they play in supporting the device I/O. Finally, the overall structure of the device driver is dictated by the execution environment in which the routines run.

The operating system supports two classes of device drivers: character and block. These classes are distinguished by the types of devices they support and the interfaces that are presented to the kernel.

Character device interface Suitable for devices, such as tapes, terminals, printers, and networks, that do not have strict definitions of fixed-size addressable data blocks. User-mode access to these character device drivers is through a “character device special file.”

Block device interface Suitable for random access storage devices with fixed-size addressable data blocks.

The ost device driver is the standard AIX tape driver (ost stands for “other SCSI tape”). It provides the basic tape functions found on all tape devices, but it does not support special or advanced features, such as sending a message to the tape display, processing tape labels, or handling multivolume files. Because ost does not support these advanced features, additional tape drivers have been developed to handle specific devices such as the 3490 or 3590 and to provide functions required by products such as ADSM. NetTAPE functions are limited when you use devices supported by the ost device driver. We cover these other tape drivers in 2.1.5, “Hardware Tape Drivers” on page 24 and 2.1.6, “ADSM Tape Drivers” on page 25.

2.1.3 Writing to and Reading from Files on Tape

Writing to and reading from files on tape is done by using rmt special files. The *r* in the special file name indicates raw access through the character device special file. A tape device does not lend itself well to the category of a block device, thus, only character device special files are provided.

Several rmt special files are associated with each tape drive defined to the operating system: /dev/rmt*, /dev/rmt*.1, and /dev/rmt*.2 through . /dev/rmt*.7. The rmt* is the logical name of a tape drive, such as rmt0 or rmt1. When you select one of the special files associated with a tape drive, you are choosing how the I/O operations related to the tape drive will be performed.

Special files associated with each tape device determine which functions are used on the device, such as:

- Rewind on close
- Retention on open
- Density settings
- Trailer label
- Unload on close

Table 1 shows the names of the special files and their corresponding characteristics. For example, /dev/rmt0.2 indicates that you want rewind on close, retention on open, and density setting #1.

Special File Name	Rewind on Close	Retention on Open	Bytes per Inch
/dev/rmt*	Yes	No	Density setting #1
/dev/rmt*.1	No	No	Density setting #1
/dev/rmt*.2	Yes	Yes	Density setting #1
/dev/rmt*.3	No	Yes	Density setting #1
/dev/rmt*.4	Yes	No	Density setting #2
/dev/rmt*.5	No	No	Density setting #2
/dev/rmt*.6	Yes	Yes	Density setting #2
/dev/rmt*.7	No	Yes	Density setting #2

The default block size used on a tape is part of the System Management Interface Tool (SMIT) device definition. Some, but not all, of the AIX tape commands have an option that enables the user to change the block size. However, reading a tape when you do not know the block size used at creation time is a challenging experience: You basically have to keep trying until you hit the right number.

2.1.4 AIX Tape Commands and Programming Interfaces

Let us look at some common administrator and end-user AIX tape commands and tape programming interfaces.

2.1.4.1 Tape Device Handling Commands

The AIX commands listed below are used for tape drive configuration and management. They are used by the system administrator, not by end users. They are commonly issued through SMIT.

cfgmgr	Automatically configures any additional devices that have been added or powered on since the last time the system was started or <code>cfgmgr</code> was run
mkdev	Defines, or defines and configures, a new device in the system, or configures a device that was previously defined
rmdev	Unconfigures a device. The command makes a device unavailable but still defined, or it removes the definition from the object data manager (ODM).
chdev	Changes the attributes of a defined or available device. For example, the following command sets the block size to 1024 bytes: <code>chdev -l rmt0 -a block_size=1024</code>
lsdev	Lists name, availability status, location code, and a brief description, For example: <code>lsdev -Cl rmt0</code>
lsattr	Lists current or default attribute values of a device. For example: <code>lsattr -E -l rmt0</code>
tapechk	Performs rudimentary consistency checking on an attached tape device by reading a tape volume

2.1.4.2 Common End-User and Administrator Tape Commands

The most common AIX end-user commands for working with tapes are:

tctl, mt	Gives subcommands to a streaming tape device to query the status or position the tape (for example, rewind or fast skip forward)
tcopy	Copies a tape to another tape (one file only)
tar	Writes to, reads from, or lists data from tape, commonly used for combining a set of disk files into one tape file or vice versa
dd	Writes to, reads from, or lists data from tape
cpio	Writes to, reads from, or lists data from tape
backup	Writes data to tape and maintains date and time of backed up files and therefore handles incremental backups
restore	Restores data from tape that has been written to the tape with the backup command
rmt	A remote tape access protocol used by the remote backup and restore programs

The most common AIX system administration commands related to tape are:

- mksysb** Makes a bootable backup of the rootvg volume group
- installp** Installs or updates software
- bffcreate** Copies installation images from installation tapes to disk for further usage

2.1.4.3 Tape Programming Interfaces

A program that processes tapes sends orders to the device through a set of `ioctl` commands. `ioctl` commands are library subroutines used to perform control functions associated with open file descriptors. The standard AIX `ost` tape driver supports `ioctl`s for doing basic tape functions only, such as rewind, rewind and unload, tape erase, tape retension, write end-of-file mark, and forward and reverse files and records.

2.1.5 Hardware Tape Drivers

Special drivers have been developed to support tape devices such as the IBM 3490 and 3590 and libraries such as the IBM 3494, 3495, and 7331.

These drivers are referred to as *hardware tape drivers*, because they are delivered with the corresponding devices. Sometimes they are referred to as *SSD drivers* because they have been developed by the IBM Storage Systems Division (SSD) to support the IBM high-end tape devices. The drivers are also referred to as *enhanced AIX device drivers*. We refer to them as *enhanced AIX device drivers* in this book.

Enhanced AIX device drivers are needed so that applications can utilize the full functions of certain devices. These drivers provide additional `ioctl`s beyond the basic ones, such as:

- Get device sense data
- Reserve and release a device
- Query and set block size
- Query and change device, driver, and media parameters
- Display messages on the drive panel
- Query and set tape position

The additional `ioctl`s are necessary for handling 3490 and 3590 devices correctly and taking advantage of all of their functions. For example, the additional `ioctl`, *set and locate blockid*, can be used to provide much faster positioning at a specific point on the tape (rather than having to skip files or records).

There are two sets of enhanced AIX device drivers for IBM 3480, 3490, and 3590 devices:

- The driver that supports SCSI-attached 3490 and 3590s is called `Atape`, and the library driver for the 3494 (when used with SCSI drives) is called `at1dd`.
- The driver that supports ESCON- and parallel-channel-attached 3480 and 3490 drives is structured as a high-level, common driver, `mtdd`, and two low-level drivers, `S390dd` for ESCON attachment, and `S370dd` for parallel attachment.

The IBM enhanced AIX hardware device drivers define the special files for 3480, 3490, and 3590 devices with the usual names, `/dev/rmtx`.

2.1.6 ADSM Tape Drivers

The ADSM server comes with its own tape driver, which provides functions ADSM needs to manage all of the SCSI tape devices, except for the 3480, 3490, and 3590. ADSM uses the AIX enhanced device drivers for the 3480, 3490, and 3590 devices because they provide the necessary level of support.

The `ost` driver defines device files for standard SCSI drives and calls them `/dev/rmtx`. Therefore, ADSM has to use other names for its special device files, `/dev/mtx`. As a consequence, the names of the special device files used by ADSM depend on the type of tape drive used. ADSM tape drives are represented by:

- `/dev/rmtx` for 3480, 3490 and 3590 drives, with any supported type of attachment, that is, the AIX enhanced device drivers are used.
- `/dev/mtx` for all other (SCSI) drives, that is, the ADSM tape drivers are used.

Note that the typical AIX tape commands, for example, `tar` or `dd`, can be used with the ADSM tape driver.

ost Device Driver Conflict with ADSM

An annoying inconvenience with the `ost` device driver is that after the system is rebooted, all of the ADSM device statuses are changed to *defined* and `ost` device special files are generated and set to *available* status. Therefore, after a system reboot on a system with an ADSM server, you must use SMIT to reconfigure the device drivers.

2.1.7 Tape Drivers Summary

You might be confused by all of these different tape drivers. When a device, such as an 8 mm tape, is supported by the ost driver as well as the ADSM driver, you will see two SMIT entries for each physical device. For example, Figure 6 shows a List All Defined Tape Drives SMIT panel.

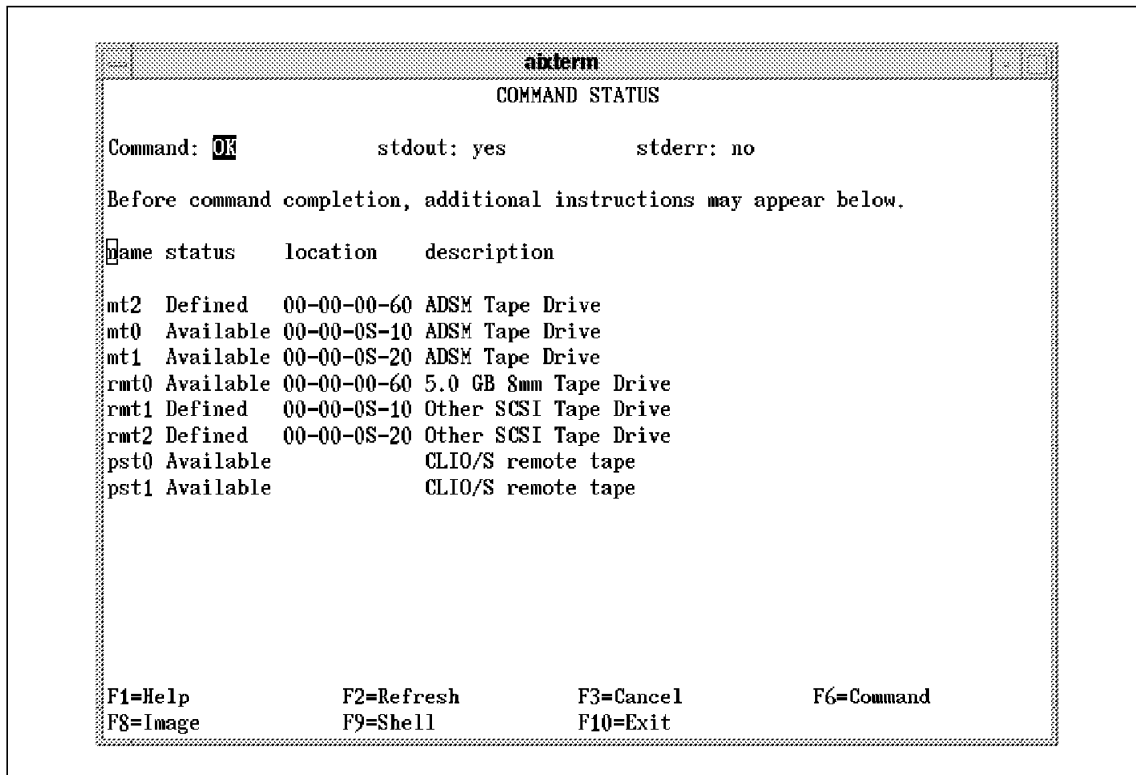


Figure 6. List All Defined Tape Drives SMIT Panel: Tape Devices and Drivers Panel

In Figure 6, three physical drives are defined: two 4 mm external drives, shown as Other SCSI Tape Drive, and one 5.0 GB internal 8 mm drive. As you can see, there is more than one entry for a particular device (as shown in the location column) when it can be used either through the ost tape driver (rmt0, rmt1, and rmt2) or the ADSM tape driver (mt0, mt1, and mt2).

The two other devices shown, pst0 and pst1, are pseudo-devices defined by NetTAPE, which do not point to real devices. We describe these NetTAPE pseudo-devices in Chapter 5, "NetTAPE Installation and Configuration" on page 105.

Figure 7 on page 27 shows how ADSM and other AIX applications can use different tape drivers to access different tape drive types, such as the 3490, 3590, or other SCSI tape drives.

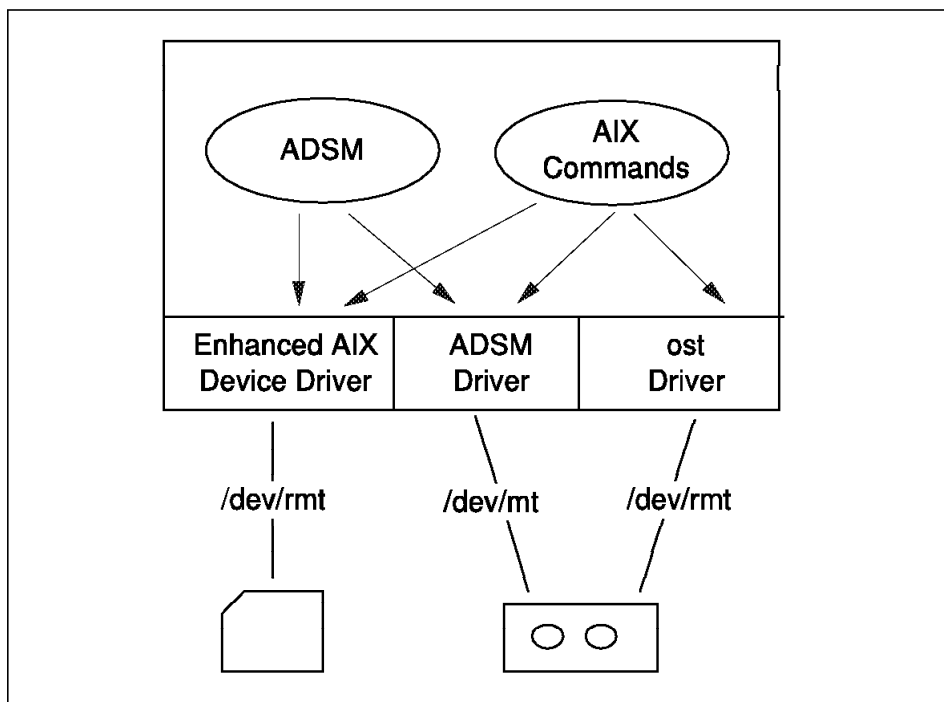


Figure 7. ADSM and AIX Command Use of Tape Drivers

2.2 Tape Technology

In this section we look at two generic aspects of AIX tape and tape library technology: recording techniques and device connections and protocols. For more details on particular tape devices, please refer to Appendix H, "IBM Tape Drive and Library Attachment" on page 245.

2.2.1 Recording Techniques

There are two main types of magnetic recording techniques, helical scan and longitudinal recording.

Helical scan is used essentially on 8 mm and 4 mm devices. Longitudinal recording is used by quarter-inch, half-inch (3480, 3490 and 3590 drives), quarter-inch cartridge (QIC), and digital linear tape (DLT) devices.

2.2.1.1 Helical Scan

Helical scan technology was developed for the video market. Because helical scan devices are produced in very large quantities for the consumer market, they have low production costs. Helical scan devices were not developed for computer applications, however, so they have some unpleasant characteristics, such as:

- Tape and head wear

On helical scan devices, the tape is wound around a cylindrical head that spins at high speed. The tape is in direct contact with the rotating read/write drum which causes the tape media and the read/write head to wear out. As a result, maintenance costs are high because the heads have to be replaced frequently and media costs are high because cartridges have to be replaced after a number of uses.

- Start/stop performance

Helical scan devices need time for the rotating head to be synchronized. Consequently, these devices perform at their rated speed when you read or write continuously (streaming), but their performance degrades drastically as soon as you have to do short bursts of reads or writes (start/stop), which happens often with computer applications.

Tape devices that use helical scan technology are typically the 8 mm and 4 mm drives.

2.2.1.2 Longitudinal Recording

Longitudinal recording devices were developed specifically for the computer industry. They generally cost more, on a unit basis, than helical scan devices, but provide better reliability, are easier to maintain, and give good performance in both streaming and start/stop mode.

The read/write head does not move and is not in direct contact with the magnetic tape, which effectively “flies” over the head. Because there is little wear on the head and the tapes, they have a much longer lifetime than on helical scan devices.

Data on longitudinal recording devices can be recorded on multiple paths. The most common technique is the 3490E technique, which writes 18 tracks in one direction and then 18 tracks in the other direction on 36-track tapes. The 3590 pushes this further by having 128 tracks and writing four times over the length of the tape; this technique is called *serpentine track interleaving*.

One advantage of using multiple paths is that the rewind time for a full tape is very short because the tape is positioned back to the load point.

Positioning to a specific point on the tape does not require the whole tape capacity to be traversed because the device is capable of going directly to the right track path, thus providing much better random access to tape data.

Performance of longitudinal recording devices is good in both streaming and start/stop modes because there is no head synchronization delay.

Tape devices that use longitudinal recording are the IBM 3480, 3490, and 3590, as well as the QIC and DLT devices.

2.2.2 Device Connections and Protocols

Most devices, in particular, tape devices, in the UNIX world are connected to the system through SCSI. However, 3480 and 3490 devices can also be connected through parallel or ESCON channels. Parallel and ESCON channel connections are typically used in installations that already have 3490 devices attached to a mainframe but want access from an AIX system. However, ESCON connections are useful in other installations as well because they provide distance and connectivity options that are not possible with SCSI devices.

2.2.2.1 SCSI Connections

There are different levels of SCSI adapters, devices, and protocols, with different characteristics for maximum bus length, maximum number of devices addressable on the bus, and bus bandwidth. Some host SCSI adapters are specifically designed for high performance applications by providing fast circuitry and performing parts of the SCSI protocol handling on the adapter, thus putting a lower load on the CPU.

Without going into too much detail, the main variations of the SCSI standard can be summarized as follows:

- SCSI-1** Referred to as SCSI, this is the first implementation of SCSI and provides a 1-byte parallel bus, with a maximum bus length of 6 m, and maximum data rates of 4 to 10 MB/sec, to which a maximum of seven devices can be attached.
- SCSI-2** SCSI-2 provides a number of additional (optional) features, such as:
- **SCSI wide**, which is a 2-byte-wide bus and allows for up to 15 devices on the bus.
 - **SCSI fast**, which permits transfer rates at 10 MHz.

Most SCSI-2 devices now operate on SCSI-2 fast and wide, combining 2-byte transfer and 10 MHz, thus permitting transfer rates of 20 MB/sec. A rate of 20 MB/sec can occur only if the device can deliver data at that rate, which is

not frequent. Attaching a 3 MB/sec device on a 20 MB/sec bus still gives you only a 3 MB/sec transfer rate from that device.

Another variation in SCSI is the single-ended versus differential SCSI bus.

Single-ended bus The single-ended bus interface consists of a single signal line for each of the SCSI data and control functions, and one ground return line. Typically, this setup allows distances of no more than 3 m or 6 m.

Differential SCSI bus The differential SCSI bus interface consists of two lines for each of the SCSI data and control functions. This interface provides much better immunity to electrical noise and therefore allows bus lengths of up to 25 m.

Most modern, high-performance tape devices use SCSI-2 fast and wide, differential SCSI bus interfaces.

2.2.2.2 ESCON Connections

You can connect 3490 tape devices through ESCON, a fiber-optic channel attachment developed for the mainframe environment. The main characteristics of ESCON are:

- Transfer rates of up to 17 MB/sec
- Maximum distance between device and host is from 3 km to 43 km, depending on the equipment used
- Using ESCON directors, which are cross-bar ESCON switches, you can establish simultaneous connections between several ESCON hosts and several ESCON devices.

Using an ESCON adapter on a RISC System/6000, you can access ESCON-capable tape control units, typically 3490s, through the appropriate device driver.

In AIX environments, ESCON devices are used primarily by installations that already have ESCON devices attached to a mainframe but want to access the devices from AIX hosts. However, ESCON provides possibilities that are beyond what is possible with SCSI, in particular much longer distances and much better connectivity. This can be important for installations that, for security reasons, have to locate their tape and library devices in a separate building from the computer room. You can easily do this using ESCON, whereas SCSI devices would require another UNIX host in the remote location, a (fast) network connection between the two buildings, and facilities for remote tape access. A direct ESCON connection does not require any host at the remote location and is thus easier to manage.

2.2.2.3 Parallel Channel Connections

Parallel channel connections are supported between a RISC System/6000 and tape devices (3480 and 3490) by the S/370 Parallel Channel adapter. The S/370 Parallel Channel adapter enables you to access installed 3480 and 3490 tape drives that were probably attached to a mainframe. This type of connection is slower than either SCSI or ESCON-attached 3490 devices, so you will probably use it only as a transition solution.

Note that the Parallel Channel adapter puts a much higher load on your RISC System/6000 CPU than either the SCSI or ESCON connections because it uses direct memory access (DMA), thus creating an interrupt for each byte transmitted.

2.3 Tape Library Technology

In this section we review the different kinds of tape libraries available for the UNIX environment. We do not go into configuration details; we look only at some of the main characteristics of libraries and how they affect library use.

2.3.1 What Is a Tape Library?

A tape library can be defined as a device that provides at least:

- One or more tape drives
- A number of slots where cartridges are stored when not in use
- One or more mechanical devices, *accessors* or *pickers*, that are capable of taking a tape cartridge from one of the cartridge slots, loading it into a tape drive, and vice versa.

Library function implementations differ in the way in which the accessor locates a cartridge in a storage slot for mounting and how much of the library inventory is known within the library or by the controlling software of the host.

Figure 8 on page 32 illustrates the components of a small SCSI library.

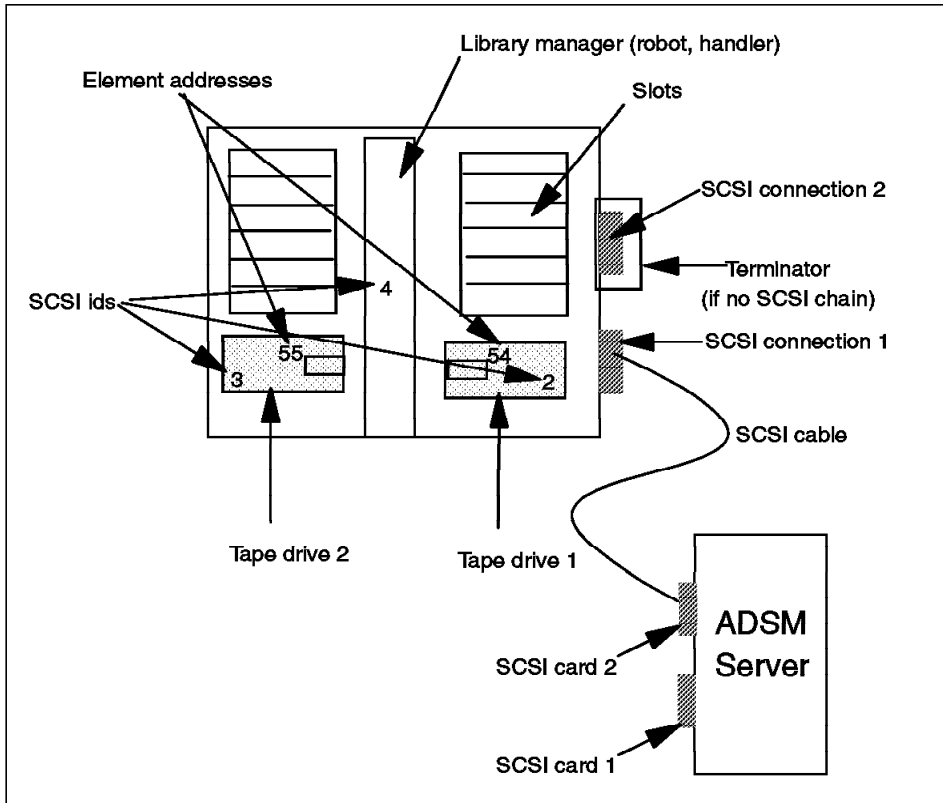


Figure 8. SCSI Library Components

2.3.2 Types of Tape Libraries

Starting from the simplest to the more advanced libraries, we find the following types:

- Small libraries

Small libraries typically have 5 to 20 cartridge slots that are little more than circular stackers. When a mount command is received, the library takes the cartridge next to the previous cartridge. It is then up to the controlling application to check that this is the correct cartridge and, if not, to ask for the next cartridge until the correct cartridge is reached. Examples of such libraries are the IBM 7332 and 7331 libraries when used in sequential mode.

- Libraries with random access to the cartridge slots

With these libraries, a mount command says, "Take cartridge in slot 8 and load it into drive 1." The controlling software in the host has to

know that the cartridge required was stored in slot 8. (An example of this type of library is the 7331-205 library without a barcode reader.)

- Libraries with external media identification capability, usually a barcode reader

With these libraries the mount command says “Take cartridge with external volume label VOL002 and load it into drive 1.” In this case, the host has to know which volumes are stored in the library but does not have to know their locations. An example of this type of library is the 7331-205 library with a barcode reader.

- Libraries with local intelligence and a database containing, among other information, the identification and location of each cartridge

The mount command is the same as for the previous case. However, the host can ask the library for information about a specific cartridge or all cartridges. Other information about the cartridges is stored and used, for example, volume categories (scratch and private) and host affinity (which cartridges belong to which host system). This is how a 3494 ATLDS works.

2.3.3 Library Device Drivers

Library commands are sent to a library through a special file. Physically, the commands can go through the bus on which the tape devices are attached, such as for the 7331 and the ESCON- or parallel-attached 3490 libraries, or it can go through a completely separate path, as with the 3494 with SCSI drives, where the library manager is addressed through the network or a serial link. In either case, you use a library device driver to control the accessor.

Specific library device drivers for the 3494, 3495, 3590 automatic cartridge facility (ACF), and 7331 have been developed and are part of the AIX enhanced device drivers described in 2.1.5, “Hardware Tape Drivers” on page 24. These library device drivers offer a command line interface, `mtlib`, which can be used to send orders to a library.

ADSM provides its own library device driver for all other types of (SCSI) libraries. The 7331 media changer is a special case; although the AIX enhanced device driver supports it, ADSM uses its own library device driver to control the 7331 library.

For a library we use a special file that differs from the files used for tape drives, as illustrated in Figure 9 on page 34.

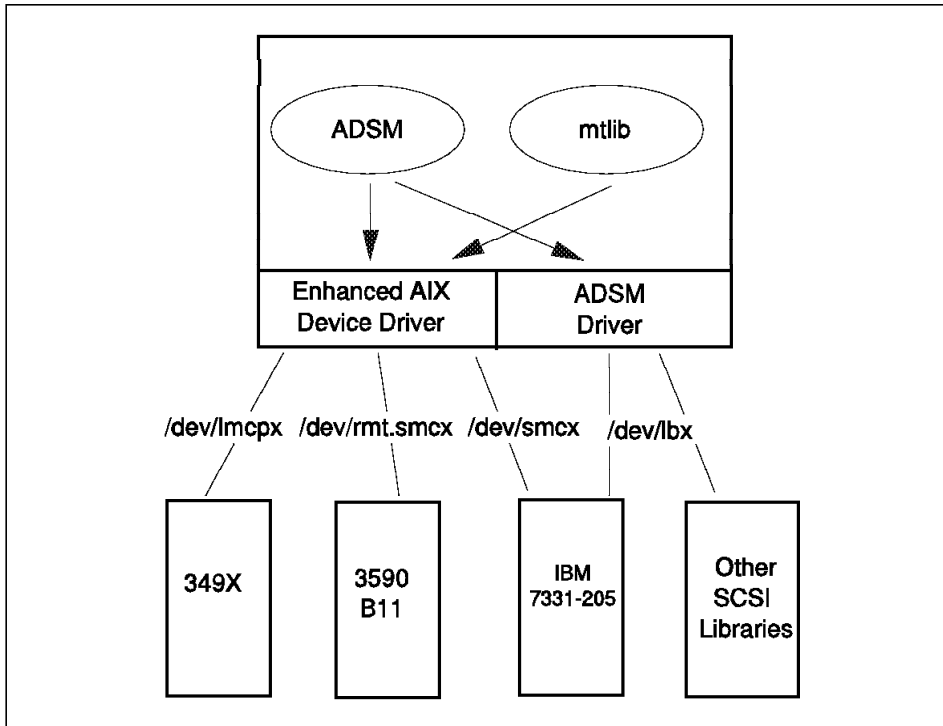


Figure 9. Library Device Drivers

Library device drivers use different names for each type of library:

- For the 3494 and 3495: `/dev/lmcp_x` (lmc stands for *library manager control point*)
- For the 3590-B11 ACF: `/dev/rmt.smc_x`
- For the 7331: `/dev/smc_x` (smc stands for *SCSI medium changer*), `/dev/lb_x` for the ADSM library
- For other SCSI libraries: `/dev/lb_x` (lb stands for *library*)

2.4 Tape Device and Library Sharing

In this section we look at how you can share a tape device between two or more hosts. Sharing a tape device in this context means that the device is physically accessible from the two systems and you can access it alternatively from either system. You cannot, and do not want to, access a drive from two systems simultaneously. We also look at tape library sharing and partitioning.

2.4.1 Tape Device Sharing

Most SCSI tape drives have only one SCSI port, that is, they can be connected to only one SCSI bus, even though you often see two plugs on the device (these are the bus-in and bus-out connections).

Multiple host access can be accomplished by having two hosts on the bus, as illustrated in Figure 10.

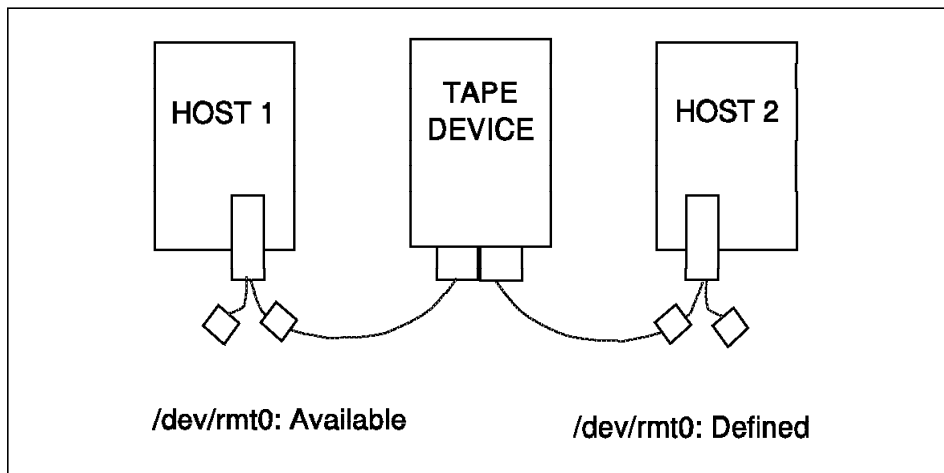


Figure 10. Two Hosts Accessing One Tape Device by Sharing the SCSI Bus

When using the configuration shown in Figure 10—a configuration that is not easy to manage—switching the device from one host to the other implies that you have to deconfigure the device on one side and make it available on the other side. The operating system does not provide support or protection.

If you make the drive available on both systems, an Open on the drive will typically do a Reserve on the device, so that the second host will get a device busy on its Open. There is no risk of double writes on the tape. However, there are (at least) two problems with this configuration:

- You cannot tell in advance whether the drive is busy (on the other host), and so you must be prepared to handle the device busy condition.
- If you have to Close and Open several tape volumes within one application, the other host can steal the drive from you between Close and Open, if you have done a release on the Close.

For these reasons, you might consider using such a configuration in an HACMP/6000 environment, where the HACMP software, running on both

systems, can manage device sharing in a coordinated manner, because each host is aware of what the other is doing.

Real SCSI tape drive sharing is practical only with devices that provide two SCSI ports, that is, the SCSI 3490 control units and the 3590. A tape device with two SCSI ports (also known as a twin-tail device) can be connected to two SCSI buses, with one host on each bus, as shown in Figure 11.

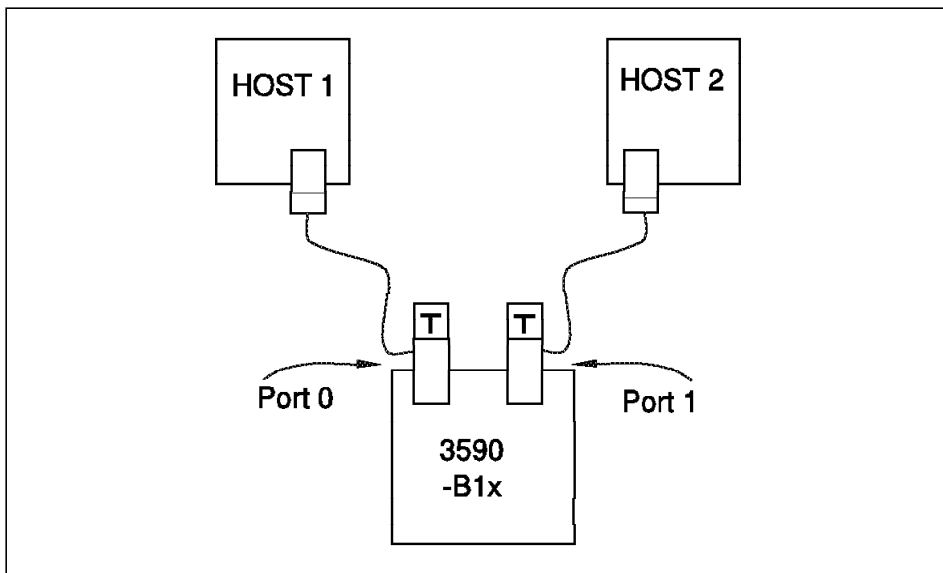


Figure 11. Two Hosts Accessing One Twin-Tailed Tape Drive

There is no SCSI bus conflict, and the drive can be made available on both systems at the same time.

Protection against concurrent use is accomplished by the Reserve/Release function in the drive: An Open from one host puts the drive in a Reserved state so that another Open will be rejected with Device Busy until the application in the first host closes the device special file, at which time the drive will be Released (by the tape device driver).

A typical example of using this type of sharing is a configuration with twin-tailed, daisy-chained drives, as shown in Figure 12 on page 37.

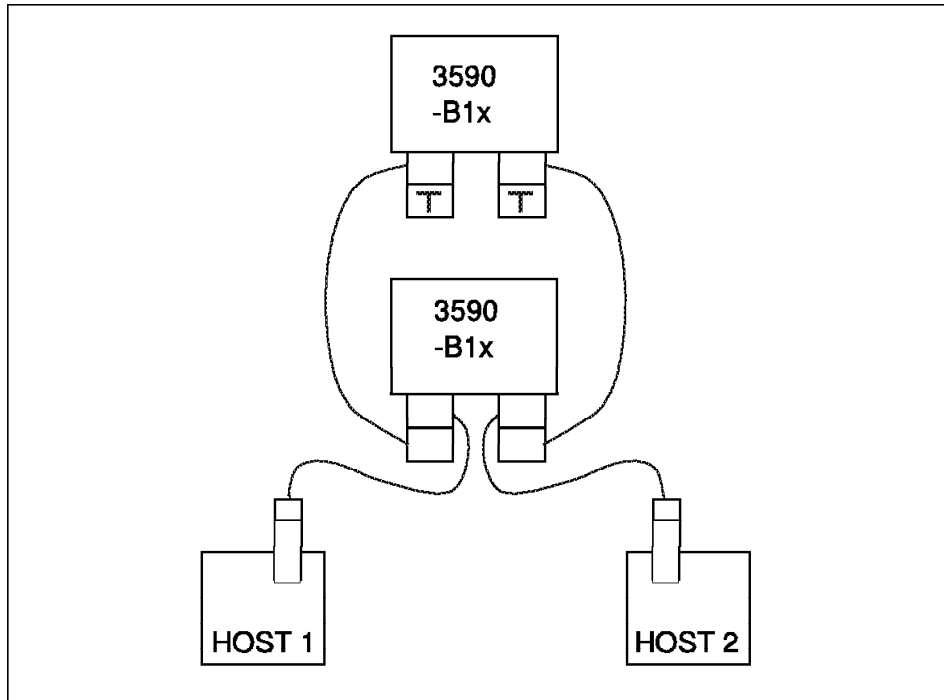


Figure 12. Sharing Two Twin-Tailed Drives between Two Hosts

The configuration in Figure 12 can be used with stand-alone tape drives or library drives. It enables you to dynamically access up to two drives from each host, if the drives are not used at the same time by the other host.

This configuration can be very useful if you have, for example, two ADSM servers on two systems: Each server can have one tape drive permanently attached, and, when two drives are required, for tape reclamation, for example, a server can access the second drive.

Another use for such a configuration is redundancy: in case one of the hosts is not available you can still access your tape drives from the other system.

2.4.2 Tape Library Sharing and Partitioning

Sharing an automated library, either between applications on the same system or different systems, is often a requirement, especially for 3494 and 3495 ATLDs, which can have a large number of drives and cartridges. Small SCSI libraries can more easily be dedicated to one application.

Sharing a library entails:

- Sending library orders, essentially mount and dismount commands, from more than one application or system
- Have taping drives connected to different systems

Sharing a library is not a problem for libraries that have a control path (for library orders) that is separate from the data paths (for reading and writing on tape). For example, a 3494 ATLDS can use a LAN attachment for the control path, which can be accessed from many systems.

The several drives found in a library can be distributed on more than one system in several ways:

- Static partitioning

Static partitioning is the simplest way of distributing drives, from an operational point of view. The same drives are always connected to the same system.
- Manual switching

Operational procedures are used to periodically move drives from one host to another.
- Dynamic switching

Device switching is handled by programs that dynamically change the device configurations on both hosts as required.

2.5 Tape Management Requirements and Trends

Tape on AIX systems is typically used for:

- Data backup, archiving, and migration, using ADSM, for example
- Data exchange, for transporting data to another system or site
- Product installation and update
- Batch jobs
- Database logging
- System backup and restore, which is performed by *mksysb* or *SYSBACK/6000*

As UNIX systems are used more and more for production environments and mission-critical applications, tape use in the UNIX environment is evolving, and there is a growing need for tape management tools of the same quality as those available for mainframes. Porting mainframe tape management systems to UNIX would seem to be the obvious solution, but this is not appropriate because of the large number of workstations involved. In a

mainframe environment, there is typically a small number of systems, each controlling a large number of tape drives. In a distributed computing environment, there are typically tens or hundreds of distributed workstations that have to access a few central tapes drives and libraries.

There are many ways of organizing a configuration of n workstations and m tape drives. The two extremes are:

- Use one tape drive (or more) for each workstation
or
- Have one workstation, with all of the necessary tape drives, providing tape services to all other workstations.

Neither of these solutions is really practical.

The main disadvantage of one tape drive per workstation is that there would be too many drives, most of them idle, without any central management. The completely centralized tape configuration is the easiest to manage but is generally not feasible because of drive attachment limitations, CPU saturation, or availability issues.

The only practical solution is somewhere between the two extremes. We typically find tape drives and libraries attached to some designated workstations (tape servers), with no tape drives on the other workstations or only very inexpensive tape drives for local use. Effective use of this configuration requires a number of functions:

- Remote access to drives as if they were local, so that classical UNIX commands are still usable (for example, `tar` and `mksysb`)
- Configuration tools that enable the (logical) sharing of the tape drives such that the drives can be assigned to an application or user when required, or to one specific host if they are directly connected to multiple systems.
- A network with enough bandwidth between the client nodes and the tape server nodes so that your tape operations are not slowed down because of the network
- Data transfer capabilities that can effectively utilize this underlying network
- A central catalog of the physical volumes, with ownership and access rights information, as well as media type and physical location, for each tape volume

A RISC System/6000 SP system presents the same kinds of problems as a set of distributed systems, as far as tape connectivity is concerned, because every node needs access to tape drives attached to one or two specific nodes.

Volume utilization is also undergoing radical changes driven by tape media technology. With the availability of fast drives with large capacities (10 GB per cartridge or more), the classical approach of assigning tape volumes (cartridges) to individual users or applications is not optimal. Not many applications have a need for that much capacity, and still less are capable of managing multiple tape files on one volume, so the media would, in most cases, be underutilized. Also, volume ownership by end users does not allow efficient management of the tape volumes, either for placing them in automated libraries (how do you know which volumes will be mounted frequently?) or when converting from one tape generation to the next (how do you combine 50 x 3480 user volumes onto one 3590, in a way that is transparent to the end user?).

For these reasons, there is now a trend to let the "system" manage tape volumes and the end users work only with files, so that the end users are not concerned about the physical media (disk, manual tape, automated library) on which their data files are stored.

Chapter 3. ADSM Tape Management

In this chapter we look at how ADSM manages tape drives, libraries, and volumes. We discuss how best to manage an ADSM server with respect to tape operations. We also discuss how drives and libraries can be shared between ADSM and other tape applications running on the same or different hosts.

The topics covered in this chapter are:

- ADSM structure
- ADSM tape device support
- ADSM tape media management
- ADSM tape operation
- ADSM device sharing
- ADSM tape performance

We describe the tape usage considerations that will enable you to maximize ADSM. We do not duplicate the information that is available in the formal ADSM manuals, so please refer to them for command syntax and general functional descriptions.

3.1 ADSM Structure

People are often confused about the relationships between the different ADSM elements that comprise the ADSM structure, in particular libraries and tape devices. Figure 13 on page 43 is an overview of the ADSM server structure that will help you visualize how the different ADSM elements fit together.

Some remarks on the conventions used in Figure 13 on page 43:

- Arrows indicate a link between elements; the link is defined in the element at the origin of the arrow. For example, a devclass is pointed to by a storage pool definition because, when you define a storage pool, you say:

```
define stg poolname devclassname ...
```

Some links are defined by a keyword; for example, you specify the storage pool to be used for client space management by:

```
def mgmt ... classname ... migdest=poolname
```

- When elements are represented as stacked boxes, there can be several of them at this place in the structure.
- An asterisk indicates that the link can be changed by an administrative command; for example, to change the library to which a device class definition points, use:

```
update devcl library=libraryname
```
- The dotted line pointing to the copy storage pool indicates that there is no link to it from the other storage pools. You specify the copy storage pool when you issue the BACKUP STG command.

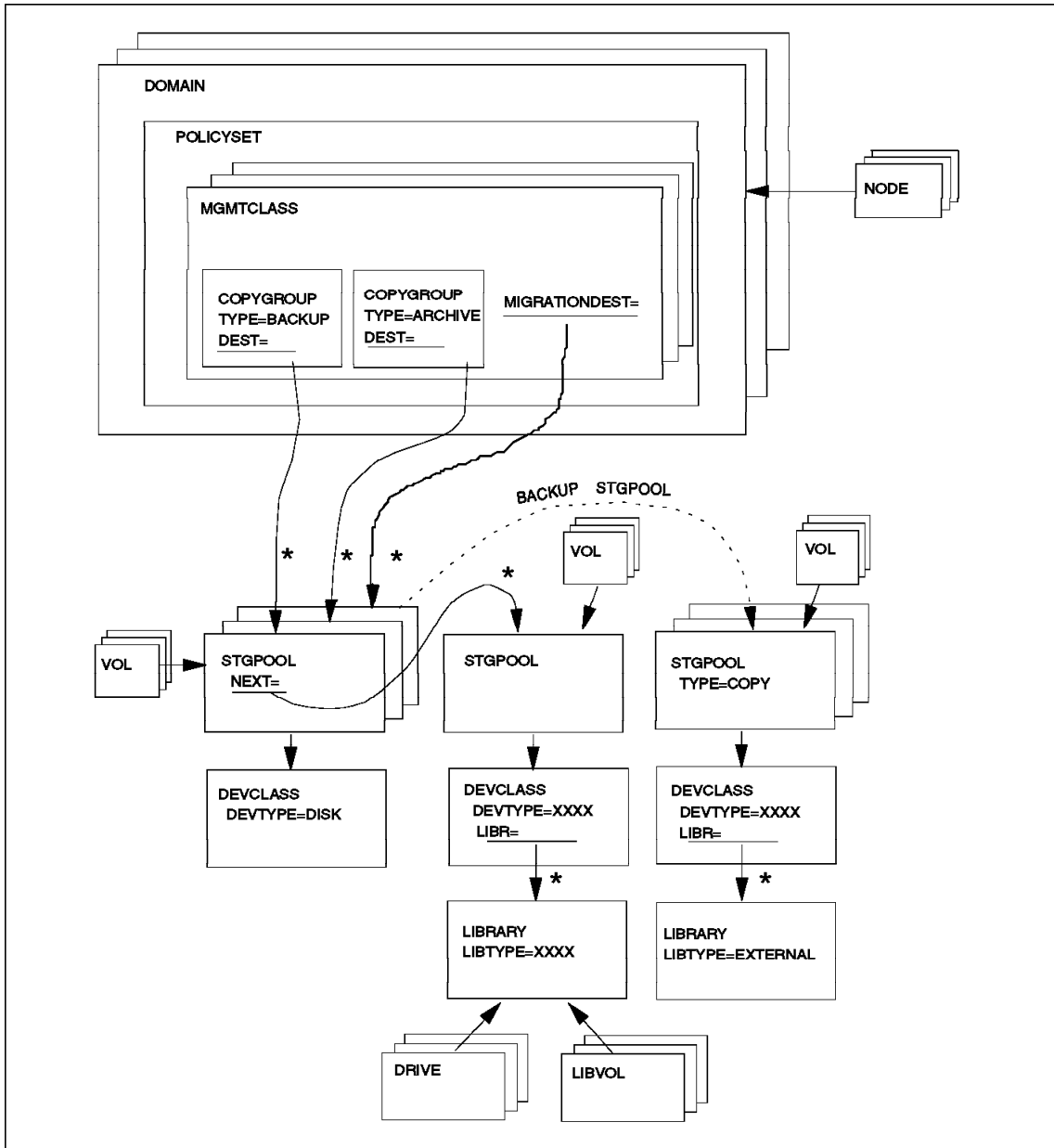


Figure 13. ADSM Server Structure

Let us walk through this figure, starting from the top:

- You can have several DOMAINS, each having client NODEs registered in them. A client is registered in only one domain, but you can have many client nodes per domain.

- There is one active POLICYSET per DOMAIN. There can be many inactive policysets; you just activate one of them. Only inactive policysets can be modified.
- Within the POLICYSET, you can have several management classes (MGMTCLASS), one of which is the default. The other, nondefault management classes will be used by the client nodes that specify them, for example, on include statements.
- A management class can have up to two COPYGROUPs, one for backup files and one for archive files. These copygroup definitions point to the first storage pool in the storage pool hierarchy.

Files coming from a space management client have their destination pool specified by the MIGRATIONDEST keyword in the management class definition instead of having a separate copygroup.

- The stack of STGPOOLS on the left indicate that there can be several of them; we show separate storage pools for backup, archive, and space management client files. All of the client data can be sent to the same storage pool, that is, you can have all of the copygroup DEST keywords and the MIGRATIONDEST keyword pointing to the same storage pool.
- Each storage pool is defined on one, and only one, device class. However, a DEVCLASS can be used (pointed to) by several storage pools.
- Storage pools have storage volumes. These are either tape volumes, or, in the case of disk storage pools, files that have been formatted for ADSM use but on AIX can be either JFS files or raw devices.
- Storage pools are chained in a linear hierarchy by the NEXT keyword. Storage pool migration moves data from one storage pool to the next.
- A DEVCLASS element defines the device type of the tape drive defined and points to a LIBRARY element by the LIBR keyword. The asterisk next to the arrow indicates that you can change the library used for a given device class, as long as you keep the same device type.
- The STGPOOL with TYPE=COPY is not linked to by other elements. The connection is done by the backup stgpool command. Otherwise, a copy storage pool is defined in the same way as a primary storage pool, by pointing to a device class that points to a library. The copy storage pool in Figure 13 on page 43 points to different DEVCLASS and LIBRARY elements than the primary storage pool; it could point to the same device class and library. We use our example to illustrate an External Library.
- A LIBRARY element can represent:

- Any number of manual tape drives
- or
- Any number of tape drives in the *same* automated library,
- or
- An External Library, which can be any number of drives, either manual, or in any number of automated libraries, or a combination of both.

A library that consists of manual tape drives or tape drives in the same automated library must contain tape drives of the same device type. An External Library can contain many different device types.

An External Library is managed externally to ADSM; no DRIVE elements are defined for it in ADSM.

For an automated tape library, the LIBRARY element defines the path used to issue library commands. Therefore all of the DRIVE elements pointing to a LIBRARY element must be in the same physical automated library.

Notice that the LIBRARY element is the highest element in the tree: all of the other elements point to it, directly or indirectly. A library does not point to any other element, so it will be the first element you define when adding new drives or device classes.

A library can be used by several storage pools, because several storage pools can point to one device class element, which points to one library element.

- DRIVE elements represent physical tape drives, either manual or automated.
- LIBVOL elements represent tape volumes in an automated library that have been *checked in*.

A tape volume defined in a storage pool, the VOL element in our picture, also appears as a LIBVOL if it has been checked in to the library. A LIBVOL element can exist without appearing as a VOL element if it has not been defined as a storage pool volume, for example, if it is to be used as a database backup volume.

Figure 13 on page 43 shows LIBVOLs pointing to the library element. This is only true for an automated library; there are no LIBVOLs for manual libraries.

3.2 ADSM Tape Device Support

A large variety of tape devices and tape libraries are supported in AIX. They cover most of the tape technologies available and are offered by a large number of suppliers. ADSM tries to keep up with all of the new devices and provides support for the most common drives and automated libraries—and the list is getting longer and longer as time goes on. You can choose the best device for your environment on the basis of your preferences, performance requirements, and budget. You can find a list of supported ADSM devices at the time of writing in Appendix A, “ADSM Supported Tape Drives and Libraries” on page 209; for the most current list, please see your ADSM server README file or contact your IBM representative. For ADSM STK device support through the ADSM for AIX StorageTek Tape Library Support Services Solution, see Appendix E, “ADSM for AIX StorageTek Tape Library Support Services Solution” on page 237. For possible ADSM STK device support through NetTAPE, see 7.1.3.2, “Additional Device Support” on page 155.

In this section we look at ADSM considerations for different types of devices and libraries and provide a checklist for defining your devices to ADSM.

3.2.1 Manual Tape Devices

Manual tape devices require human intervention for mounting tapes. Because an ADSM production system generally requires a lot of tape mounting, manual tape drives are practical only if you have operators who can handle ADSM tape mounting in addition to their other tasks.

3.2.2 Small SCSI Tape Libraries

We refer to small SCSI libraries for devices which provide automatic mount capability for about 10 to 20 cartridges, that is, a few hundred gigabytes. The library mechanism is addressed by SCSI commands which are sent to a specific SCSI ID in the device.

For ease of use with ADSM, the library should preferably contain a barcode reader (see 3.3.5, “Barcode Support” on page 56 for details) and at least two tape drives (to allow reclamation). Small libraries, such as the 7331, are fine for small ADSM installations.

3.2.3 Magstar 3590-B11 Libraries

The Magstar 3590-B11 is a single-drive unit with an automatic cartridge facility that allows random access to 10 volumes. It is, in fact, a small library (100 GB) and is viewed as an automated library by ADSM. However, it is a library with only one tape drive, and, even if you have two 3590-B11s, you cannot do tape reclamation from one drive to the other because two

(ADSM) libraries cannot be in the same storage pool. See 3.4.3, “Tape Reclamation” on page 64 for information about how you can perform tape reclamation in single-drive environments.

3.2.4 3494 and 3495 Libraries

The IBM 3494 and 3495 ATLDSs offer cartridge storage capacities from 300 to 1700 slots. They can contain either 3490E or 3590 cartridges and drives (or both) and support SCSI attachments (in the 3494, for 3490E and 3590) as well as ESCON connections (to a 3490E, in either a 3494 or a 3495) in AIX environments. The IBM 3494 ATLDS can also contain 3480 CST cartridges.

A *library manager* processor controls the robotics in ATLDSs and contains an inventory of all of the cartridges in the library. Library orders for mounting volumes are not sent to the tape drive, as they are with small SCSI libraries, but rather to the library manager. The connection between the library manager and the AIX system can be either serial, ESCON, or TCP/IP with Ethernet or Token-Ring.

You must have a special device driver to talk to the library manager; this special device driver is part of the enhanced AIX device drivers, *Atape* and *mtdd*.

See 2.3.3, “Library Device Drivers” on page 33 for more information about the library device drivers.

3.2.5 ADSM Tape Device Driver

ADSM development had to write a special driver, *adsm_mt_dd*, because the standard AIX *ost* tape driver does not provide sufficient functionality, in particular for label processing and end-of-volume handling. The *adsm_mt_dd* device driver is table-driven so that it is flexible and can cope with the many peculiarities of different tape devices. It also has code to control the robot as well as a device driver for optical storage devices. For the IBM 3490 and 3590 devices, the *Atape* and *mtdd* device drivers provide the functionality needed by ADSM, so ADSM uses these drivers. Please refer to 2.1.6, “ADSM Tape Drivers” on page 25 for details on the device drivers.

When you install ADSM, you must ensure that you are using the correct driver for the tape and library devices that ADSM will use.

3.2.6 ADSM Device Definitions

People commonly have problems when trying to connect tape or library devices to ADSM because they are using the wrong device driver.

Use the following checklist when you initially install ADSM or add a new tape drive or library type to your ADSM server:

1. Check that your device type is supported by ADSM.

You can find this list on the World Wide Web (<http://www.storage.ibm.com/adsm>) or in the ADSM server README file.

2. Ensure that your ADSM server level supports your devices.
3. Print and read the README file shipped with the ADSM server code.
The README file gives you all of the information you need for defining your tape devices and libraries. It also contains worksheets for preparing your definitions.
4. If you are using 3490 or 3590 devices, ensure that you have the correct device drivers at the correct level. These drivers are not shipped with ADSM; they are shipped with the hardware.
5. Install the ADSM server code, if it has not been installed.
6. Check that you have an ADSM Devices entry in the SMIT menu for devices. For 3490 and 3590 drives, the entries appear under the Tape Drive option of the Devices menu. You should also see them listed under the List All Supported Tape Drives option.
7. Connect the drives and libraries to your machine, if they have not been connected, and run `cfgmgr` to have AIX configure the new devices. You do not have to run `cfgmgr` if you reboot after connecting the tape.
8. In the SMIT List All Defined Tape Drives menu, your tape drives should be displayed with a status of *Available*. Write down the entry under *name* that is displayed for each of your drives.
9. If you are installing a new drive, first try using it from AIX to make sure that the hardware and connections are OK. For example, write a small file, using `tar`. If it succeeds, your tape drive works and is correctly connected.

You are now ready to define the drives to the ADSM server. Drives are always defined in ADSM libraries, even if they are manual drives, and a library is pointed to by one or more storage pools. As shown in Figure 13 on page 43, the LIBRARY element is pointed to by all of the other elements, so you must define the library first. The sequence of steps (from an ADSM administrator session with system privileges) is:

10. Define the library, with the `DEFINE LIBRARY` command.

The syntax varies according to library type. Refer to the *ADSM for AIX Administrator's Reference*. For the rest of this section let us assume that you have called your library MYLIB.

11. Define a device class with the appropriate devtype and point it to your library. Also, specify the number of drives that ADSM can use simultaneously in this library:

```
define devcl 8mm devtype=8mm library=MYLIB mountlimit=2
```
12. Define each drive in this library and give it a name, such as DRIVE1, by using the define drive command:

```
define drive MYLIB DRIVE1 device=/dev/rmt1
```
13. You will probably want to use these drives for storage pools, so define or update one (or more) storage pools, with the device type corresponding to your drive type, and have it use the MYLIB library you just defined.
14. If this tape storage pool is to be used behind a disk storage pool, update the disk storage pool to point to it:

```
update stgpool backuppool next=tapepool
```
15. If you have more than one drive, and want to use, for example, two drives simultaneously for migration, update your disk storage pool definition as follows:

```
update stgpool backuppool migprocess=2
```

Your drives are now ready for use under ADSM. All you have to do now is label and define the tape volumes as described in the next section.

3.3 ADSM Tape Media Management

In this section we describe how ADSM defines and manages the tape volumes. The topics we cover are:

- Types of ADSM tape volumes
- Scratch and private volumes
- Preparing volumes for ADSM
- Managing volumes in automated libraries
- Bar code support
- Cleaning tapes
- Managing offsite volumes

3.3.1 Types of ADSM Tape Volumes

ADSM tape volumes can be used as:

- Primary storage pool volumes

- Copy storage pool volumes
- Database backup volumes
- Export and import volumes

You can query storage pool volumes with the `query volume` command, but the database backup and export volumes do not appear in the output; they are only in the volume history file. To query the database backup volumes, enter:

```
query volhist type=dbb
```

To query the export volumes, enter:

```
query volhist type=export
```

3.3.2 Scratch and Private Volumes

ADSM works with scratch or private tape volumes, or a combination of both. Storage pool, database backup, and export volumes can be either scratch or private volumes.

The main practical difference between private and scratch volumes is that, for example, for storage pool tape volumes, you define private volumes with the `define volume` command, whereas with scratch volumes you simply specify how many scratch volumes are allowed in the pool by using the `maxscratch=` parameter in the storage pool definition. You can also run in mixed mode, that is, you predefine volumes in the storage pool but also allow a number of scratch volumes. When there are no empty private volumes left, ADSM asks for a scratch.

A scratch volume becomes a private volume as soon as it is used by ADSM and then is returned to scratch status again when it becomes empty.

For database backup and export operations, you can either specify specific volume names or scratch.

When ADSM needs a new cartridge, it asks for either one of the defined volumes, if you are using private volumes, or for any scratch volume if you are using scratch volumes.

Whenever ADSM needs data residing on an existing tape volume, it asks to mount a specific volume. This operation is sometimes referred to as a *specific mount*.

Note: Scratch volumes must be labeled, like any other ADSM volume, in order for ADSM to use them. See 3.3.3, “Preparing Volumes for ADSM” on page 51 for details on labeling volumes.

Some reasons for using scratch volumes are:

- With scratch volumes you only specify how many are allowed in a tape storage pool. You do not have to redefine them as you must for private volumes.
- If you are using manual drives without operators present, for example, during night shifts, you can premount scratch volumes so that ADSM write operations can proceed without human intervention. Premounting is especially useful if the tape drives have automatic stackers.
- Database backups are a good candidate for scratch volumes because you typically perform a database backup once a day and cycle the backup tapes after a certain number of days. Using scratch volumes makes it easier to define an administrator schedule for these database backups.

Some reasons for using private volumes are:

- Your installation has a policy of using private volumes only.
- You want to control which set of cartridges is used for ADSM.

ADSM Scratch Tape Check

If you are using a scratch pool, you will need some way of monitoring how many tapes you have left. Here is a short sample script:

```
#!/bin/ksh

#####
### ADSM Scratch Tape Check ###
#####
password=pass

# Now set up the default ADSM command
adminid=admin
DSMADMC="dsmadm -id=$adminid -password=$password"

# Check scratch tapes
countscratch= ` $DSMADMC q libv ] grep Scratch ] wc -l

print "There are $countscratch Scratch tapes left in the library."
```

3.3.3 Preparing Volumes for ADSM

ADSM always uses standard labeled tapes so that it can verify that the correct tape has been mounted. Therefore, before ADSM can use a tape, you must write a label on it. ADSM provides a utility, `dsmlabel`, to write

labels on sequential media. You must label all of the tape volumes that ADSM will use including storage pool, database backup, and export volumes. `Dsmlabel` is used with all ADSM supported tape and optical drives, whether they are stand-alone units or installed in automated libraries.

There are two modes in which volumes can be labeled with `dsmlabel`, manual and automatic. You can also use your own compatible labeling utility.

3.3.3.1 DSMLABEL in Manual Mode

In manual mode you load volumes into the drive one at a time when prompted by messages issued by `dsmlabel`. To label volumes manually, you issue the following command:

```
dsmlabel -drive=/dev/mt0
```

The `-drive` parameter specifies which drive to use for the labeling operation. If you are using an automated tape library for tape volume labeling, you specify the `-library` parameter which enables the library robotics to retrieve and return each labeled volume to the entry/exit port of the library:

```
dsmlabel -library=/dev/lb0 -drive=/dev/mt0
```

To use multiple drives in a SCSI tape library, for example, you must specify the `element` parameter of the `dsmlabel` command for each drive you use, so that `dsmlabel` can distinguish which drive contains the volume to be labeled. To label tapes in an AIWA AL-D220 4 mm tape library using both drives, we issued the following `dsmlabel` command (32768 and 32769 are the elements):

```
dsmlabel -library=/dev/lb0 -drive=/dev/mt0,32768 -drive=/dev/mt1,32769
```

You are prompted to remove a volume from the drive when the labeling operation is complete. If the library is equipped with an entry/exit port, volumes are returned to this port. To keep volumes in the library, specify the optional `-keep` parameter on the `dsmlabel` command:

```
dsmlabel -library=/dev/lb0 -drive=/dev/mt0,32768 -drive=/dev/mt1,32769 -keep
```

3.3.3.2 DSMLABEL in Automatic Mode

You can write labels on all of the tape volumes residing in the library when using an automated tape library for the first time. To label all of the tapes contained in our AIWA AL-D220 device, we issued the following command:

```
dsmlabel -drive=/dev/mt0,32768 -drive=/dev/mt1,32769 -library=/dev/lb0 -keep -search
```


This command searches all of the tape storage slots in the library for volumes and labels each volume that it finds. By default, each volume is returned to its original storage location.

For SCSI libraries that do not have barcode readers, this search is performed by mounting every cartridge in sequence. You are then prompted to enter the label for each volume, so you should write down the volume serial numbers as you place them in the library and then make sure you enter them in the correct sequence. Otherwise you may end up with cartridges where the external label does not match the internal label.

Entering the volume labels into a file and then having `dsmlabel` read the file will preclude human error and make the operation more automatic. A sample shell script for automating the labeling of volumes can be found in Appendix G.1, “Automatic Labeling of a Library” on page 243.

If your SCSI library is equipped with a barcode reader, or if you have a 3494 or 3495, the library can check the external labels, and `dsmlabel` does not prompt you for each volume.

3.3.3.3 Do-It-Yourself Labeling

Your installation may have operational procedures in place for labeling all new tape volumes. You may not want to, and may not have to, use `dsmlabel`, if your labeling utility is compatible with ADSM.

`Dsmlabel` writes three label records, VOL1, HDR1, and HDR2, and these records are what ADSM expects to see on the cartridges you label yourself. The fields that must be present in the three records of your pre-labeled cartridges are documented in Appendix B, “Records at the Beginning of a Labeled Tape” on page 219.

3.3.4 Managing Volumes in Automated Libraries

For ADSM to use volumes that reside in an automated tape library, such as a SCSI 7131-205 or a 349x library, you must explicitly tell ADSM about the volumes. ADSM does *not* use a volume in a library unless the volume is first defined to ADSM and added to the library volume inventory, which is part of the ADSM database. ADSM keeps track of the names and status of each volume that resides in an automated tape library.

Each volume in a library is categorized as either private or scratch. In an automated library, a mount request for a scratch volume specifies a volume category but no specific volume serial number. The library manager selects any volume from that category and proceeds with the mount. The application that requested the scratch mount will, after using the volume, change its category to specific, so that another scratch mount request does

not overwrite it, whereas in an automated library, a mount request for a specific volume specifies the cartridge volume serial number.

When you define an automated library to ADSM, you decide which category numbers ADSM will use for private and scratch volumes. You can choose any number, but be careful not to specify a category number used by another application accessing the same automated library.

3.3.4.1 Checking In Volumes to an Automated Library

The checkin process provides information to the ADSM server about a volume that resides in an automated library and enables the volume to be used for ADSM operations.

There are two ways of using the CHECKIN LIBVOLUME command—manual and automatic:

- Manual

The format of the command for manual mode is:

```
checkin libvolume 'libraryname' status=PRIVate 'volumename'  
                status=SCRatch
```

In manual mode you check in only a single volume.

For SCSI devices, a mount operator will be requested to load the volume into a drive within the library.

For 349x libraries, you can use the manual mode of operation even for volumes that have already been inserted into the library by the convenience or bulk I/O station.

For 349x libraries, you must enter an extra parameter on the command indicating the device type, for example, devtype=CARTridge for 3490 devices and devtype=3590 for 3590 devices.

- Automatic

The format of the command for automatic mode is:

```
checkin libvolume 'libraryname' status=PRIVate search=yes  
                status=SCRatch
```

ADSM automatically searches the library for new volumes that have not already been added to the library volume inventory.

In 349x libraries, checkin searches for new volumes only in the following categories:

- Private
- Scratch
- Insert

The scratch and private categories are specified on the define library command. When a volume is selected from the scratch category, its category is changed to the private category. When the volume becomes scratch again, its category is changed back to scratch. If you are using a 349x library, you would use the mtlib command to place volumes into the categories ADSM is going to use.

With the support for barcodes in SCSI libraries (see 3.3.5, “Barcode Support” on page 56) there are improvements in the way checkin works. The following summary is from the ADSM device README file:

The meaning of invoking CHECKIN LIBVOLUME and specifying CHECKLABEL=NO is changed. If CHECKLABEL=NO is specified, CHECKIN LIBVOLUME will do the following for SCSI libraries (the parameter will be ignored for any other type of library):

- a) Before processing the CHECKIN LIBVOLUME command, barcode data will be requested for all slots in the library if it is not already available.
- b) If a cartridge has a barcode, the first 6 characters of the barcode will be used as the volume name for the cartridge and it will be checked in under that name and inserted in the server volume database.
- c) If the cartridge does not have a barcode, the server will still mount it in a drive and attempt to read the volume name as it does currently.

Sample Checkin Libvolume Command:

```
Checkin libvolume libname status=scratch search=yes checklabel=no
```

3.3.4.2 Checking Out Volumes from an Automated Library

Any automated library, and especially a small one, can eventually get full, that is, there are no empty slots or volumes left. When you get to this stage, or preferably a bit earlier, you must decide which volumes to take out to make room for new volumes.

We assume that your library has not become full because you forgot, for example, to run expire inventory or tape reclamation on a regular basis. If you have not performed one of these two operations recently, do so before deciding that you have to take volumes out.

Look for the most inactive volumes (perhaps export or database backup volumes) to take out. You must perform a checkout operation on the volumes you select to basically tell ADSM that these volumes are not in the automated library. You can also update the volume information to specify a location, such as a desk drawer, which might help the tape operator in locating the cartridge.

When a checked out volume is needed, ADSM sends a mount command asking for the volume to be inserted into the library. Manual mounts in an automated library can usually be done by putting the cartridge in a convenience I/O slot or directly into the drive, depending on the library type.

3.3.5 Barcode Support

Some automated libraries provide a barcode reader that scans the external label on the cartridges. The ADSM server, as of Version 2, Release 1, Modification Level 7 (V2R1m7), supports barcode readers for SCSI libraries.

There is no need for special barcode support in ADSM for 349X libraries because the library manager handles the reading of the barcode labels on the cartridges. ADSM gets the information directly from the library manager.

Here is some information about barcode support that was gathered from the README file:

ADSM's intent is to exploit barcode hardware to make server operations more efficient and require less human intervention when using ADSM with larger SCSI libraries.

Barcodes are external labels that can be applied to the exterior of cartridges in a library. The barcode is meant to be read by a barcode reader, which is an optional feature of most SCSI libraries. The number of characters that can be included in a barcode varies from library to library and can be 32 or more characters long. ADSM labels tape cartridges with a six-character volume name. Therefore, it uses only the first six characters of a barcode as the volume name for the cartridge.

ADSM server operation and installation are enhanced because:

- Cartridges in a SCSI library can be labeled by a single command with no requirement to enter volume names from the console.
- CHECKIN LIBVOLUME processing for a SCSI library can be completed in seconds even when the number of cartridges is large.
- AUDIT LIBRARY processing can also be performed in seconds.
- CHECKOUT LIBVOLUME processing is faster and notices conditions that might indicate that an audit is required.

A new parameter, `-barcode`, has been added to the `DSMLABEL` command. Here is how it works:

- `Dsmlabel` requests that barcodes be returned from the library along with other information, such as slot status.

- When labeling a cartridge, the first six characters of the barcode string are used instead of prompting the user for a volume name string.
- If a cartridge does not have a barcode, it is not to be labeled if `-barcode` was specified.
- If the library does not contain any barcoded cartridges, or if the library hardware does not support barcodes, no cartridges are labeled, and `dsmlabel` terminates.
- Any barcode that exceeds six characters produces a warning message, and the first six characters are used for the volume name.

3.3.6 Retiring and Replacing Volumes

Tape volumes, most notably when you use helical scan devices, wear out and should be replaced after a number of read and write passes. ADSM keeps information about the number of times a volume has been mounted (as shown in the query `volume f=d` command), so that you can spot heavily used volumes and decide to retire them after a certain number of mounts. You will have to decide what this certain number is on the basis of your tape supplier specifications and your own experiences.

Another criterion for replacing a volume is its age, or the amount of time it has not been used, that is, its shelf life. After several years of perhaps less than ideal storage conditions, some cartridges tend to produce read errors. You can use the date on which the volume was last read from or written to, also obtained by using the query `vol f=d` command, to decide when it must be copied to fresh media. Note that it is not very likely that you will have to replace ADSM storage pool tape volumes because of their shelf lives because reclamation processes will usually access all of your storage pool tape volumes after a while.

You can use the `move data` command to copy the data from the volume you want to retire to other volumes and then delete the retired volume from ADSM.

ADSM can detect temporary errors on volumes; it marks the volume as `READ ONLY` and displays a message. ADSM also keeps a count of read and write errors for each volume, and again, you can see these counts with the query `volume` command.

Here is a sample output of the query `volume` command. Volume `ABC018` has been mounted 27 times, was last read from and written to on 5/22/96, and has had one write error.

```
adsm> q vol abc018 f=d

          Volume Name: ABC018
          Storage Pool Name: TAPEPOOL
          Device Class Name: ES3490
          Estimated Capacity (MB): 360.0
              %Util: 2.0
          Volume Status: Filling
              Access: Read/Write
          Pct. Reclaimable Space: 0.0
          Scratch Volume?: No
          In Error State?: No
          Number of Writable Sides: 1
          Number of Times Mounted: 27          <-----
          Write Pass Number: 2
          Approx. Date Last Written: 05/22/1996 20:48:21 <-----
          Approx. Date Last Read: 05/22/1996 19:03:36 <-----
          Date Became Pending:
          Number of Write Errors: 1          <-----
          Number of Read Errors: 0          <-----
          Volume Location:
          Last Update by (administrator): ADMIN
          Last Update Date/Time: 05/21/1996 16:42:06
```

If you have a media error on one or more storage pool volumes, follow this procedure to recover as much data as possible from the damaged volume (ADSM automatically sets the volume's access to READ/ONLY so that no new data is written to it):

1. Issue `MOVE DATA` to move as much data as possible off the damaged volume to a new volume. After the move data completes, any other data left on that volume may be lost.
2. Issue `QUERY CONTENT` of the damaged volume and save the output to a file. You now have a list of the files left on the volume that could not be moved, so you know which backup versions may be lost.
3. Issue `AUDIT VOLUME` with `FIX=YES`. This updates the ADSM database so that pointers that were pointing to the damaged backup versions are removed.
4. Issue another `QUERY CONTENT` to determine whether any backup files are left on the volume.
5. If files are left, you may be able to move them off the volume by running a second `MOVE DATA`. You must then compare the two query content lists to figure out which files are actually lost. Lost file information can also partially be obtained from the output of the audit volume process. This step may not work with some types of tape errors you receive. Also, you will not be able to issue the second `MOVE DATA` command if the tape

is from a scratch pool because after the first MOVE DATA command, the tape is released and returned to the scratch pool.

Let us step through an example to demonstrate this recovery process. As shown in Figure 14, the damaged storage pool volume, Bad Tape, has good data in parts A and C that we want to try to save by moving them to a new volume. The data in part B is lost (see step 5 below for details on how you might be able to re-create the lost files).

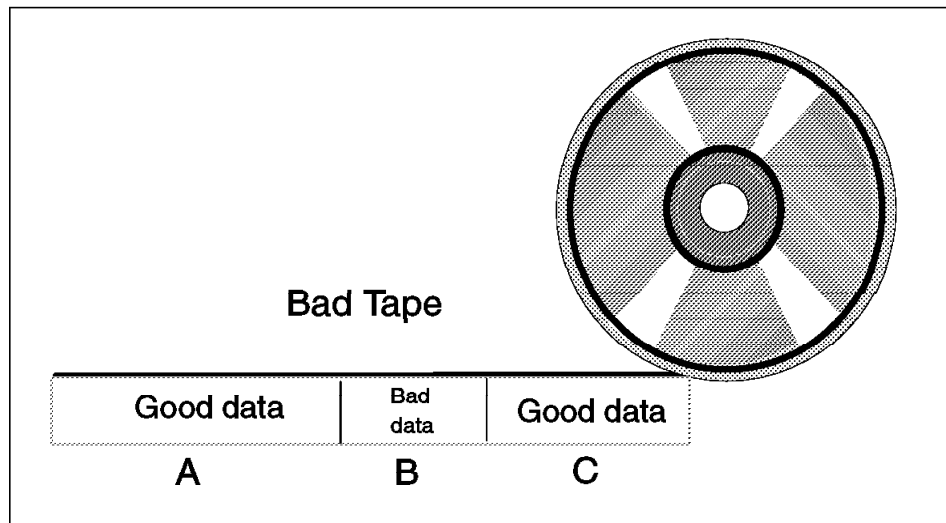


Figure 14. Damaged Storage Pool Volume Recovery

Here are the steps to try and save parts A and C. Your results may vary according to how bad the tape error is, that is, how bad part B is.

a. Issue MOVE DATA.

The data from part A is moved off the volume. You then get some kind of I/O error as the MOVE DATA tries to process part B. MOVE DATA will not skip to part C.

b. Issue QUERY CONTENT.

A list of all files that are on parts B and C is generated. This information is obtained from the AD SM database, which still believes that all part B and C data is available on the volume.

c. Issue AUDIT VOLUME.

All database references to all files in part B are deleted. All database references to all files in part C are preserved.

d. Issue QUERY CONTENT.

A list of all files that are on part C is generated. This is the only information left on the volume that the ADSM database believes is valid and accessible.

e. Issue MOVE DATA.

The data from part C is moved off the volume.

6. Re-create the lost files

If the original files still exist in the client file system, you can perform a new backup to re-create the lost backup files. The re-creation occurs the next time your regular incremental backup kicks off, or you can schedule a special incremental backup just for the file systems involved. Information about which file systems have lost backup data is obtained from the QUERY CONTENT lists. It is also possible to schedule a selective backup of the files involved. The contents of the original files in the client file system may have changed. Thus the contents may be different from that of the lost backup version, but this is the best re-creation that you can do.

The number of clients that are affected when you lose a single storage pool volume depends on whether you are using collocation. If you do not use collocation, which is the default, several clients will probably be affected. If you use collocation, probably only one client will be affected (but in this case the one client may potentially lose all of its backup data).

There may be extra copies of the files in the copy storage pools.

3.3.7 Cleaning Tapes

Cleaning tapes are special cartridges that clean the read/write head of the tape drive. You must clean a tape drive periodically; otherwise you will get an abnormal number of errors. The required cleaning frequency varies with the type of tape drives so please consult the appropriate manual or ask your supplier.

In some automated libraries, head cleaning is automated because the library itself decides when a drive needs cleaning and then mounts a cleaning cartridge.

Cleaning cartridges can create two types of problems for ADSM:

- If a cleaning cartridge, instead of a normal data cartridge, is mounted in the drive that ADSM tries to access, ADSM will get an I/O error.
- During automated operations, such as `dsmlabel` with search, automated checkin, or audit library, ADSM looks at all of the volumes in the library. Depending on your library model, the head cleaning cartridges are

distinguished from normal data cartridges either by their slot location, volume category, or external label (for example, they are called SHAMPO or CLNxxx).

However, in some library models there is no such distinction and ADSM cannot tell that a given cartridge is a cleaning cartridge. So ADSM will happily include the cleaning cartridge in its current operation, which results in undesirable situations when the cartridge gets mounted in a drive for writing.

Here is some additional information about 4 mm data and cleaning cartridges:

- DAT tape - The DAT designation on a 4 mm tape cartridge does not mean it is a data tape. DAT stands for *digital audio tape*, and it is not compatible with the 4 mm tape drive.
- DDS]]]] - The Digital Data Storage symbol that includes the four vertical bars signifies that the tape is compatible with the 4 mm tape drive in both read and write modes. The part number for this type of tape is P/N21F8758.
- DDS tape - The Digital Data Storage symbol without the four vertical bars signifies that the tape is compatible with the 4 mm tape drive in read mode only.

Use only those tape cartridges that are marked with the DDS symbol and four vertical bars. Tapes without the four vertical bars are read only and cannot be written by a 4 mm tape drive.

3.3.8 Managing Offsite Volumes

Placing copies of ADSM volumes offsite enables your installation to recover from partial or total destruction of your primary ADSM volumes, and, in the worst case, to reconstruct your ADSM server environment on another system in another location.

The types of ADSM V2.1 data volumes you must secure are:

- Database backup volumes
- Copy storage pools volumes

The method of taking backups of the database and storage pool volumes does not change because you put the volumes offsite, but some additional considerations apply. For details on managing offsite volumes, see *ADSM for AIX: Advanced Topics*, SG24-4601.

3.4 ADSM Tape Operation

In this section we describe how tapes are handled for some common ADSM client and server functions. The topics we cover are:

- Direct client backup, archive, and migration to tape
- Storage pool migration
- Tape reclamation
- Collocation
- Backing up ADSM
- Export and import
- Tape drive preemption

3.4.1 Direct Client Backup, Archive, and Migration to Tape

ADSM allows a client node to transfer data directly to tape and thereby bypass the usual disk storage pool. We do not generally recommend writing directly to tape because a tape drive is dedicated to one client for the entire duration of the client session. Writing directly to tape is not practical if you have many clients in simultaneous sessions and few tape drives.

However, here are some cases where you may want to write directly to tape:

- The amount of data to be backed up or archived is very large (whether the data is from one or more client nodes), so there is little point in storing the data on disk when it will migrate to tape a few hours later.
- The total elapsed time for backing up all of your clients enables you to stagger them during the day (or night), so that you can afford to have the clients write to your drives one at a time. This will save disk storage pool space.
- Your backup or archive window for a few selected clients is short, and you happen to have tape drives that are faster than your disks (not an uncommon situation with Magstar drives). If your network connection is fast enough, writing directly to tape will shorten the client session elapsed time.
- With space management clients, the file systems that ADSM manages are specific applications, not someone's home directory. If the files in the file systems are large and accessed primarily by batch jobs, recall times of minutes may be acceptable, and therefore it might make sense to migrate directly to tape.

There are two alternatives for implementing client data writes directly to tape:

- Define a special management class with a copygroup pointing to a tape storage pool. You direct the client data to use this management class by either making it the default class for the domain in which the client is registered or specifying the management class on the include statement (for backups) or as an archive parameter (-archmc).
- Use a management class pointing to a disk and tape storage pool hierarchy, but set the maximum size threshold so that large files will go directly to tape and bypass the disk storage pool.

3.4.2 Storage Pool Migration

Storage pool migration usually moves data from disk to tape, but it can also be used to move data from disk to disk or tape to tape. Migration is a server-initiated process based on the threshold settings of the disk storage pool, so it starts automatically when needed. However, you may have good reasons for wanting to control when migration occurs, rather than let it go at any time. For example:

- You do not have an automated library and do not want migration to occur when nobody is present to mount tapes.
- You are sharing drives between ADSM and other applications and you want the drives to be reserved for these other applications at certain times.
- You do not want migration to occur while the ADSM clients are doing their backups because it may overload the ADSM server, or, if you have a relatively slow tape device and a fast network, your disk storage pool may fill up faster than migration can empty it.
- You want to make sure that you have backed up your disk storage pool before migration takes place to avoid tape mounts for two drives. Two drives would be required for backing up a tape storage pool.

If you want to control migration timings, you can set the disk storage pool threshold to 100% and lower it by command (which can be scheduled) when you want migration to take place.

Note: Setting the high threshold to 100% prevents migration, but direct writes to the tape storage pool can still occur because you made the client data go directly to tape or the disk storage pool is full.

Because ADSM can migrate a storage pool to more than one tape device simultaneously, migration throughput is improved. The number of migration

processes is controlled by the *migprocess* parameter for the disk storage pool, which by default is set to 1.

See 3.6, “ADSM Tape Performance” on page 78 for parameters that affect the performance of migration.

3.4.3 Tape Reclamation

Reclamation reduces tape cost and the number of low-filled sequential volumes (optical disks and tapes). The tape fill-level is often not visible to administrators, especially as more and more ADSM expired files reside on volumes without being erased. ADSM logically erases the expired files by removing its database entries, but the files themselves still reside on the physical storage pool volume media.

The ADSM server checks sequential media volumes periodically and recognizes whether the amount of space on the volume has already reached a predefined reclamation percentage. If this percentage is reached, the ADSM server moves all valid files to an empty volume and redefines the reclaimed tape volume as empty.

Tape reclamation, like storage pool migration, is a server-initiated process based on thresholds. It requires two tape drives within the same library. If you want to control when tape reclamation occurs, you can change the reclamation threshold.

The reclamation threshold defines the percentage of reclaimable space on a volume, that is, the space that has been freed up through processes such as file expiration. The %Util is displayed on the query volume command. If query volume shows a volume as 45% utilized, that volume has 55% reclaimable space. If your reclamation threshold is 60%, the volume is not a candidate for reclamation.

Do not specify a reclaim value less than 50%. The primary goal of reclamation is to have more volumes empty after reclamation than before. If you set your reclaim threshold to 40%, for example, you might start with two volumes each 60% full and end up with two new volumes, one 100% full and the other, 20% full. You would not have gained additional empty volumes. Keeping the 60% default value is a good choice.

Only volumes marked as *full* are subject to reclamation because it would not make sense to perform reclamation on a filling volume. A full volume can no longer be written to because it has already been written to the end of volume. It is not necessarily full of valid data; its utilization may be low because of expired files.

The ADSM server cannot perform reclamation of a sequential storage pool on removable media (such as tape or optical) when only a single drive is available. In a single drive environment, however, you can perform reclamation manually (as shown in Figure 15 on page 65) by copying the valid data from the sequential storage pool into a temporary random access storage pool (a disk storage pool), thus consolidating the data, and then copying the data back to new volumes in the sequential storage pool. The old volumes can then be reclaimed for reuse. The occupancy of the new volumes will be lower because the inactive data (for example, deleted and expired files) occupying space between active data on the old volumes will no longer be there. For this process to be effective, it should not be initiated until the percentage of inactive data (or reclaimable space) on the old volumes has reached a realistic value, such as 60%, which is the ADSM default value for sequential storage pools.

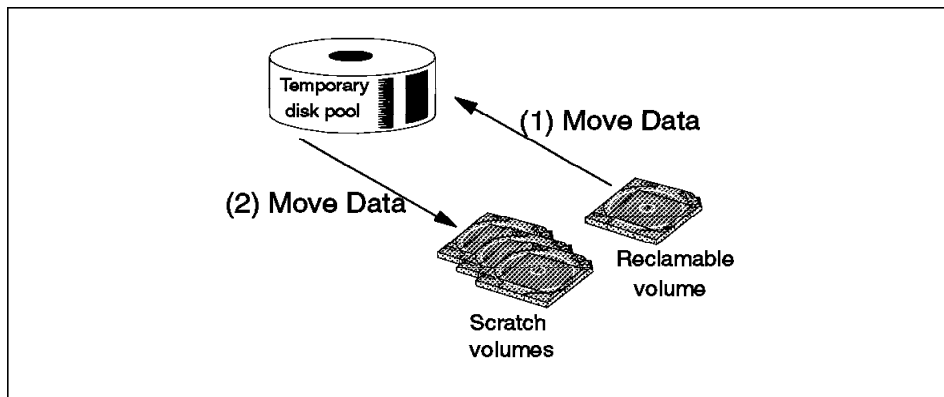


Figure 15. Single Drive Reclamation Process

To automate the single drive reclamation process, Van Stokes, Jr., a Network Development Engineer at Specialized Medical Management, Inc. in Fort Worth, Texas, wrote a script for his ADSM/6000 server. Mike Kaczmarek, IBM ADSM server development, updated the script and provided it as an AS-IS utility on the ADSM ftp server. Nick Higham, formerly from the IBM ITSO in Austin (currently an AIX and storage specialist in the UK), converted the script to REXX for use with ADSM for OS/2 servers. The OS/2 REXX version of the script is on the ITSO ftp server, ftp.almaden.ibm.com, in the SG244740 directory.

Here is a summary of the script. The script itself is in Appendix C, "Single Drive Reclamation Process" on page 221.

1. Establish a temporary random access storage pool

Confirm that a temporary random access storage pool exists and that it has a next pool defined in case of overflow. This script works best if the temporary storage pool is large enough to hold the valid data from at least one volume that will be copied from the sequential storage pool that is being reclaimed. If your system is random access storage constrained, the temporary pool, when not being used for reclamation, can be used as an ADSM disk pool or by other applications. Ensure that migration is turned on for the disk pool. When the disk pool becomes full and reaches the migration threshold, migration occurs, and the move data request (step 6) is suspended or canceled.

2. Obtain information about the sequential storage pool to be reclaimed

Query the server for vital parameters relating to the sequential storage pool to be reclaimed. The vital parameters are the percentage of free space at which reclamation should begin and the number of scratch volumes available on which to move the valid data.

3. Clear out the temporary random access storage pool

Ensure that the temporary disk storage pool is empty by initiating a move data request to the sequential pool. The disk pool is cleared out after this script runs, but there is nothing to prevent inadvertent use of the disk pool.

4. Determine which volumes are eligible for reclamation

Examine each volume in the sequential storage pool and check to see whether the percentage of free space exceeds the specified threshold.

5. Simulate reclamation (using move data) of each candidate volume

For each volume whose status is full and whose free space exceeds the specified threshold, initiate a move data request to the temporary storage pool. The move data copies all of the valid data from the volumes into the temporary storage pool.

6. Move the valid data back to the sequential storage pool

Initiate a move data request from the temporary storage pool back to the sequential storage pool. Before the move data request is made, the script checks that enough scratch volumes are available in the sequential pool to support the move data.

The sample script is an example of how to implement an automated sequential storage pool reclamation process when only a single drive is available. It is important to note that several variables must be set in order for the script to operate correctly. Please read the comments in the script for a complete explanation.

See 3.4.5.3, “Reclamation of Offsite Copy Volumes” on page 72 for additional considerations on tape reclamation in the case of offsite volumes.

3.4.4 Collocation

Collocation reduces restore performance by minimizing the number of tape volume mounts when restoring client data.

This is what can occur without collocation: Assume you have done incremental backups daily and the data migrates to tape practically every day. After a few months, the files of a particular node will be spread across many tape volumes. The day a user has to restore a whole hard disk, presumably after a disk crash, ADSM must mount all of the tape volumes that have the client’s data, which can take a lot of time. Your user is probably already a bit nervous after his disk crash!

Using collocation will make things much better because it tries to store each client’s files on different tape volumes from other client’s files (assuming that you have enough tape volumes). When you have to restore a whole client’s disk, ADSM will use much fewer tapes to mount than it would without collocation.

Collocation has a price: Migration from disk to tape entails more mounts than if collocation is turned off because the migration process asks for a new tape volume each time it migrates data for another client. But you do not have users on your back during the migration process!

If you have enough tape volumes, collocation will dedicate at least one volume per client node. When the size of tape cartridges was only a few hundred megabytes, this was not a problem because one client would quickly fill up one or more volumes. But the tape devices available now store 10 GB or more on one cartridge. If you have client nodes, such as small personal computers, they usually generate only a small fraction of that amount (especially if their include/exclude list has been defined judiciously). Therefore, they will never fill up a cartridge. If you have many small users, collocation could end up underutilizing many tape volumes. However, if you restrict the number of tape volumes, ADSM will store more than one client’s data on each volume.

If you have big client nodes that easily fill up more than one cartridge and you want to provide reduced restore times, collocation is ideal for such clients.

If you want collocation for your big clients but not for your small clients, you could define two tape storage pools, one with collocation and one without.

By assigning nodes to different management classes, in either the same or different domains, you can accommodate both types of clients.

Here are sample definitions for this situation (for simplicity, assume we have already defined a library, MYLIB, and a devclass, BIGTAPES):

The definitions for the small users would be:

```
def stg backuppool_tape_small collocate=NO
def stg backuppool_disk_small next=backuppool_tape_small
def mgmtclass domain_small standard smallfry
def copygroup domain_small standard smallfry standard -
  dest=backuppool_disk_small
```

The definitions for the big users would be:

```
def stg backuppool_tape_big collocate=YES
def stg backuppool_disk_big next=backuppool_tape_big
def mgmt Domain_big standard bigguys
def copygroup domain_big standard bigguys standard -
  dest=backup_disk_big
```

Do not forget to activate the policysset:

```
activate policysset standard standard
```

Register your client nodes in the appropriate domains:

```
register node smalluser1 smallsecret domain=domain_small
register node biguser1 bigsecret domain=domain_big
```

Now just add volumes to the pools and you are ready to go.

Note

Collocation is usually not very useful for archive data because users do not usually retrieve a whole set of files, although this may occur.

If you have tape volumes with data in a storage pool without collocation and you decide to turn on collocation one day, you will, after a while, end up with a collocated storage pool because each client's data will be written out on different volumes with reclamation processes. Inversely, if you start with collocation turned on and then you turn it off, each client's data will end up combined on the tape volumes.

3.4.5 Backing Up ADSM

ADSM is here to safeguard your users' data. ADSM will look after it as best as it can, but nothing is immune from hardware failures, media destruction, fires, or earthquakes. If you want your user data to be secure, ADSM itself must be able to recover from any of these potential problems.

ADSM Version 2 provides these server backup functions in addition to database and log mirroring:

- The ability to have two or more copies of every client's files by using the copy storage pool function
- The ability to back up the ADSM database in an incremental, nondisruptive manner
- The ability to prepare all of the data that must be stored offsite and to restore a working server environment on a recovery site by using the Disaster Recovery Manager (DRM) feature. DRM utilizes the copy storage pool and database backup functions.

Details on implementing ADSM server recovery are described in *ADSM for AIX: Advanced Topics*, SG24-4601. Below we provide an overview of backing up ADSM.

Here is what can happen to you, ordered by increasing levels of pain, and how ADSM can help you survive:

- A tape volume is destroyed, lost, or unreadable.

All of the data on the volume may be inaccessible. If you are using large capacity cartridges, this can be a lot of data.

If you have backed up the storage pool, using a copy storage pool, there is a copy (on other cartridges) of all of the files that were on the destroyed cartridge.

Copy storage pool cartridges are not image copies of the primary storage pool volumes. ADSM may need more than one cartridge from the copy storage pool to recover the data that was on one cartridge.

- The disks where the ADSM database are stored have crashed. The disks were not RAID disks, and the database was not mirrored.

The ADSM server will be down until you restore the database volumes. If you have a backup of the database, you can restore either the full database or one or more of the database volumes and restart the server.

- The RISC System/6000 system on which you run the ADSM server is somehow destroyed, but all of your tapes are OK.

You must buy or borrow another system, reinstall ADSM, and connect your I/O devices (disks and tape). Once you have hardware up and running, you have to reformat the ADSM database, log, and disk storage pool volumes, restore the database from your latest backup, and restart the server.

It should be clear from the last two cases that taking regular backups of the ADSM database is critical. Without a reasonably up-to-date database backup, you can be in serious trouble.

- Disaster strikes: Your whole machine room is destroyed as well as some part of the world around it.

But you (or your substitute) managed to escape in one piece, are sufficiently knowledgeable, and have the documentation required for the recovery.

If you have stored your database backup and copy storage pool volumes in a safe place, you can rebuild your whole ADSM environment in another location where there are machines that are in working order and somehow accessible from the client nodes.

We have only touched on some of the thorny things you must prepare for to cope with a disaster. The main messages are: ADSM has functions that enable you to be prepared; it is still up to you to do most of the work; things can get rather complex; and all the help you can get from the DRM feature will be of value to you if you must be ready for disaster recovery.

If your company does not have a disaster recovery site, you may not have to move cartridges to a remote site. Backing up the storage pools and storing the backup volumes in the same location as the primary volumes will still be useful for the cases where a primary volume is destroyed or lost.

We now look at some tape management considerations that are relevant to the backup of ADSM:

- What options are available for storing tape data outside the machine room, building, or city?
- What operational steps must be put in place to manage the movement of tape volumes between the production site and the offsite location?
- How are copy storage pool volumes reclaimed?
- What are the impacts of these operations on the ADSM server configuration?

3.4.5.1 Offsite Storage of Tape Volumes

There are basically two ways of getting data on tape to a remote location:

- Write the tape volumes locally and send the volumes to the remote location.
- Have tape devices installed at the remote location and send the data there using a wire between your ADSM server and the tape devices.

Shipping tape volumes is a known technology, so it is a low-risk option, although it must be well organized. The tricky part is bringing back the volumes from the remote site, because they have become empty or are needed for recovery. You probably do not want to bring back all of the volumes (you never want to put all of your eggs in one basket), so procedures must be in place to establish a list of required volumes and find the volumes in the remote location. A list is needed for both scheduled cartridge returns (the empty cartridges) and unscheduled returns (for partial or complete recovery).

The ADSM DRM feature can generate a list of scheduled cartridge returns (including both copy storage pool and database backup cartridges), using the command:

```
query drmedia * wherestate=vaultretrieve
```

The list of unscheduled cartridge returns required for ADSM server recovery is contained in the disaster recovery plan generated by the `drm prepare` command. The list also contains the offsite location of the tape volumes.

Accessing remote tape devices is the more advanced solution. There are two basic approaches:

- Have a distant device that is nevertheless viewed as a local drive by AIX.

This approach is possible only by using ESCON, which uses a fiber optic cable to allow distances of up to 43 km. If this distance is feasible for reaching your safe site, then it is relatively easy to operate: There are no software impacts because AIX and ADSM use the drives as if they were installed in the same machine room.

The only tape device that can presently be ESCON-attached to an AIX system is the IBM 3490.

One aspect to consider, however, is that an operator must be at the remote location, or you would need an ATLDS, such as the 3494.

- Have tape drives attached to a remote AIX system that can be accessed by your ADSM system as if they were directly attached.

ADSM does not support this remote tape access even with NetTAPE, but it is technically possible, so it may become available in the future. Even if this approach becomes feasible, the network bandwidth needed might make it impractical or too expensive.

3.4.5.2 Moving Cartridges to and from a Remote Location

Assuming you choose to send cartridges to your remote location rather than have a remote connection, you must take the database backup and copy storage pool tapes that you create everyday and send them to the remote site along with a list of the volumes. You must also tell ADSM that the tapes are offsite by updating the volume status to `access=offsite`.

If you use the ADSM DRM feature, the movement of cartridges to and from remote locations is simplified. The DRM `move drmedia` and `query drmedia` commands handle both ADSM database and copy storage pool volumes. A single list of the volumes is provided. DRM can also expire old database backup series (the combination of full and incremental database backups), using the `set drmbackupexpiredays` command.

If you are using an ATLD, you must also check out the volumes (see 3.3.4.2, “Checking Out Volumes from an Automated Library” on page 55 for considerations on checkout).

3.4.5.3 Reclamation of Offsite Copy Volumes

ADSM is rather clever in handling reclamation of offsite copy storage pool volumes.

An offsite volume is eligible for reclamation if the amount of empty space on the volume exceeds the reclamation threshold for the copy storage pool, even if the volume has not been filled. Thus a partially filled volume can be moved offsite and still be reclaimed if it contains more empty space than is allowed by the reclamation threshold.

Reclamation processing is performed as follows for offsite volumes:

- The server determines which files are still active on the volume to be reclaimed.
- These files are obtained from primary storage pools (or from another copy pool, if necessary).
- The files are written to one or more new volumes in the copy storage pool.
- A message is issued indicating that the offsite volume has been reclaimed.
- Empty scratch volumes are not automatically deleted.

Onsite scratch volumes in a copy storage pool are handled in the same manner as scratch volumes in a primary storage pool. However, ADSM provides special handling for scratch volumes whose ACCESS value is OFFSITE. If these scratch volumes become empty, they are not immediately returned to the scratch pool, because that would cause the volumes to be deleted from the database, and it would be more difficult to determine which volumes should be returned to the onsite location.

If a scratch volume whose ACCESS value is OFFSITE becomes empty, the volume is not returned to the scratch pool until the ACCESS value for the volume is changed to READWRITE, READONLY, or UNAVAILABLE. Thus the administrator can query ADSM for empty, offsite scratch volumes and return them to the onsite location:

```
QUERY VOLUME * ACCESS=OFFSITE STATUS=EMPTY
```

The query produces a list of all volumes that are stored offsite and are empty:

Volume Name	Storage Pool Name	Device Class Name	Estimated Capacity (MB)	%Util	Volume Status
TAP1	COPYPOOL	IBM8MM	0.0	0.0	Empty
TAP3	COPYPOOL	IBM8MM	0.0	0.0	Empty

You can then bring the empty, offsite volumes onsite. When the volumes are onsite, update their ACCESS mode to READWRITE:

```
UPDATE VOLUME * ACCESS=READWRITE WHEREACCESS=OFFSITE WHERESTATUS=EMPTY
```

All volumes updated with the UPDATE VOLUME command are deleted from the ADSM database and returned to the system as scratch volumes:

```
ANR2208I Volume TAP1 deleted from storage pool COPYPOOL.
ANR2208I Volume TAP3 deleted from storage pool COPYPOOL.
```

3.4.5.4 Impact on Your ADSM Server Configuration

There is little impact on your ADSM operations when you take database backups. You will probably take one database backup per day, either full or incremental, and this is a relatively fast operation.

Storage pool backups can have a much bigger impact, especially if your client nodes are sending large amounts of data. Storage pool backups result in essentially doubling the amount of data that the ADSM server has to move: Your average daily client data is migrated from disk to tape (if you are not writing directly to tape from the clients), and the same amount of data is written to the copy storage pool tapes.

If your disk storage pool is not large enough to hold all of your daily traffic, backing up the storage pools will be a tape to tape operation which requires two drives in one library. Your total ADSM drive occupancy will increase with the backup of ADSM itself, so you may require more tape drives than if you did not back up your storage pools. However, if you can schedule your operations so that your storage pool backups occur while the primary storage pool data is still on disk storage, you can avoid tape-to-tape operations.

Storage pool backups also affect the amount of space required for the database. ADSM must keep track of where the backup copy or copies are stored in addition to the primary copy. Backups of the storage pool do not double the database size, but about 200 bytes are added for each backup copy entry for a file.

Moving more data and having a larger database requires more CPU and central memory on your CPU. This is especially the case if the RISC System/6000 is not dedicated to ADSM. The storage pool backups may result in the ADSM load becoming significant during prime shift, whereas you may have been able to limit ADSM work to the night shift otherwise.

3.4.6 Export and Import

The ADSM export function enables you to put server definitions and node-related information and data on sequential volumes. The exported data can then be imported on another ADSM server. Some reasons you may use export are:

- You move your entire server from one system or platform to another. Export puts all of the necessary definitions and data on cartridges so that you can rebuild your server on the new system.

Note

Export and import require that you have compatible tape devices on both platforms, unless you use a device type of *file*.

If you have been backing up all of your storage pools, you are better off using the copy storage pool tapes.

- Some of your clients must be moved to a different ADSM server, for one of two reasons: The client stations are physically moving and you want to connect them to an ADSM server that is closer to their new location, or you have to balance the load on more than one ADSM server.

There are no other special tape considerations for export or import except that you must have the same type of tape drives on both systems.

Exporting a whole server with data, or even a few large client nodes, may mean moving a lot of data, so it is something for which you must plan. Performing an export with the `preview=yes` option tells you how much data will be involved for a particular export operation. Performing a small test export for one file system as an example enables you to estimate the time needed for your configuration.

3.4.7 Tape Drive Preemption

Given a finite amount of a resource, there is always a possibility for contention, that is, more potential users than resources. In particular, contention can occur with ADSM tape drives. For example, you might have two tape drives on your server, with one migration process running and writing to tape, and one session performing a retrieve from tape. If another client session or another server process starts up and needs a tape drive, there is none left.

But not all tape needs are equal; some are more important than others. For example, tape reclamation is not usually something very urgent whereas restoring the hard disk of your manager's personal computer is generally considered important!

ADSM has defined different levels of priority for tape tasks. A high priority task *preempts* a lower priority task; that is, the lower priority task will be terminated, and the drive will be dismounted and given to the higher priority task.

The priority levels for tape preemption are from highest to lowest priority:

1. Client restore, retrieve, and recall (from tape)
2. Database backup, export, import, and dumpdb
The above tasks can preempt mount points from the tasks below.
3. Audit volume and restore from a copy storage pool
4. Storage pool backup
5. Move data
6. Storage pool migration
7. Client backup (directly to tape)
8. Client archive and migration (directly to tape)
9. Tape reclamation

For example, if a database backup process was started, it would cause the termination of a migration process so that the database backup could take place.

3.5 ADSM Device Sharing

In this section we describe the sharing of tape drives between ADSM and other applications. ADSM for AIX provides an exit for tape operations, the External Library Manager, which can be used to improve the sharing of drives and libraries.

We look at ADSM device sharing without the External Library Manager. We describe how to use the External Library Manager to share devices between applications on the same and different systems. We also discuss library sharing and partitioning from a hardware perspective.

3.5.1 Without the External Library Manager

Tape drives and libraries are defined to ADSM as real devices without the External Library Manager exit. Therefore, ADSM expects these devices to be available when it needs them. ADSM asks for a mount on `/dev/mt1`, for example; If `/dev/mt1` is in use, ADSM reports an error and does not try using another drive even if another drive happens to be free.

The only way to tell ADSM not to use a particular drive is to delete the drive (by using the `delete drive` command). However, you must redefine the drive when ADSM needs it, which is hard to predict. Therefore, you tend not to use ADSM drives for other applications. An ADSM server typically needs tape drives for only a few hours a day, dedicating the drives for ADSM is not a very efficient use of resources.

3.5.2 Sharing with Other Applications on the Same System

You do not define any drives in ADSM when you use the External Library Manager. When ADSM needs a tape volume, it sends a mount request for that volume (either private or scratch). The External Library Manager, after handling the mount, returns to ADSM a special file name that points to the tape device. ADSM opens the special file and proceeds to access the tape.

The fact that the ADSM server is disconnected from the physical tape and library configuration affords greater flexibility—the External Library Manager selects any free device (of the appropriate type).

When you define a library as external, it can represent either manual drives or drives in automated libraries. In either case, you will only need to define the ADSM volumes that you will use if you are using private volumes. You

will not need to checkin or checkout volumes for an automated library. ADSM is no longer keeping track of the volumes in the automated library.

Volume categories in a 349x, either for private or scratch volumes, will not be known to ADSM, and therefore, the External Manager must manage the categories, that is, it will have to know, for example, from what category to ask for a scratch volume.

NetTAPE provides an External Manager for ADSM, so you can have NetTAPE manage the tape operations for your ADSM server.

You could also write your own External Manager by following the description in appendix A of the *ADSM Administrator's Guide*.

3.5.3 Sharing with Other Systems

If you use an External Library and have twin-tailed tape devices, the External Library Manager might be able to handle the reserve and release commands, in conjunction with sharing host, so that you can switch tape devices between two hosts.

3.5.4 Library Sharing or Partitioning

For considerations on sharing drives and libraries from the hardware and system points of view, please see 2.4, "Tape Device and Library Sharing" on page 34. The ADSM points for consideration are:

- Volumes

You have to ensure that the volumes used by ADSM and other applications accessing the same automated library are strictly distinct sets. Specify the correct volume categories, for both private and scratch volumes, in your library definition.

- Library control

Library control may not be a problem for certain types of automated libraries, for example, LAN-attached library managers such as the 3494. There may be a conflict (for example, with small SCSI libraries) when either sharing the SCSI bus or having two SCSI accesses to one library is not feasible.

- Drives

Drive sharing within a library is no different from drive sharing for manual drives. Classically, ADSM keeps its tape drives open, which makes them inaccessible to other applications on either the same or different systems. The only way to transfer a tape drive between ADSM and another application involves deleting the drive from ADSM when the

other application needs it and redefining it to ADSM afterward. This is obviously not a very desirable way to work.

One solution that makes drive sharing a lot easier involves using the NetTAPE product and the ADSM External Library Manager. Please refer to Chapter 7, “ADSM Tape Management with NetTAPE” on page 153 for details.

3.6 ADSM Tape Performance

We do not address tape performance measurement or tuning in this book; however, a few general considerations may be of interest.

Tape performance is important for the following reasons:

- To minimize drive occupancy

If ADSM writes faster to tape, typically during migration, drive occupancy will be lower and you will be able to use the drives for other tasks, either within ADSM or for other applications.

- To reduce restore times

Performance is especially important during restore operations because your users are normally in a hurry to get their files back. The time to restore should be part of a service agreement, and you should have a tape and system configuration that enables you to meet the requirements of the agreement.

- To improve client performance

You may choose to back up or archive directly to tape, bypassing disk storage pools, for performance reasons. Use a tape device that at least matches the client and network performance. If the client happens to reside on the same host as the server or is going through the SP2 switch, the bottleneck will probably be the client disks rather than the network.

Generally speaking, the ADSM server is a data mover machine, and it must be adequately configured to sustain a high throughput. Therefore:

- Choose a tape device that can sustain the data rate needed.
- Configure your disks appropriately, in terms of:
 - Type of disk
 - Type of adapter (SCSI or serial storage architecture (SSA))
 - Number of adapters
 - Placement of database, recovery log, and storage pool volumes on a sufficient number of disks

- Using disk striping with AIX V4
- Choose the network type (fiber distributed data interface (FDDI) or asynchronous transfer mode (ATM), Ethernet or Token-Ring)
- Correctly set ADSM performance options for your environment

Tape performance depends on a number of factors, which we illustrate with the IBM 3590:

- Native device capability

For example, an IBM 3590 is rated at 9 MB/sec, noncompacted.

- Effective device throughput

Given a rated transfer rate, the effective throughput achieved varies with the block size used. For the 3590, you need blocks of at least 128 KB or 256 KB to achieve close to 9 MB/sec.

Note: ADSM currently uses a 32 KB block size.

- Application efficiency

The code that writes to the tape must include good algorithms, for example, good buffering algorithms, so that it does not slow down the data transfer.

- Data profile

Transferring one big file will be faster than transferring many small files for the same total amount of data because there is overhead associated with processing each new file.

- Throughput of the source

You will not be able to write to tape any faster than you can read from the device on which the data is coming. The source device is generally the bottleneck when writing to a 3590 tape drive because your disk or network configuration will often not be capable of delivering data as fast as the 3590 can write it.

- Other factors, such as system load, network congestion, and interference with other devices on the SCSI bus, may be limiting factors.

Note: For additional performance tuning information, see the Server README file and the *ADSM Performance and Tuning Guide, Version 2 Release 1*. The APPC and TCP/IP tuning recommendations in the Server README file and the *Tuning Guide* apply if the client is backing up directly to tape, and for restore and retrieve operations.

3.6.1 Data Movement Batches

Two server parameters enable you to control how ADSM batches data movements between ADSM storage media, that is, storage pools on disk or tape. The parameters apply to migration, reclamation, storage pool backup and restore, and move data, but not, for example, to database backup or export and import.

The size of the data batches dictates how often the ADSM database is updated; that is, you can choose whether to do a few database updates frequently or a lot (up to 1000 files) of database updates less frequently. When the database is updated, the process stops until the update for the transaction is completed. The effect on the total process throughput depends on your configuration, but generally speaking, large transactions provide better throughput. This is especially true if you have tape devices that are very inefficient in start/stop mode.

`MOVEBatchsize nnnn` specifies the number of files that are to be grouped together. The default value is 32 in ADSM server levels before ADSM V2.1.x.6 but was changed to 40 starting at ADSM server levels V2.1.x.6. (where *x* is 5 for AIX V4.1.4 and 0 for other AIX levels).

`MOVESizethresh nnn` specifies the number of megabytes to be moved as one batch. The old default value is 1 MB; it has been changed to 500 MB in the latest releases.

The number that will be reached first determines the size of the batch; for example, assume you specified a `MOVEBatchsize` of 100 files and a `MOVESizethresh` of 200 MB. If you are migrating 200 1 MB files, for a total of 200 MB, the batch size will be limited by your `MOVESizethresh` parameter, that is, 100 files, or 100 MB. If each file is 4 MB, for a total of 800 MB, the batch size will be 200 MB, limited by your `MOVEBatchsize` parameter.

One batch is one ADSM transaction. If the transaction size is increased by using these two parameters, it will influence the amount of recovery log space needed by the server. Therefore, you may have to increase the size of the recovery log.

3.6.2 Management Classes for Directories

Another parameter that is particularly important for restore performance (very often a crisis situation!) is the `DIRMC` client parameter.

Directory entries from client nodes are usually stored directly in the ADSM database, except when information is attached to the directory entries. Access list information (in the bindery) is attached to directory entries for NetWare file systems, and access control list information is attached to

directory entries for UNIX file systems such as the Andrew File System (AFS). The access control information that cannot be stored in the database is stored in storage pools; if the client does not specify anything special, the storage pool hierarchy used is the same as that used for the backup or archive files.

Using the same storage pool hierarchy will be a problem at restore time if the client restores a whole tree because ADSM has to restore the directory structure first before the files. If the directory entries are stored on tape volumes, ADSM must mount all of the volumes and scan them for the directory entries. This is a lengthy operation on all tape devices, especially on slow devices such as 4 mm and 8 mm tapes.

To avoid having to mount all of the volumes and scan them for the directory entries, you can use the client *DIRMC* option, which enables you to specify a special management class for directories. If you direct this management class to go to a disk storage pool that is not migrated to tape, you will see significant improvements in the restore times when restoring a whole disk or directory structure. Directory entries are small, so it is not going to cost you a large amount of disk space.

You have two ways of forcing directory entries to go to that special management class and stay on disk:

- Use the *DIRMC* option in the client *dsm.sys* file.
- Ensure that the management class you want to use for directories has the longest retention period of all of the management classes in the policy domain. ADSM always stores directory entries in the management class with the longest retention period to ensure that the directories will not disappear before the files.

Chapter 4. NetTAPE Overview

This chapter provides an overview of the NetTAPE product. The topics we cover are:

- Objectives
- NetTAPE functions, including what is supported, what is not supported, and when you should use NetTAPE
- Technical details
- Hardware and software platforms supported

4.1 Objectives

NetTAPE's main objective is to provide a single-system image of data stored on tapes accessed from multiple systems.

The main NetTAPE functions are:

- Central tape operations
- Remote tape access
- Logical pools of tape units
- Processing of labeled tapes on AIX
- Support of automated tape libraries

NetTAPE builds on a previous IBM offering, CLIO/S, which provides fast data transfer capabilities between MVS and AIX and remote tape access from AIX to MVS. CLIO/S also offers remote tape access between AIX systems, with limited functions compared to NetTAPE. See Chapter 8, "CLIO/S Tape Management" on page 177 for detailed information about CLIO/S functions.

Note: NetTAPE is a collective name for two IBM program products:

- IBM Network Tape Access and Control System for AIX (NetTAPE) 5765-637
- IBM NetTAPE Tape Library Connection, 5765-643, which contains the automated library support

4.2 Functions

In this section we describe the functions that NetTAPE offers. How to use these functions is described in Chapter 6, "NetTAPE Administrator Operations and End-User and Application Use" on page 127. For

clarification we list those functions that are not included in the present level of NetTAPE. We conclude with a summary of when you would use NetTAPE.

4.2.1 What NetTAPE Supports

The main NetTAPE functions are:

- Facilities for central operation of distributed tape operations

NetTAPE provides facilities for centrally managing tape operations. Tape operators, from one or several operator stations, get mount and unmount messages for a set of tape devices attached to multiple systems. You can split the NetTAPE-managed devices into operator *domains*.

Tape operations are typically done under the NetTAPE XWindows, Motif-based graphical user interface (GUI). A command line interface is also available but does not provide all of the functions that the GUI offers.

- Device pools

Users can ask for a drive by using a generic name, for example, MANUAL3490, which is actually a device pool defined by the NetTAPE administrator. MANUAL3490, in our example, contains all of the manually operated 3490 drives. A user does not have to select a specific tape drive. Providing device pools is more user friendly because it hides the physical configuration of the tape devices from the user. i2.in NetTAPE

- Tape device allocation

NetTAPE uses device pools to perform device allocation, that is, the selection of a drive appropriate for the user request.

NetTAPE treats all of the drives and libraries as global resource pools, thus allowing better allocation strategies than if device selection were left to the user.

NetTAPE tape drives are assigned *states*:

- Up

Up indicates that the device is available for allocation.

- In use

In use indicates that the device has been allocated to an application or user.

- Reserved

Reserved indicates that the device is temporarily unavailable for allocation by NetTAPE, for example, because it is used by an application that cannot go through NetTAPE.

- PenRs

PenRs indicates that the device is in pending reserve status, that is, the operator has put the device in reserved status but the device is currently in use. The device becomes reserved when the user unallocates the drive or when it reaches the end of volume.

- Down

Down indicates that the device is unavailable for allocation.

- Tape library support (provided by the NetTAPE Tape Library Connection product)

NetTAPE provides support for automated tape libraries, thus integrating tape library operations with manual drives, from the end-user perspective.

- Remote tape access

With NetTAPE a user is not restricted to the tape drives directly connected to the system to which he or she happens to be connected but can access any tape drive in the NetTAPE network.

- Tape label support

NetTAPE supports the tape label formats usually used on mainframe systems, for example, MVS, thus enabling the interchange of tape cartridges between MVS and AIX applications.

NetTAPE also supports multivolume tape files for AIX.

- Record formats

NetTAPE supports tape files with record formats typically used on mainframes but not normally supported on UNIX. The record supported formats enable you to process tape volumes created by mainframe applications in an AIX environment.

- End-user interfaces

NetTAPE has two end-user interfaces, a command line and an API. The basic functions they provide are:

- Allocate (a drive)

Allocate requests NetTAPE to select a tape drive and mount the specified volume on the drive. Specific characteristics can be specified, such as the type of tape labels and record formats.

Allocate also opens the tape device at the file number indicated (the default is file 1).

- Open (a file)

Open is used for accessing the next file on the tape volume, that is, for multifile processing.

- Close (a file)

Close tells NetTAPE to save the information about a tape file that was just created.

- Unallocate (a drive)

Unallocate unloads and unmounts a volume and releases the tape drive.

The NetTAPE API, callable from either C or Fortran, provides the same functions as the command line interface, plus normal tape functions, such as read, write, and skipping tape files.

- Support for the ADSM External Library Manager

NetTAPE provides an ADSM External Library Manager so that you can integrate ADSM tape operations with other applications and thus facilitate drive sharing. See Chapter 7, “ADSM Tape Management with NetTAPE” on page 153 for more information.

- Coexistence with CLIO/S

If you have CLIO/S installed in addition to NetTAPE, probably because you are accessing MVS tape data directly from AIX, NetTAPE will filter and pass the NetTAPE commands that must be processed by CLIO/S to CLIO/S.

If you want to use CLIO/S for accessing AIX-attached tape drives, the drives cannot be shared with NetTAPE (but you really have no reason for doing this).

- Automatic volume recognition

NetTAPE provides automatic volume recognition (AVR), a function available with MVS for many years. With AVR the system, in this case NetTAPE, detects when a volume is mounted on a tape drive by reading and checking the volume labels. When the volume is later requested by an application, NetTAPE allocates the drive directly to the requester. AVR speeds up tape access for the application and allows your operators to premount volumes that you know an application will use, for example, overnight.

- Tape volume information

Although NetTAPE does not contain a tape catalog function, it does provide a facility for recording basic volume information so that the end user does not have to enter the same parameters every time. The information is stored in *VFILES*, volume files, with one volume file per tape file or data set.

If you already have a tape catalog, on MVS, for example, you could transfer the relevant information from the catalog to NetTAPE VFILES to facilitate tape allocation for your users.

4.2.2 What NetTAPE Does Not Support

In this section we list functions or devices that NetTAPE, in the first release, does not support. You may think that some of these items are available if you read the product description too fast or are inclined to wishful thinking, but:

- NetTAPE does not support libraries other than the IBM 3494, 3495, 3590-B11, and STK libraries listed in 4.5.5, "Tape Library Support" on page 104. In particular, there is no support for any small SCSI library, either IBM, such as the 7331-205, or non-IBM. For a complete list of supported devices, see Appendix D, "NetTape Supported Tape Drives and Libraries" on page 235.
- The ADSM NetTAPE interface does not allow ADSM to access remote tape drives. ADSM also cannot access remote tape drives when not using NetTAPE.
- No software platforms other than AIX are supported; this is also true for the NetTAPE component that runs on the client systems.
- There is no tape volume catalog function in NetTAPE or any possible link to an external tape management system catalog. Therefore, for example, there is no volume ownership control in NetTAPE.
- There is no nickname facility for tape drives. Thus you cannot have NetTAPE use identifications other than the drive names defined in AIX, such as `/dev/rmt0`.

The problem with the AIX drive names is that they are subject to change when you move drives from one host to another or reconfigure a system. The physical drives stay the same, so many installations have given names to the drives, for example, EXB102 or MAGS1002, and operators know the drive locations by these names.

NetTAPE will not use your drive names.

4.3 When Would You Use NetTAPE?

Let us review some of the reasons why your installation would want to use NetTAPE. On the basis of the functions outlined in 4.2, “Functions” on page 83, consider using NetTAPE if you have one or more of the following requirements:

- Your tape devices must be accessible from several applications on the same (AIX) system; for example, you want to share your tape drives between ADSM and other applications.
- You have to access tape devices attached to other hosts than where your applications are running
- You have to read from or write to tape volumes created on a mainframe, for example, under MVS.
- You have to consolidate your tape operation tasks.
- You have to create multifile tape volumes under AIX.

If any of these requirements applies to your environment, NetTAPE will provide the functions necessary for you to implement your tape management solution and allow you to have a single-system image of your tape devices and servers.

4.4 Technical Details

This section provides an overview of the NetTAPE components and how they relate to each other. We use a simple scenario to illustrate the different steps involved in processing a user request and describe each NetTAPE function in detail. The topics we cover are:

- NetTAPE components
- End-user sample scenario
- Central operations
- Device pools
- Tape device allocation
- Tape libraries
- Mount queues
- Remote tape access
- Tape labels
- Record formats

- End-user interfaces
- Support for the ADSM External Library
- Volume information (VFILE)

4.4.1 NetTAPE Components

A NetTAPE installation generally consists of several AIX hosts, each with one or more NetTAPE components.

These components are the:

- NetTAPE status daemon

The status daemon, the NetTAPE control point, knows about the entire NetTAPE configuration. It generally has local tape drives, although this is not a requirement.

- NetTAPE tape server

The tape server is a system with directly attached tape drives.

- NetTAPE library server

The library server is a system that controls a library. It can also have the library tape drives attached to it, and, if it does, it is both a NetTAPE library server and a NetTAPE tape server.

An example of a system that is only a NetTAPE library server would be a system that controls a LAN-attached 3494 library, without any (SCSI) tape drives attached.

- NetTAPE operator

The tape operator interface runs on any host within the NetTAPE network.

- NetTAPE client

The client is a system from which users request tape mounts and process tape files.

- Network

All communication among NetTAPE components, as well as data transfers to and from remote tape devices, uses TCP/IP.

You will have in your NetTAPE network:

- One, and only one, NetTAPE status daemon node
- As many tape and library server nodes as you have systems with tape drives attached that you want NetTAPE to manage

- One or more operator nodes
- Probably many client nodes

In terms of licensing, you will need:

- One basic license for NetTAPE (product number 5765-637) for the NetTAPE status daemon server
- One basic license for NetTAPE (product number 5765-637) on each tape server
- One NetTAPE Tape Library Connection license (product number 5765-643) on each library server

Separate licenses are not required for the operator and client systems, even if you run them on different hosts.

Figure 16 shows a sample NetTAPE configuration.

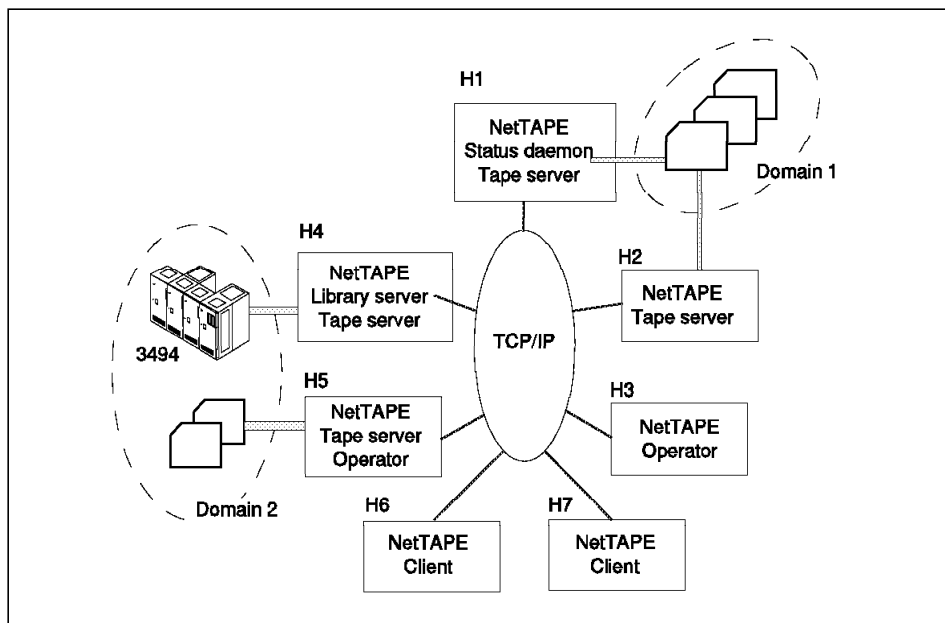


Figure 16. Sample NetTAPE Configuration

Referring to Figure 16, we see:

- Host H1 is the NetTAPE status daemon node. It also has manual tape drives attached to it, so it is also a tape server node.
- Host H2 is a NetTAPE tape server node. The tape drives attached to it are also attached to host H1.

- Host H3 is used as a NetTAPE operator node responsible for managing the drives in operator Domain 1, that is, all of the drives attached to hosts H1 and H2.
- Host H4 is a NetTAPE library server node because it manages a 3494 ATLD. It is also a tape server node because the library drives are directly attached to it.
- Host H5 is a NetTAPE tape server node because it has manual tape drives directly attached to it. It is also a NetTAPE operator node that controls the devices in Domain 2, that is, the 3494, the drives in the 3494 library, and the drives attached to host H5.
- Hosts H6 and H7 are NetTAPE clients nodes. They do not have any tape drives attached to them.

Notes:

1. All of the hosts communicate by using TCP/IP over a network, which can be Ethernet, Token-Ring, FDDI, ATM, or any other network which supports TCP/IP.
2. NetTAPE clients can be used on any of these nodes, not only on hosts H6 and H7.
3. Users on any NetTAPE client from anywhere in the network can access tape drives on any host, as long as they have a user ID on the NetTAPE server host that controls the tape drives.
4. All of the hosts are RISC System/6000 systems or RISC System/6000 SP nodes.
5. Although several nodes are illustrated, all of the NetTAPE components can run on one node.

4.4.2 End-User Sample Scenario

To illustrate how NetTAPE processes a user or application request, let us look at a simple scenario. The syntax of various user commands is explained with more examples in 6.2, “End-User and Application NetTAPE Use” on page 137.

In this scenario, the ADSMRES2 user on host H7 in Figure 16 on page 90 asks for a standard labeled tape volume, writes some data on it, and releases the volume.

In general terms, this is what happens:

1. The user enters the NetTAPE `cltalloc` command:


```
cltalloc -v VOL004 -V VOL004 -l SL -d mydata
```

where `-v` specifies the external volume label, `-V` specifies the internal volume label, and `-d` indicates the tape file (data set) name that the user will write on the tape volume. A device pool has not been specified, so the default is used. The user must know the volume serial number that he or she wants to access.

2. The NetTAPE client code on H7 checks the syntax of the command and, because it is correct, sends the request to the NetTAPE status daemon on H1.
3. The NetTAPE status daemon executes the following steps to process the request:
 - a. It checks which drives are free in the default device pool. Let us assume that the default pool consists of the tape drives in Domain 1 and the drives attached to host H5.
 - b. It selects one drive from the default pool, let us say a drive on H2.
 - c. The drive on H2 is a manually operated drive, so NetTAPE sends a mount request message to the operator GUI screen on H3, as shown in Figure 17.

Mount Message Area										
Msg #	Actor	Pool Name	Server	Device	Type	Label	Vol/SN	Client	User	Message
5	Mount	ANTBMM	being	mt0	MM8	SL	VOL00	being	adsmres2	mount vsn:VOL004 dev:/dev/mt0 node:being

Figure 17. NetTAPE GUI Mount Message

NetTAPE also displays a mount message on the tape drive display if the tape drive has a display.

4. The operator gets the VOL004 cartridge from a rack, mounts it in the drive indicated on the mount message (as `/dev/mt0`), and clicks on the **Mount** button (not shown).
5. The NetTAPE status daemon on H1 creates a pseudo-device for this request on H7, starts a tape server daemon on H2, and communicates to the NetTAPE client on H7 that the mount has been completed.
6. The NetTAPE client creates the device special file, in this example `mytape`, in the end-user home directory, and the `cltalloc` command exits with the message:

```
cltalloc: VOL004 now mounted on /u/adsmres2/mytape
```


7. The user opens the tape device, by entering an AIX command, for example, `dd`, or by starting an application program. The user does not address the device name, for example `/dev/rmt0`, but uses the special file name returned by NetTAPE, in this case `$HOME/mytape`. A `dd` command to write file `data.file` on the tape would be:

```
dd if=data.file of=$HOME/mytape obs=1024
```

8. The NetTAPE client checks the volume label. If the volume label does not match the requested volume, the NetTAPE client sends an *intervention required* message to the operator. Let us assume that the correct volume has been mounted.
9. The user command or application writes a file on the tape. Data flows directly from H7 to H2 without going through H1. This is a standard labeled tape so the tape is positioned after the label.
10. The user issues:

```
cltunall
```

The NetTAPE tape server daemon on H2 unloads the volume and notifies the NetTAPE status daemon on H1 to release the volume. The NetTAPE status daemon sends a message to the operator on H3 to unmount the volume, and deletes the pseudo-device on H7 and the tape server daemon on H2.
11. The operator puts the cartridge back in its rack and clicks on the **Clear message** button (not shown).
12. NetTAPE marks the drive as free and ready for allocation.

Notes:

1. The user must know which type of device is suitable for the volume he or she wants to use. In our case it was simple because the default pool was suitable. If the default pool were not suitable, the user would have had to know which device pool to specify.
2. If the user does not issue the `cltunall` command, the cartridge will remain on the drive. It will be up to the operator to detect that either the drive is inactive or the requester has disappeared and perform a reset on the device to free it.

4.4.3 Central Operations

With NetTAPE, using the XWindows GUI or command line interface, one or several workstations can:

- Answer mount and dismount messages for manual drives

- Manage mount queues, that is, change the sequence of queued mount requests for a device or device pool
- Move mount requests from one device to another device (the devices can be on different hosts)
- Update the NetTAPE device configuration
- Monitor NetTAPE operations for error situations, for example

There are no mount or dismount operator actions when dealing with tape libraries because the mount and dismount commands are sent directly to, and handled by, the automated library. The NetTAPE operator interface, when dealing with automated libraries, is used only for monitoring error situations.

You can define several operator domains, each controlling a subset of your tapes and libraries, thus allowing tape operations management by device type, physical location of the drives, or any other criteria. These operator domains do not have to correspond to device pools. The domain for a given operator session is selected when you start the operator command, either for the GUI or the command line interface. When using the GUI, an operator can switch from one domain to another.

The GUI gives operators a much easier way to work and provides a comprehensive online help facility. The GUI also provides capabilities that are not available in the command line interface. For example, with the GUI, the operator can immediately see a mount request, whereas with the command line interface, mount messages are not displayed synchronously but instead must be queried by command. The command line operator would have to enter, at regular intervals:

```
cltopr -m
```

to see an outstanding mount request. This is cumbersome for the operator and results in longer mount times for the end users.

Only the GUI provides the *Move Mount* function, which allows an operator to move a mount request from one device to a different device (in the same device pool). The different device can be on a system other than the system on which the device was originally requested. Move mount can be very useful, for example, when a tape device or host is not operational.

The GUI can also show the operator information about the number of blocks that the end user is transferring to the tape volume. The time interval between updates of the number of blocks (tape position information) is an installation-definable parameter.

The other important function that is not available through the operator command line interface is *Mount Queue* management, which enables an operator to change the priority of mount requests queued on a device pool.

We recommend that you use the command line interface only if you want to automate tasks by means of shell scripts.

4.4.4 Device Pools

With NetTAPE, you group your tape devices, either manual or library drives, into NetTAPE pools, which can have up to three levels (see Figure 18 on page 96):

- First level—device pools

Device pools group tape drives of the same device type. This is a first level pool. For example, you could define a device pool, MAN3590_1, for one set of manual 3590s.

- Second level—esoteric device pools

Esoteric device pools are groups of device pools. For example, you could have a pool, SJ_MAN3590, that could include the MAN3590_1 device pool and another device pool, MAN3590_2. SJ_MAN3590 would contain all of the manual 3590s in your San Jose location. This is a second-level pool.

- Third level—esoteric device pools.

An esoteric device pool can contain other esoteric device pools.

For example, a pool, ALL_MAN3590, could include SJ_MAN3590 and TUC_MAN3590. ALL_MAN3590 would contain all of your manual 3590 drives in your San Jose and Tucson locations. ALL_MAN3590 is a third-level pool. LIB3590_1 is a device pool for your automated 3590s in library number one (for example, a 3494).

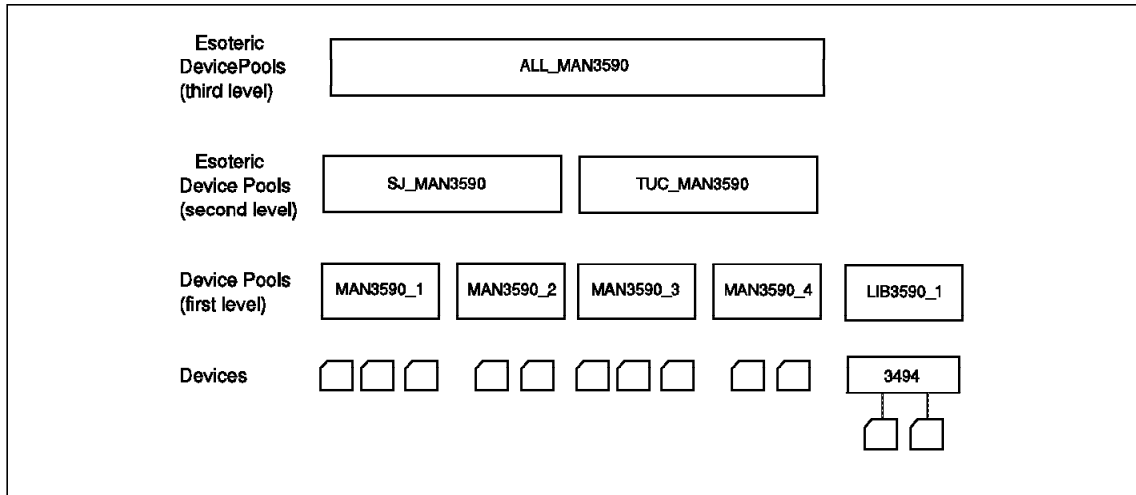


Figure 18. NetTAPE Pool Example

Knowing which hosts control which tape devices is irrelevant for pool definitions.

One of the pools is designated as the default device pool. The default can be either a device pool or an esoteric device pool.

The rules for pool definitions are:

- In a device pool or an esoteric device pool (second or third level) all of the devices must be of the same type.
- Esoteric pools cannot overlap, that is, a device pool can be a member of one and only one esoteric device pool.
- In the first release of NetTAPE, libraries can be addressed only as first level device pools.

4.4.5 Tape Device Allocation

The NetTAPE status daemon manages all of the tape drives in the network and handles tape drive allocation for user requests. A user can request one of the following allocations:

- Nothing special, in which case NetTAPE allocates a drive from the default pool
- A specific pool, which can be a device pool or an esoteric device pool
- A specific drive, in which case NetTAPE tries to allocate that drive and will be successful if the drive is free.

The user can, in addition, specify the host where he or she wants the allocated drive to be attached.

NetTAPE tries to select a drive on the same host as the client if there is a free drive; otherwise, it selects a free drive on some other tape server.

4.4.6 Tape Libraries

The NetTAPE Tape Library Connection program integrates tape library management with manual tape operations in a way that is transparent to end users. NetTAPE supports scratch and private volume categories in the 3494 and 3495 ATLDSS.

4.4.7 Mount Queues

If the type of drive a user requests is not available, NetTAPE, based on a user option, either queues the mount request or returns to the user with a nonzero return value. Then it is up to the user to reissue the mount at a later time.

If mount requests are queued, the NetTAPE operator, through the GUI, can manage the mount queues, that is, change the sequence of queued mount requests.

Figure 19 is the window the operator sees after clicking on the mount queue option (not shown).

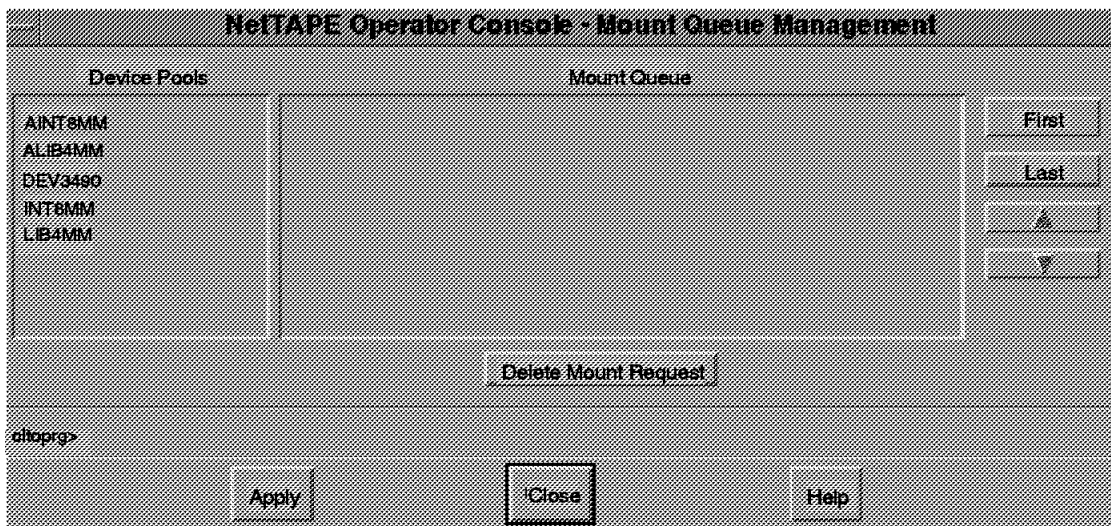


Figure 19. Mount Queue Management Window with Device Pools

A double-click on the device pool name, for example, the DEV3490 pool, displays the Mount Queue Management window (as shown in Figure 20 on page 98) where the operator can change the mount queue sequence.

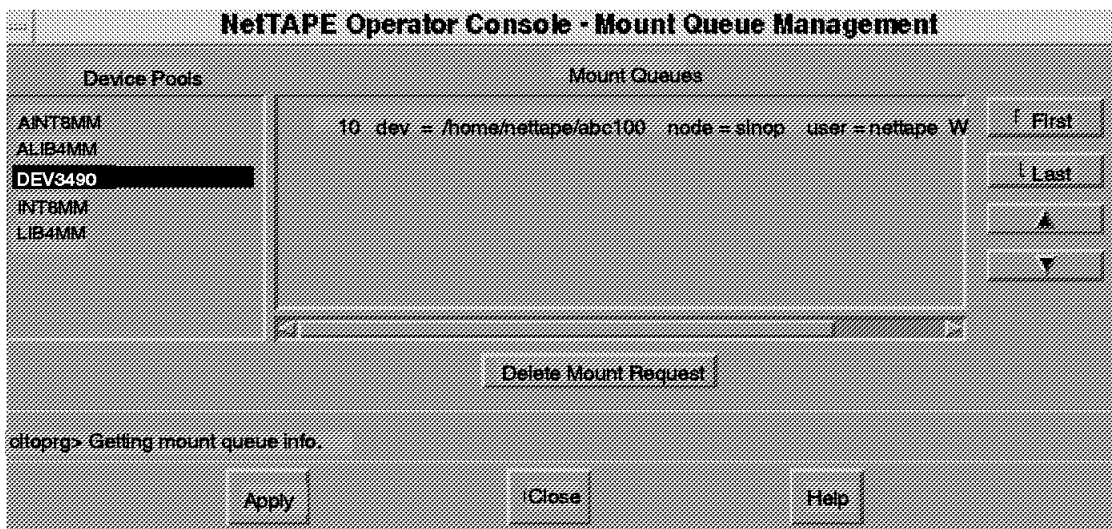


Figure 20. Managing a Mount Queue for Device Pool DEV3490

Please refer to Chapter 6, “NetTape Administrator Operations and End-User and Application Use” on page 127 for details on mount queue operation.

If necessary, the operator can reset a tape device, but note that resetting cancels the application using that device.

4.4.8 Remote Tape Access

With the NetTape remote tape access functions, users can access tape drives connected to another host other than their login host as if the tape drives were locally attached.

NetTape uses a pseudo-device driver on the client host that communicates with the NetTape tape server through TCP/IP to transfer the tape data.

A tape pseudo-device is associated with a tape client daemon, `cltcIntd`, which starts a tape server daemon, `tapesrv`, on the remote host. The tape server daemon passes the data between its remote client and its local tape device. A user must have a valid AIX user ID on the remote tape server node.

ADSM cannot use the remote tape function because some of the functions (actually `ioctl`s) that ADSM requires are not available through the NetTAPE pseudo-device interface.

4.4.9 Tape Labels

NetTAPE provides label processing for tape drives supported through the AIX enhanced tape drivers, such as, the 3480, 3490, and 3590 devices.

At the time of writing, the AIX 4.1 version of NetTAPE contains tape label support for 8 mm, 4 mm, QIC, and 9-track tape drives. However, there are limitations, such as only single volume data sets are supported and users must know how much data they will write to the tape so that enough room is left at the end of the tape for the trailer labels. A refresh of the AIX 3.2.5 version of NetTAPE is planned that will also contain this support.

Tape label support allows processing of tape volumes created on mainframes, for example, MVS, and creation of tape volumes on AIX that you can process on MVS.

The label types supported by NetTAPE are:

- ANSI X3.27-1987 (ISO 101-1986)
- IBM EBCDIC standard labels
- Unlabeled

NetTAPE label processing includes NetTAPE client verification that the correct tape volume has been mounted before the application reads from or writes to the volume. Label processing also gives the user access to existing label information for the file record format.

NetTAPE does not check the expiration date when mounting a tape volume. If, however, you put an expiration date in NetTAPE and later mount the tape on MVS, for example, the necessary information is kept in the label.

Bypass label processing (BLP) is a mode where NetTAPE does not do any label checking even though the tape volume has a label. Because all checks are bypassed in BLP mode, the application could overwrite any tape volume. Therefore, the user ID must be specifically authorized for BLP mode in the NetTAPE configuration file.

NetTAPE also supports:

- Generation numbers

NetTAPE saves the generation number specified by the user in the tape label so that you will have access to it when using that tape volume on MVS.

- Expiration date and retention period

An expiration date and retention period can be specified. NetTAPE stores them in the label but does not use the information when mounting a tape volume.

- Multiple files on a tape volume

- On read

The `cltalloc` command enables users to specify which file number they want to access on a tape volume.

- On write

At file creation time, the `cltopen` command enables a user to write multiple files on one volume.

4.4.10 Record Formats

NetTAPE enables a user to specify the record formats to be used on a tape volume when reading or writing.

NetTAPE supports record formats U (unformatted), F (fixed length), and FB (fixed length blocked). Mainframes can process tapes created with these record formats. Note that the V (variable) format is not supported.

See the *IBM NetTAPE and NetTAPE Tape Library Connection User's Guide and Reference* for detailed information about using tape formats.

4.4.11 End-User Interfaces

Users access NetTAPE functions in two ways:

- Command line

The main NetTAPE user commands enable you to:

- Ask to mount a volume on a drive. Parameters can be specified for additional options, such as label processing and block and record formats.
- Access the next file on a mounted tape volume
- Ask for a tape volume to be dismounted and the drive unallocated.

- Application programming interface

You can use the NetTAPE application programming interface (API) subroutines from a C or Fortran program. The API provides the same

functions as the command line interface, as well as basic tape functions. Use the NetTAPE API if you want to integrate NetTAPE functions with your own tape applications.

4.4.12 Support for the ADSM External Library

The ADSM External Manager, in conjunction with NetTAPE, enables you to share tape devices between ADSM and other applications. However, ADSM does not presently support writing on remote tapes, with or without NetTAPE.

Chapter 7, “ADSM Tape Management with NetTAPE” on page 153 contains more information about the ADSM External Library support in NetTAPE and how to use it.

4.4.13 Volume Information (VFILE)

NetTAPE provides a way of storing information (in a file) about a tape volume. When you allocate a tape device to access a labeled tape volume through NetTAPE, you can specify, on the `cltalloc` command, the `-o VFILE` option. When using labeled tapes, you must also specify a data set name.

The VFILE option tells NetTAPE to create a file, whose name is the data set name. The file contains information that will be useful the next time you use that same volume, such as:

- Pool name
- Volume serial numbers, both internal and external
- File sequence number on the volume

Note that VFILES are not systemwide entities; they are created at user request time in the home directory of the user (by default).

The VFILE facility cannot be used as a tape catalog, even a very rudimentary one; it is merely a way of saving some of the NetTAPE allocation parameters.

Please refer to Chapter 6, “NetTAPE Administrator Operations and End-User and Application Use” on page 127 for details on the how to use VFILES.

4.5 Platforms Supported

In this section we review NetTAPE support for workstation hardware and software, networks, tape devices, and libraries.

4.5.1 Workstation Hardware

NetTAPE is supported on all IBM RISC System/6000 systems, including IBM RISC System/6000 SP nodes. The AIX 4.1 version of NetTAPE is supported on RISC System/6000 Symmetrical Multiprocessing (SMP) models.

4.5.2 Workstation Software

NetTAPE supports AIX Versions 3.2.5 and 4.1.4.

NetTAPE is a prerequisite for the NetTAPE Tape Library Connection product.

4.5.3 Network Support

TCP/IP communication is used between NetTAPE hosts; any type of TCP/IP on AIX network is supported, including Ethernet, Token-Ring, FDDI, the High Performance Switch, Fiber Channel, and ATM.

4.5.4 Tape Device Support

Here are the IBM and AMPEX NetTAPE supported tape devices as well as unsupported tape devices and functions:

4.5.4.1 IBM Tape Devices

Here are the IBM supported tape devices:

- The IBM 3420 Magnetic Tape Unit
- The IBM 3480 Magnetic Tape Subsystem, including support for the automatic cartridge loader (ACL) in serial mode
- The IBM 3490 Magnetic Tape Subsystem Enhanced Capability Models C10, C11, and C22, including support for the ACL in serial mode
- The IBM 3490 Magnetic Tape Subsystem Enhanced Capability Models E01 and E11, including support for the ACL in serial mode
- The IBM 3490E Magnetic Tape Subsystem, including support for the ACL in serial mode
- The IBM 3590 High Performance Tape Subsystem, including support for the ACF in random or serial mode

The following devices are supported but are limited to no-label format:

- The IBM 7206 Model 005 External 4 mm Tape Drive (and its internal version)
- The IBM 7207 Model 012 External 1/4-inch Cartridge Tape Drive (and its internal version)
- The IBM 7208 Model 011 External 8 mm Tape Drive (and its internal version)

- The IBM 9348 Model 12 1/2-inch 9-track

4.5.4.2 AMPEX Tape Devices

The following Ampex tape device is supported but not warranted by IBM:

- The Ampex DST 310 Tape Drive

4.5.4.3 Unsupported Tape Devices and Functions

You can use devices other than those mentioned in sections 4.5.4.1, “IBM Tape Devices” on page 102 and 4.5.4.2, “AMPEX Tape Devices.” You can (almost) perform label processing on devices where NetTAPE does not officially support it.

Non-IBM SCSI Tape Devices: As long as non-IBM SCSI tape devices use the ost driver, NetTAPE will not see any differences, and these devices should work.

Label Processing on SCSI Drives: The reason there is a difference in label support between devices is that the standard AIX ost device driver does not allow you to write after the logical end of volume, thus precluding trailer label processing. Therefore, you cannot provide full label support using the ost driver whereas the 3490 tape drivers can handle end-of-volume situations.

To avoid end-of-volume problems, there is logic in the AIX 3.2.5 version of NetTAPE (at the time of writing) to disallow label processing and record format options, if the device type is 8 mm, 4 mm, QIC, 9TRK, or ost.

As discussed in section 4.4.9, “Tape Labels” on page 99, the AIX 4.1 version of NetTAPE contains tape label support for 8 mm, 4 mm, QIC, and 9-track tape drives. However, there are limitations, such as only single volume data sets are supported and users must know how much data they will write to the tape so that enough room is left at the end of the tape for the trailer labels. A refresh of the AIX 3.2.5 version of NetTAPE is planned that will also contain this support.

There are also provisions in NetTAPE to bypass the logic that checks for labels and record formats. Therefore you can effectively perform label processing and record formats on these types of devices, **provided that you ensure that your data does not span multiple volumes**. You can bypass the processing by specifying the device type backwards: mm8, mm4, CIQ, KRT9, or TS0. This bypass process is not documented, except in this redbook. We tested label processing and record format handling on an 8 mm drive by specifying mm8, and it worked! The device type names are not case

sensitive. NetTAPE internally converts the device type names to upper case.

4.5.5 Tape Library Support

The NetTAPE Tape Library Connection product supports the following tape libraries:

- IBM 3494
- IBM 3495
- StorageTek 4400 Automated Cartridge Subsystem
- StorageTek PowderHorn 9310/9311 Automated Cartridge Subsystem
- StorageTek WolfCreek 9360 Automated Cartridge System

Notes:

- The StorageTek products listed are supported but not warranted by IBM.
- The 3590-B11 ACF is not listed under libraries in the NetTAPE Tape Library Connection documentation, although it effectively works like a (small) library if you use the ACF in random mode. You do not need the NetTAPE Tape Library Connection product to support the 3590-B11 in random mode.

The connections supported for the 3494 and 3495 libraries are the same connection supported under AIX, that is:

- 3494
 - LAN and serial attachment for SCSI-connected tape drives (3490 and 3590)
 - ESCON library connection for ESCON-attached 3490 drives
 - Parallel channel library connection for Block Multiplexer Parallel Adapter (BMPA)-attached 3490 drives
- 3495
 - ESCON library connection for 3490 drives
 - Parallel channel library connection for BMPA-attached 3490 drives

Chapter 5. NetTAPE Installation and Configuration

In this chapter we describe NetTAPE planning, installation, and configuration. We provide a checklist of the steps to take and decisions to make when planning your NetTAPE environment. We give an overview of NetTAPE installation; we do not try to duplicate the formal product install books. We provide an overview of the NetTAPE daemons and their associated files and logs. We describe configuring NetTAPE, in particular, the four configuration files. We conclude with the additional steps required to install and configure tape libraries.

5.1 Planning

Before configuring NetTAPE, you must carefully analyze your requirements, tape drive and library configurations, operational needs and constraints, and network.

Here are the logical steps to take:

1. Review your tape drive and library configuration.

You may want to reconsider your tape drive and library configuration to optimize tape access to tape drives on the basis of the new NetTAPE functions, for example, remote tape access. Your tape drive and library configuration will also have an impact on your network load.

2. Select the tape devices to be included in the NetTAPE configuration.

Determine which tape devices you want NetTAPE to manage. You can choose to have NetTAPE manage only those tape drives that are used by your important applications or have to be shared, not the small internal drives on your different systems.

3. Group the NetTAPE-managed devices into device pools.

All drives in a pool must be of the same device type. Thus, for example, you can group different models of 8 mm drives into one pool, but that you cannot have one pool with both 3480 and 3490 drives.

You do not have to put all of your drives of the same type into the same device pool. For example, you may want to have one pool for tape drives located in your main machine room and another pool located in a more secure room. All of the drives can be of the same type, but you separate them to better control which tape requests go where.

Remember that pools are used for drive allocation and primarily to make it easier for applications to obtain access to tape devices. With

pools, a user or application issues a mount request specifying only the name of the pool, rather than a specific device name.

4. Group the device pools into esoteric device pools.

Esoteric device pools give you additional freedom for grouping device pools to ease device allocation.

If you have one device pool of a certain type, you may want to define an esoteric device pool for that device type. If you must add another device pool later on, an esoteric device pool makes the addition easier because your users can continue specifying the esoteric device pool name and automatically get access to the new device pool. li. Select a default device pool.

Once you have determined the device pools for your configuration, you have to select a default device pool. If a mount request omits a device pool name (c|talloc without -P), a device from the default device pool is used to satisfy the request.

You can use an esoteric device pool as your default device pool.

5. Define operator domains.

Defining more than one operator domain enables you to distribute tape mounting operations. Typically, you define multiple domains if you have tape drives in different locations. If you define one operator domain per location, the operators will see only the messages related to the drives for which they are responsible.

6. Define operators and operator authorities.

NetTAPE has three levels of operator authority. These are applicable to the GUI only.

Starting from the most powerful, the levels are:

Administrative Operators These operators can stop the NetTAPE status daemon.

Operators These operators manage the tape drives and respond to mount and unmount requests.

View Operators These operators can display the GUI but cannot perform any action.

Note: If you do not define a user ID for an operator category, you are effectively authorizing everybody to access the corresponding function. Therefore, do not leave these lists empty!

7. Decide whether you want to queue mount requests.

You decide centrally whether NetTAPE will do mount queuing or not. Mount queue processing specifies whether mount requests are to be queued when a device is not available to satisfy the request. If you do not select to use mount queuing, mount requests will be immediately returned to the requester if there is no suitable free drive.

Even if you allow mount queuing, end users can still specify whether they want their mount requests queued or not.

8. Configure your NetTAPE server hosts.

You must decide which host will run the NetTAPE status daemon. Your tape and library configuration will determine which hosts are the NetTAPE tape and library servers. You should also define a user ID for the NetTAPE servers because it is better not to run them under root.

9. Determine your NetTAPE clients.

You must ensure that users who require access to tape devices on a given system have valid UNIX user IDs on that system. This is the only, albeit indirect, way of controlling access to NetTAPE because NetTAPE clients are not registered in any NetTAPE configuration file.

5.2 Installation

The NetTAPE products, NetTAPE and the NetTAPE Tape Library Connection, consist of three separate packages:

- `nettape`

This is the NetTAPE server code as well as the GUI operator code. It is about 15 MB.

- `clnettape`

This is the common CLIO/S NetTAPE code and is about 4.5 MB.

- `nettape_tlc`

This is the NetTAPE Tape Library Connection code and is about 2.3 MB

After installation, the NetTAPE server code is placed in the `/usr/lpp/nettape` directory, the NetTAPE client code in the `/usr/lpp/clnettape` directory, and the NetTAPE library code in the `/usr/lpp/nettape_tlc` directory.

Installing the NetTAPE code on multiple hosts can be accomplished with different approaches: You can choose to mount the `/usr` file system globally by using a network file system (NFS), or you can install copies of the code on each host that needs it.

The different NetTape components require access to the NetTape code as shown in Table 2 on page 108.

<i>Table 2. NetTape Components and Code Packages</i>			
NetTape Component	nettape	nettape_tlc	clnettape
NetTape status daemon	Yes	No	Optional
NetTape library server	Yes	Yes	Optional
NetTape tape server	Yes	No	Yes
NetTape operator	Yes	No	No
NetTape client	Yes	No	Yes

You will probably install the client code on all of your NetTape server nodes because you probably have NetTape clients on those hosts as well.

5.3 NetTape Server User IDs

The NetTape components communicate through rexec and use the .netrc file for automatic login on the different hosts. The .netrc files contain passwords, so if you choose to run the NetTape servers under root, the root password will appear in the clear in the files, which is generally something to be avoided, for security reasons.

Therefore, you should run the NetTape servers under a specific user ID whose password is less sensitive than root's. This does not cause any particular problem because the NetTape server user ID's do not require special AIX authority. In our examples, we choose to call the user ID *nettape*. You must put the user ID and password of the NetTape users, including the NetTape servers, in the different .netrc files.

You must run the NetTape status daemon and all of the NetTape tape and library servers under that same user ID.

5.4 Configuration

We describe the basic customization required without going through every parameter. Please refer to the *&net tug*. for additional information.

For each NetTAPE component we show its daemon and its associated files and logs. The main customization required is defining four configuration files, three that are unique to NetTAPE and one that is not unique to NetTAPE and is used by other AIX applications. We describe the four files in detail, including sample definitions for different tape configurations.

We conclude by describing the additional installation and configuration steps required for tape libraries.

5.4.1 Daemons

Figure 21 on page 110 shows the NetTAPE node daemons with their associated files and logs.

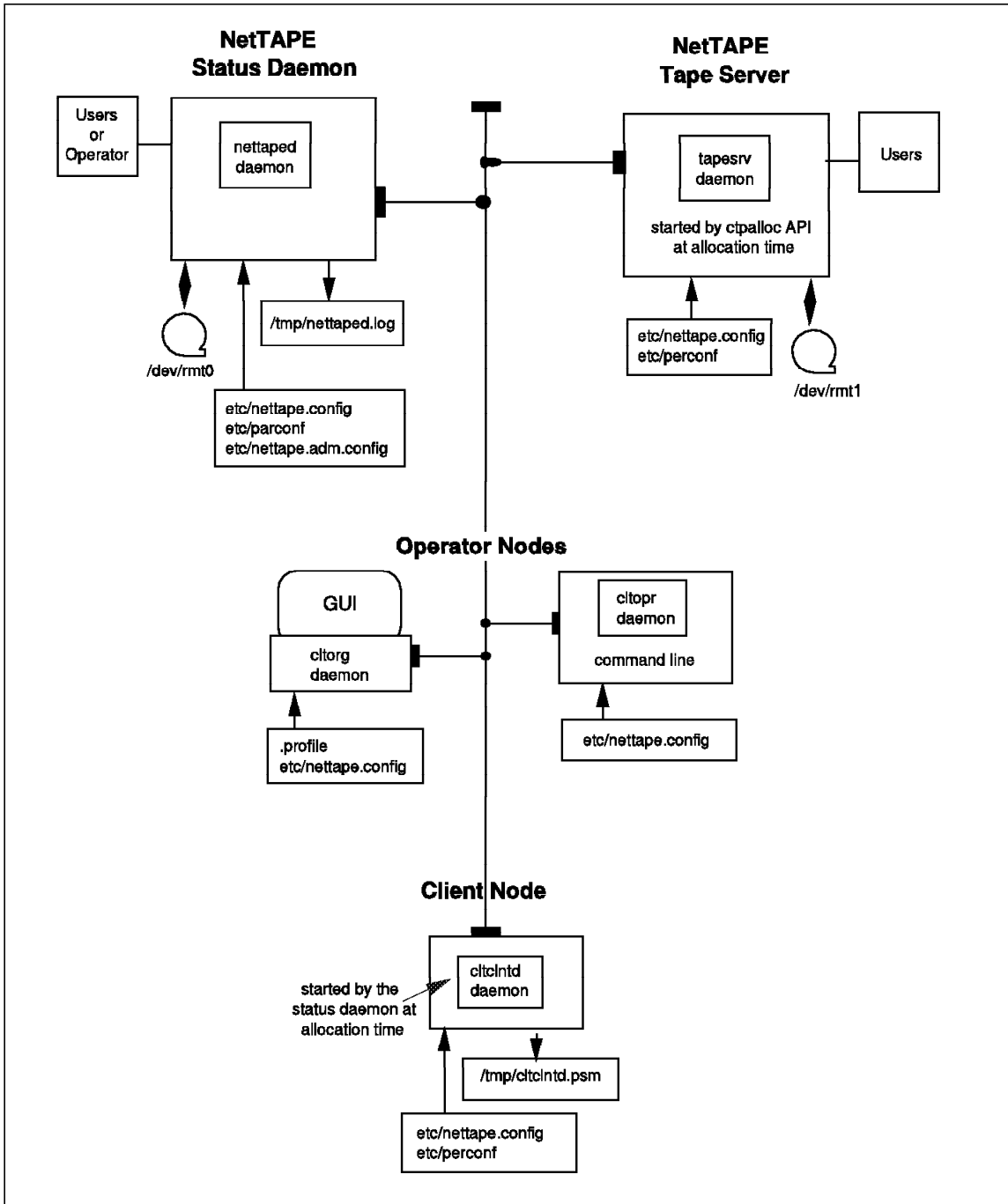


Figure 21. NetTape Daemons and Their Associated Files and Logs

The master daemon for NetTAPE is the status daemon, `nettaped`. It is sometimes also referred to in the NetTAPE manuals, with the IEEE 1244 Mass Storage Reference Model terminology, as the *physical volume library* daemon. The status daemon listens at a known socket, waiting for user applications to contact it.

`CltcIntd` is the tape client daemon, started by the NetTAPE status daemon, which in turn starts the `tapesrv` daemon to handle functions related to remote tape access. The `cltcIntd` daemon runs on the AIX system where the user invokes the `cltalloc` command.

The tape operator daemons are `cltorg` for the GUI, and `cltopr` for the command line interface.

5.4.2 Configuration Files

NetTAPE is controlled by four configuration files, three of which reside in `/etc`, which you will edit to adapt to your environment. The NetTAPE configuration files are used so that applications using NetTAPE functions do not have to know the details of the network connections, hardware technologies, or software protocols being used to support the tapes.

To set up your NetTAPE configuration, you have to customize the following NetTAPE configuration files:

- `parconf`
- `nettape.config`
- `.netrc`
- `nettape.adm.config`

The first three files are customized with the same definitions no matter which tape device configuration you use. The `nettape.adm.config` file is customized differently for each type of tape drive or library.

Before installing these files, create a NetTAPE system administrator user ID, for example, `nettape`. Make sure this is a valid user ID on all of the physical tape device hosts that you plan on defining in the `nettape.adm.config` file.

Let us describe the four configuration files in more detail:

- `/etc/parconf`

The `parconf` file defines the NetTAPE status daemon and the NetTAPE server machines, including their TCP/IP paths. The `parconf` file is the first place the NetTAPE status daemon looks for configuration

information. Parconf contains the information NetTAPE uses to locate nodes and to know how to connect to them.

The parconf file must be accessible by all of the NetTAPE components including the tape servers, operators, and clients. You can make parconf accessible by having it remotely mounted through NFS or by duplicating the parconf file on every host in the NetTAPE network.

- /etc/nettape.config

The nettape.config file defines the host name and port number that must be used to address the NetTAPE status daemon. The file is used to specify information needed to configure the NetTAPE status daemon. The default port number used by the nettaped daemon is 3401. For all existing TCP/IP applications, we recommend registering the port number of the status daemon in the AIX /etc/services file for your network.

The nettape.config file must be accessible on each system where a NetTAPE component (server, operator, and client) runs, either by using a remote mount or by replicating the file on each system.

- .netrc

The netrc file is not a NetTAPE-specific file; it is used by other AIX applications such as rsh. NetTAPE uses .netrc for the same purpose as rsh, to provide user ID and authorization information.

The NetTAPE status daemon uses .netrc to start the tape and library servers, so you must define all of the tape and library servers in the .netrc file on the NetTAPE status daemon and NetTAPE tape and library server hosts.

- /etc/nettape.adm.config

The nettape.adm.config file is the main configuration file where you define most of the parameters that control the NetTAPE configuration, including:

- Tape devices and device pools, their operator domains, AVR capability, and initial drive state definitions
- Esoteric device pools
- Operator IDs, what they are authorized to do, and for which domain
- Mount queue characteristics
- Libraries
- Default scratch and private categories for libraries
- Actions if there is a tape label mismatch

Only the nettape.adm.config file is required to reside on the NetTAPE status daemon node. The other NetTAPE components get the

information that they need in their conversations with the NetTAPE status daemon node.

Note: There are no specifications for explicitly authorizing end users to use NetTAPE. Authorizing end users is done indirectly because only users who have a user ID on the nodes where the tape drives are attached will be able to access these drives. Users do not have to have an ID on the other hosts in the NetTAPE network.

5.4.3 Client User IDs

You also must define user IDs on the NetTAPE tape servers so that the NetTAPE clients can access tape drives on these hosts.

All of the client user IDs on the NetTAPE tape server require access to the `.netrc` file pointing to the NetTAPE tape servers. The `.netrc` file contains the password of the user, so ensure that the file is readable only by the client user ID on the NetTAPE tape server. NetTAPE checks that the `.netrc` file has file permissions defined as `-rw-----`.

5.4.4 Updating NetTAPE Code

When you install a new level of NetTAPE, you will have to redo a step you did at initial installation, namely:

- Change ownership of the client daemon on the NetTAPE servers

```
chown nettape:adm /usr/lpp/c1nettape/bin/cltc1ntd
```

5.4.5 Sample Configuration Files

Figure 22 on page 114 is our test environment; we use it to illustrate the configuration file examples in this section.

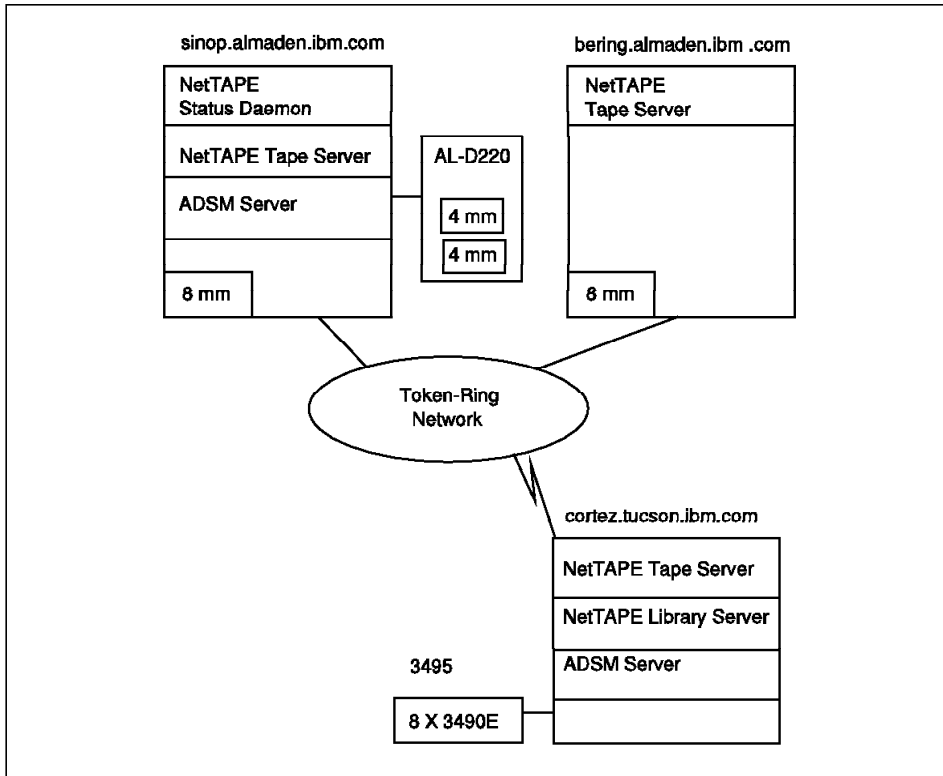


Figure 22. Sample NetTape Configuration

5.4.5.1 Sample parconf File

The syntax for the configuration information needed for the NetTape status daemon to locate and connect to nodes is:

```
nickname type protocol node_address
```

where you specify:

- type=1 for a RISC System/6000 or SP node
- protocol=1 for TCP/IP communications
- node_address = the internet address (long or short text, or dotted decimal). If you use the dotted decimal IP address, you can prefix it with an at sign (@) to indicate that NetTape should not use the name server to resolve the IP address.

Here is an example of customizing the parconf file for our sinop and bering nodes.

```
sinop 1 1 @129.33.160.102 # TCP/IP, int 8mm + AIWA D220 4mm
bering 1 1 @129.33.160.145 # TCP/IP, int 8mm
```

5.4.5.2 Sample nettape.config File

Here is an example of how to define the nettape.config file:

1. Define the NetTAPE status daemon host and port values

```
stdsmn.hostname      sinop      # host on which the nettaped daemon runs
stdsmn.portno        3401      # port address
```

2. Define the tape activity status interval

The tape activity status interval is the number of tape records between updates of the tape position information on the NetTAPE GUI's device status display. The tape activity status is useful for operators who are monitoring tape drive activity. However, there is a performance cost associated with the data gathering.

You specify the tape activity status interval with an integer of a minimum value of 100. If you omit this option, tape position information is neither collected nor displayed.

```
client.tapepos      1000
```

5.4.5.3 Sample .netrc File

You can create a .netrc file under the system administrator user ID or the nettape user ID's \$HOME directory.

For example, the .netrc file on the NetTAPE status daemon host in the nettape user ID's home directory might look like this:

```
machine bering.almaden.ibm.com login nettape password secret
machine cortez.tucson.ibm.com login nettape password secret
```

This example assumes that the NetTAPE tape and library servers run under the nettape user ID on the remote servers.

You do not have to do the reverse, that is, define the NetTAPE status daemon in the NetTAPE tape and library servers .netrc files.

Issue this AIX command:

```
ls -ae .netrc
```

to see the AIX file permissions for the .netrc file.

```
-rw----- 1 nettape adm 108 Mar 15 15:10 .netrc
```

Each user must have a `&home/.netrc` file defined. To avoid creating `.netrc` files for each user, using the `nettape` user ID, issue the following commands on each client where NetTAPE will be used or where a NetTAPE device is attached.

```
chown nettape /usr/lpp/clnettape/cltclntd
chmod ug+s   /usr/lpp/clnettape/cltclntd
```

The first command, `chown nettape`, changes the ownership of the `.netrc` file to the `nettape` user ID, and the second command, `chmod ug+s`, sets the user ID and group permission to the owner of the `.netrc` file, in this case, the `nettape` user ID. Now when `cltclntd` is executed, the effective group and user IDs are set to those of `nettape` instead of the user calling `cltclntd`. Therefore the client gets the `nettape` user ID permissions.

You must run these commands if you run the NetTAPE tape and library servers on a user ID other than `root`, because the `.netrc` file, as distributed, is set up to run under `root`.

Note

One problem with these commands and technique (if the `cltclntd` daemon is owned by a nonroot user ID such as `nettape`) is that users cannot access local tape devices with NetTAPE's pseudo tape device because NetTAPE changes the ownership of the physical tape device special file to the user ID that requested the tape. Remote tape access is not affected because the `tapesrv` daemon is owned by `root` and has the *sticky bit* turned on so that `tapesrv` can write to any file.

At the time of writing there is no known fix. The NetTAPE direction is to remove the dependency on the `.netrc` file in a future NetTAPE release.

5.4.5.4 Sample `nettape.adm.config` File

The `nettape.adm.config` file contains most of the NetTAPE configuration definitions that are used by the NetTAPE status daemon.

Here are the parameters you must define in the file:

- NetTAPE port number for the `clutil` utility daemon. This daemon is internal to NetTAPE. You can use any port number except the one used by the NetTAPE status daemon.

We recommend registering the port number in the `/etc/services` file for your network.

- User access control list

- Device pool definitions
- Esoteric device pool definitions
- Default device pool
- Mount queue processing definitions
- AVR definitions

Below we look at how to define these parameters for different tape configurations: manual tape drives, ACL or ACF 3480, 3490, and 3590 drives, and shared drives between two host systems.

Manual Tape Drives: This example can be used to define all 4 mm, 8 mm, and QIC tape drives:

- NetTAPE port number for the internal utility daemon

```
stsdmn.util.portno    3402        # port address for clutil daemon
```
- User access control list

The access control list specifies the user IDs authorized to perform certain tasks. The levels of authority from highest to lowest are:

- Administrative functions (such as starting and stopping NetTAPE)
- Operator functions
- Authorization only to view the NetTAPE GUI

```

                                User IDs
                                -----
stsdmn.operators.adm    operator adsmres3 root

stsdmn.operators        operator adsmres3 root adsm

stsdmn.operators.view   operator lilian  adsmres3 cybres root

```

- Device pool definitions

Here are the values you can specify for operation mode, the operator domain of the device pool, and the state of the device:

- op_mode = manual (for manual tape mount operation)
- options
 - od = *i* (operator domain) where *i* is an integer
 - od = up (device is in up state)
 - od = down (device is in down state)
 - od = resv (device is in reserve state)

- od = avr (turns on the automatic volume recognition flag)

We include two definitions for the same physical drive, the internal 8 mm drive on sinop. The first definition uses the ADSM driver, and the second definition uses the ost driver:

```
hostname.device pool_name device_type tape_name op_mode options
-----
sinop.device      aint8mm   mm8       /dev/mt2   manual    op=1,up
sinop.device      int8mm    8mm      /dev/rmt0  manual    op=1,down
```

Here are our definitions for our 4 mm library:

```
hostname.device pool_name device_type tape_name op_mode options
-----
sinop.device      alib4mm   4mm      /dev/mt0   manual    up
sinop.device      alib4mm   4mm      /dev/mt1   manual    up
sinop.device      lib4mm    4mm      /dev/rmt1  manual    down
sinop.device      lib4mm    4mm      /dev/rmt2  manual    down
```

Here are our definitions for the internal 8 mm drive on bering:

```
hostname.device pool_name device_type tape_name op_mode options
-----
bering.device     int8mm    8mm      /dev/rmt0  manual    down
bering.device     aint8mm   mm8      /dev/mt0   manual    up
```

- Esoteric device pool definitions

```
                esot_dev_pool dev_pools
-----
stdsmn.esoteric    all8mm   int8mm
stdsmn.esoteric    all4mm   lib4mm
```

- Default device pool

The default device pool can be an esoteric device pool.

```
                dev_pool
-----
stdsmn.devpool.default int8mm
```

- Mount queue processing definitions

If you want mount requests to remain queued until they can be satisfied, omit the last two keywords, mount.queue.scan.interval and mount.queue.max.wait.

```
mount.queue          yes # Allow queuing of mount requests
mount.queue.scan.interval 5 # interval in minutes for scanning
mount.queue.max.wait 30 # max tape client wait time. (in minutes)
```

- AVR scans values

```
avrsmn.interval      5 # interval in minutes for timeout scanning
```

ACL or ACF 3480, 3490, and 3590 Drives: The device pool definition section of the `nettape.adm.config` file differs slightly for ACL or ACF 3480, 3490, and 3590 drives. The `op-mode` parameter is used by the NetTAPE status daemon to determine which devices are capable of handling ACL or ACF devices. The other sections of the `nettape.adm.config` file are defined the same as for manual tape drives.

- Device pool definitions

Here are the `op-mode` values you would use for ACL or ACF devices:

- `ibmac1` (for ACLs with sequential access)
- `ibmacf` (for ACFs with random access)

The following devices have ACL turned on:

hostname.device	pool_name	device_type	tape_name	op_mode	options
gb980.device	3480_1	3480	/dev/rmt3	ibmac1	
gb980.device	3480_1	3480	/dev/rmt4	ibmac1	down
v08n03.device	3490_1	3490	/dev/rmt1	ibmac1	

The following devices also have AVR mode turned on:

hostname.device	pool_name	device_type	tape_name	op_mode	options
k80n05.device	3490_2	3480	/dev/386	ibmac1	od=1,up,avr
k80n05.device	3490_2	3480	/dev/387	ibmac1	od=1,up,avr

The following 3590 devices have ACF turned on:

hostname.device	pool_name	device_type	tape_name	op_mode	options
k80n05.device	3590	3590	/dev/rmt0	ibmacf	
epacrs8.device	3590	3590	/dev/rmt5	ibmacf	

- Esoteric device pool definitions

	esot_dev_pool	dev_pools
stsdmn.esoteric	a113480	3480_1
stsdmn.esoteric	a113490	3490_2
stsdmn.esoteric	a113590	3490_3

- Default device pool

The default device pool can be an esoteric device pool.

	dev_pool
stsdmn.devpool.default	a113490

Shared Drives between Two Host Systems: The device pool definition section in the `nettape.adm.config` file differs slightly for drives that will be shared between two host systems. The other sections of the `nettape.adm.config` file are the same as for manual drives.

A *devices suffix* is specified to identify shared devices. Devices that have the same suffix number are shared between two hosts. In this example, `k80n06.device.1` and `k80n03.device.1` represent one physical drive that is shared between two hosts. Device `k80n06.device.2` and `k80n03.device.2` also represent one physical drive that is shared between two hosts.

- Device pool definitions

```

hostname.device pool_name device_type tape_name op_mode options
-----
k80n06.device.1 3490_2      3490      /dev/518  ibmac1  od=1,up,avr
k80n06.device.2 3590_3      3590      /dev/519  ibmacf  od=1,up,avr

(Second host)
k80n03.device.1 3490_2      3490      /dev/518  ibmac1  od=1,up,avr
k80n03.device.2 3490_2      3490      /dev/519  ibmac1  od=1,up,avr

```

- Esoteric device pool definitions

```

                esot_dev_pool dev_pools
                -----
stsdmn.esoteric    a113490    3490_2

```

- Default device pool

The default device pool can be an esoteric device pool.

```

                dev_pool
                -----
stsdmn.devpool.default a113490

```

5.5 Additional Steps to Configure 3494, 3495 and STK Tape Libraries

The configuration of tape libraries requires some additional steps. We describe the overall steps for the 3494 and 3495 and STK tape libraries as well as the changes needed in the `nettape.adm.config` file.

5.5.1 Overall Steps

After you have installed the NetTAPE Tape Library Connection product, follow this procedure to configure the IBM 3494 and 3495 and STK tape libraries to work with NetTAPE:

- Select your tape library server node

Select a RISC System/6000 workstation on which the NetTAPE tape library daemon (cl3494d or clslod) will run. This workstation is the NetTAPE library server node. You have to run a separate NetTAPE tape library daemon for each library subsystem or partition and give it a unique library name. You will need the library name when you update the nettape.adm.config file.

- For the IBM 3494 or 3495 tape library with parallel or ESCON attachment feature.
 1. Install and configure the tape device driver. Install and configure the IBM parallel and ESCON tape device driver (mtd) on the NetTAPE library server node. See the *IBM Parallel and ESCON Tape Device Driver: Installation and User's Guide*, GA32-0311, for detailed information.

After you successfully configure the tape libraries, a library manager control point special file, lmc^{*}, is created in the /dev directory. Verify that this special file is created by using the command:

```
ls -l /dev/lmc*
```

2. Check that the tape library server has all of the tape devices configured.

Ensure that the library server node has all of the tape devices configured on it so that the cl3494d tape library daemon can issue tape mount requests for every tape device in the library. You can still attach the library tape devices to multiple workstations according to your workload and hardware configuration requirements, but the library server node must have all of the devices configured on it.

3. Configure the nettape.adm.config file as explained in 5.5.2, "Configuring the nettape.adm.config File" on page 124.

- For the IBM 3494 tape library with SCSI attachment feature

1. Install and configure the IBM tape device driver. Install and configure the IBM library device driver (atld) on the NetTAPE library server node. We recommend that you use the LAN configuration option of the IBM TLDD product. See the *IBM SCSI Tape Device Driver: Installation and User's Guide*, GC35-0154, for detailed information.

After you successfully configure the tape libraries, a library manager control point daemon, lmc^d, will run in the background, and a library manager control point special file, lmc^{*}, is created in the /dev directory. Verify that the daemon is running and the special file is created by issuing these commands:

```
ps -ef | grep lmcpd
ls -l /dev/lmcp*
```

2. With the LAN configuration option, the NetTAPE library server node is not required to have any library tape devices attached to it. The cl3494d library daemon can perform remote tape mount requests through TCP/IP.
 3. Configure the nettape.adm.config file as explained in 5.5.2, "Configuring the nettape.adm.config File" on page 124.
- For an STK tape library
 1. The NetTAPE STK tape library support is based on StorageTek's Client System Component (CSC) Developer's ToolKit Version 2.1. The CSC Developer's Toolkit is designed specifically for the development of remote tape library applications, which function as clients to the ACS library server. The remote procedure call (RPC)-based communication mechanisms are tailored to run on UNIX hosts other than Automated Cartridge System Library Server (ACSL) hosts.
 2. Select the NetTAPE library server node to start the storage server interface. After you designate a NetTAPE library server node on which to run the NetTAPE STK tape library library daemon (clsilod), start the storage server interface (SSI) process on the node. The SSI process establishes the necessary connection with the ACS Library Manager (ACSLM) located at a remote host.

To start the SSI process:

- a. After you install the NetTAPE Tape Library Connection product, change to the /usr/lpp/nettape_tlc/csc/bin directory.
- b. Invoke the t_startit.sh script file and respond to the prompts as shown below:

```
***** Welcome to t_startit.sh, Version 2.01 *****
```

```
This is the automated tester and startit script for the TOOLKIT.
Simply answer the questions which follow.
```

```
Executable live in: /usr/lpp/nettape_tlc/csc/bin
```

```
Would you like Multi-Host or Single-Host testing?
```

```
Enter one of the following followed by ENTER:
```

```
M Multi-host testing
S Single-host testing
```

X exit this script

Enter choice: M

>>> Select option M <<<

Would you like to define the server side or client side
for Multi-Host testing?

Enter one of the following followed by ENTER:

S Server side

C Client side

Enter choice: C

>>> Select option C <<<

The Remote Host Name is the name of the server which
has the ACSLS software running on it.

Enter Remote Host Name (CSI_HOSTNAME): splsvr0

The Remote Host Version number is the ACSLS Packet
Version level which the server is expecting.

Here are the valid choices:

ACSLS Release Number	Remote Host Version Number
----------------------	----------------------------

3.x	2
-----	---

4.x	3
-----	---

5.x	4
-----	---

Enter Remote Host Version (ACSAPI_PACKET_VERSION): 4

Starting /usr/lpp/nettape_tlc/csc/bin/mini_el...

Attempting startup of /usr/lpp/nettape_tlc/csc/bin/mini_el ...

Starting /z/others/ronglee/csc/ssi...

Attempting startup of PARENT for /usr/lpp/nettape_tlc/csc/bin/ssi...

SIGHUP received

Parent Process ID is: 14869

Attempting startup of /usr/lpp/nettape_tlc/csc/bin/ssi...

SIGHUP received

Parent Process #14869 EXITING NORMALLY

Initialization Done.

Do you want to see the processes created? (Y or N): y

```

16663 pts/0 S      0:00 /usr/lpp/nettape_tlc/csc/bin/ssi 14869 50004 23
17434 pts/0 R      0:00 grep /z/others/ronglee/csc
17683 pts/0 S      0:00 /usr/lpp/nettape_tlc/csc/bin/mini_el

```

3. Configure the nettape.adm.config file as explained in 5.5.2, "Configuring the nettape.adm.config File."

5.5.2 Configuring the nettape.adm.config File

These additional definitions are required in the nettape.adm.config file:

- The tape library
- The retry interval for tape library daemon
- Turning on the parameter for checking between the physical tape volume label and the requested tape volume label
- The device pool for the tape library
- The device address in the device pool if you are using an STK library
- The esoteric device pool for the tape library
- The default device pool for the tape library
- The default scratch and private tape categories

Here is an example for either the IBM 3494 or 3495 or STK tape libraries:

- Tape library definitions

The IBM 3494 with Block Multiplexer Channel Adapter (BMCA)/PCA or ESCON requires the cmd_port and synch_port parameters:

	lib_name	host	Library manager	
			ctl point	cmd_port asynch _port
stdmn.ibmaccs	ibm3494_1	cortez	/dev/lmcp0	/dev/lmcp0
stdmn.ibmaccs	ibm3494_2	k80n07	/dev/lmcp0	/dev/lmnp

Definitions for the STK Silo

	lib_name	host	Library manager	
			ctl point	cmd_port asynch _port
stdmn.stkaccs	stksilo_1	k80n06		

- Specify the retry interval for the tape library daemon (in minutes):
stdmn.library.retry 2

- Compare the physical tape volume label and the requested tape volume label.

You can either turn on volume label checking, turn it off, or ignore it.

```
stsdmn.library.volcheck yes
```

- Define the device pool for the tape library.

There are two new `op_mode` values for tape libraries:

- `ibmmacs` (for IBM 3494 or 3495 tape libraries)
- `stkacs` (for STK Silo tape libraries)

The following example shows both 3490 and 3590 tape devices attached to the same IBM 3494 library, `ibm3494_1`. In this case, each device type should have a unique device pool name assigned.

hostname device	pool name	device type	tape name	op_mode options
cortez.device	3490_2	3490	/dev/rmt4	ibmmacs od=1,down,ibm3494_1
cortez.device	3490_2	3490	/dev/rmt5	ibmmacs od=1,down,ibm3494_1
k80n07.device	3590_2	3590	/dev/182	ibmmacs od=1,down,ibm3494_1
k80n07.device	3590_2	3590	/dev/183	ibmmacs od=1,down,ibm3494_1

The following example shows devices attached to an STK Silo library. Silo tape drives require additional `dev_addr` information.

				ACS,LSM,PANEL,DRIVE			
				\	/		
k80n03.device	3490_3	3490	/dev/rmt0	STKACS	0,0,1,0	od=1,down,stk	stksilo_1
k80n03.device	3490_3	3490	/dev/rmt1	STKACS	0,0,1,2	od=1,down,stk	stksilo_1
k80n03.device	3490_3	3490	/dev/rmt2	STKACS	0,0,2,0	od=1,down,stk	stksilo_1

- Define the esoteric device pool for the tape library:

	esot_dev_pool	dev_pools
stsdmn.esoteric	all3490	3490_2 3490_3

- Default device pool

The default device pool can be an esoteric device pool:

dev_pool	
stsdmn.devpool.default	all3490

- Default scratch and private tape categories:

	dev_pool	cat_id	

stsdmn.devpool.cat.scratch	3490_2	FBC	#(1 to 0xFEFF)
stsdmn.devpool.cat.private	3490_2	F11	#(1 to 0xFEFF)
stsdmn.devpool.cat.scratch	3490_3	900	#(1 to 0xFEFF)
stsdmn.devpool.cat.private	3490_3	100	#(1 to 0xFEFF)
stsdmn.devpool.cat.scratch	3590_2	5D0	#(1 to 0xFEFF)
stsdmn.devpool.cat.private	3590_2	6D0	#(1 to 0xFEFF)

Please note that STK Silo tape libraries only support scratch tape categories.

Chapter 6. NetTAPE Administrator Operations and End-User and Application Use

In this chapter we describe some of the common administrator operations that use the NetTAPE command line interface and GUI. We also describe how an end user and application would use NetTAPE from a command line.

6.1 NetTAPE Administrator Operations

In this section we explain the tasks a NetTAPE administrator would perform from either a command line or GUI. The topics we cover are:

- How to start and stop the NetTAPE daemon
- Starting the NetTAPE GUI
- Typical operator tasks
- Occasional operator tasks
- Error handling

6.1.1 How to Start and Stop the NetTAPE Daemon

The AIX administrator starts and stops the NetTAPE product. Usually the AIX administrator would automatically start NetTAPE by using the standard AIX `etc/inittab` file, the same technique used to automatically start any other application.

6.1.1.1 Starting and Restarting NetTAPE

From the user ID of the NetTAPE administrator, the administrator can start the NetTAPE status daemon or the `net taped` daemon located on the same host. The following command shows you how to start the NetTAPE status daemon on the `sinop` host and put the log for the NetTAPE status daemon in `/home/nettape/pvl.log`:

```
cltopr -S pvl -l /home/nettape/pvl.log
```

You will receive the following message:

```
cltopr : Attempting to start pvl.  
Check process status for result.
```

If you do not use the `-l` option, the log and error messages are saved in the `/tmp/nettaped.log` file.

The log file after a normal NetTAPE start looks like this:

```
sinop/S 19:36:42 (pid=22102): - OpDmn domain id:1 not available
sinop/S 19:36:42 (pid=22102): - No Operator Daemon available
sinop/S 19:36:43 (pid=22102): - OpDmn domain id:1 not available
sinop/S 19:36:43 (pid=22102): - No Operator Daemon available
sinop/S 19:36:43 (pid=22102): - OpDmn domain id:1 not available
sinop/S 19:36:43 (pid=22102): - No Operator Daemon available
sinop/S 19:36:43 (pid=22102): - OpDmn domain id:1 not available
sinop/S 19:36:43 (pid=22102): - No Operator Daemon available
```

The log seems incorrect at first glance, but it does show a normal NetTAPE startup!

To start the same NetTAPE server on sinop from another host in the network, issue this command:

```
cltopr -S pvl -h sinop
```

6.1.1.2 Stopping NetTAPE

The command to stop NetTAPE stops all NetTAPE daemons and brings down the operator GUI. Before stopping NetTAPE, verify that there are no tape jobs running by issuing the `cltopr -q` command and searching for any devices with an *InUse* status. You can also verify that there are no tape jobs running by using the operator GUI because the operator GUI displays volume ID information if a tape drive is currently in use.

The administrator can stop NetTAPE on the same host, using the user ID of the NetTAPE administrator. To stop the NetTAPE server located on local host sinop, enter this command:

```
cltopr -H pvl
```

After executing the command, you receive the following confirmation message:

```
Do you want to Halt pvl daemon? [yØn]
```

If you enter *y*, the following message appears:

```
cltopr : Stop daemon request sent.
```

To stop the same NetTAPE server on sinop from another host in the network, issue this command:

```
cltopr -H pvl -h sinop
```

6.1.2 Starting the NetTAPE GUI

A script file, `xcltopr`, is provided to automate the NetTAPE operator GUI environment setup and all standard X options. In the `/usr/lpp/nettape/samples` directory, you can find the following files:

- NETTAPE

The NETTAPE file contains the NetTAPE Operator Console Motif/X Window system default configuration.

- font.choice

The font.choice file contains font choices for the NetTAPE operator console.

The `xcltopr` script file is located in the `/usr/lpp/nettape/bin` directory.

To start the GUI, issue:

```
xcltopr -display host_name:0
```

or

```
xcltopr -xrdp -display host_name:0
```

6.1.3 Typical Operator Tasks

In this section we describe some of the typical operator tasks that an operator would perform on a regular basis:

- Managing device pool mount queues
- Displaying outstanding operator messages
- Replying to an operator message
- Displaying the current allocation status

6.1.3.1 Managing Device Pool Mount Queues

The command line operator interface, `cltopr`, does not handle mount queue management, and it does not receive any intervention messages from the NetTAPE status daemon.

Each device pool and esoteric device pool has its own mount queue. Jobs, which NetTAPE queues in first-in first-out (FIFO) order, are placed on the mount queue for the device pool or esoteric device pool specified (or taken as the default) on the allocation request.

Only the NetTAPE GUI's mount queue management window lets you view each device pool mount queue and reorder or delete queued jobs. Esoteric device pool mount queues cannot be viewed or changed. Figure 23 on page 130 shows the operator GUI. In the Mount Message Area (in the center of the window), you can see two mount messages queued.

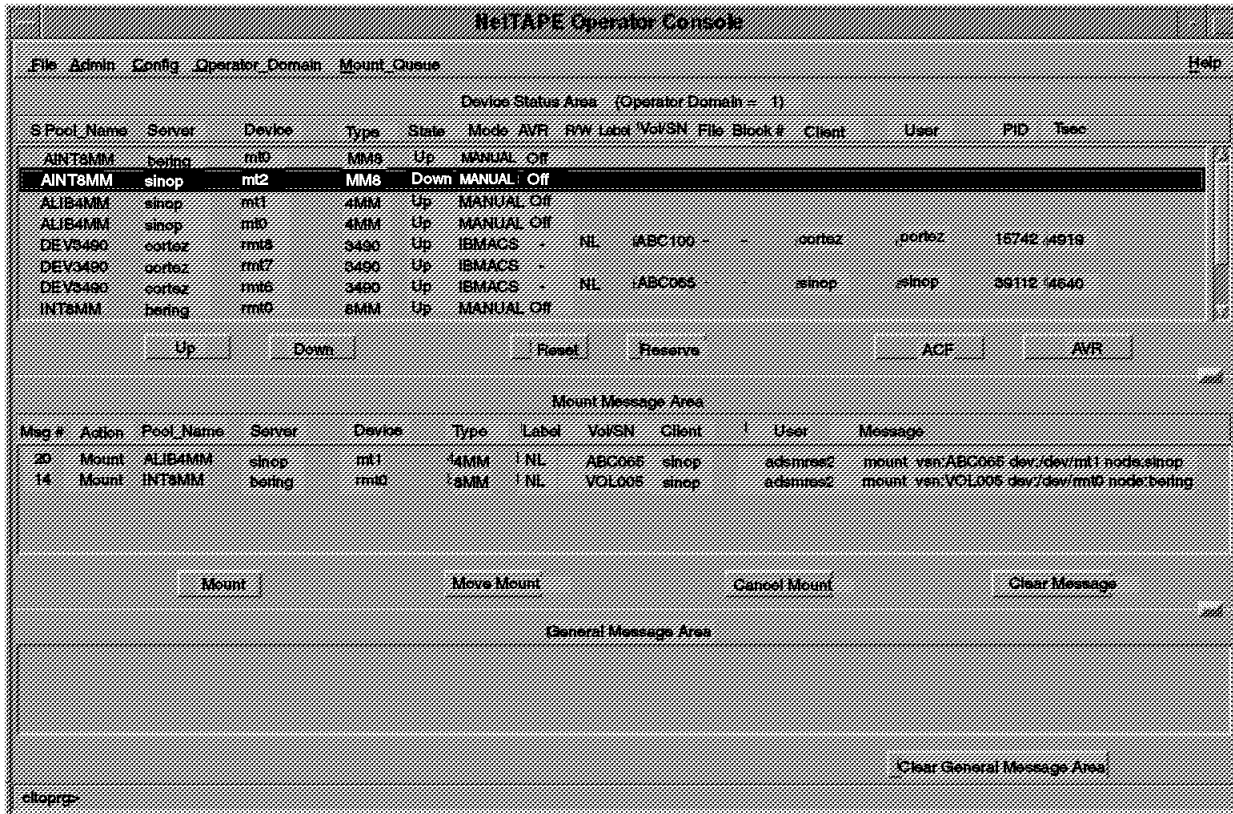


Figure 23. NetTape Operator GUI Mount Message Area

6.1.3.2 Displaying Outstanding Operator Messages with the Command Line

All outstanding mount requests are listed in the GUI Mount Message Area.

To display the outstanding mount requests by using the command line interface, use the `cltopr -m` command. The command line interface does not provide mount queue management and it does not receive any intervention messages from the NetTape status daemon. The `cltopr -m all` command determines all of the outstanding mount messages.

The result displayed from the `cltopr -m` command is:

```

All outstanding mount request...
Mount request : 20
  pool=ALIB4MM node=sinop device=/dev/mt1 type=4MM
  label=NL volser=ABC065 client=sinop user=adsmres2
  msg=mount vsn:ABC065 dev:/dev/mt1 node:sinop
Mount request : 14
  pool=INT8MM node=bering device=/dev/rmt0 type=8MM
  label=NL volser=VOL005 client=sinop user=adsmres2
  msg=mount vsn:VOL005 dev:/dev/rmt0 node:bering

```

To display a specific outstanding operator message, for example, 14, use this command:

```
cltopr -m 14
```

The result from this command is:

```

Mount request : 14
  pool=INT8MM node=bering device=/dev/rmt0 type=8MM
  label=NL volser=VOL005 client=sinop user=adsmres2
  msg=mount vsn:VOL005 dev:/dev/rmt0 node:bering

```

6.1.3.3 Replying to an Operator Message

The operator can perform several actions in response to an operator message. The typical actions are:

- Mount

The mount request is satisfied by either a device being allocated as requested or the mount request being queued to wait for an available device.

- Move

The mount request is moved to a different device from the device requested in the message. You cannot move a mount to a different device pool or swap two pending mount requests.

Moving a mount request does not produce a new mount message. When you move a mount request, NetTAPE assumes that the tape volume has been mounted on the device where you moved the mount. The end user sees a different response when a mount is moved as compared to a normal (not moved) mount.

When the operator confirms a mount and the device is not ready, the user gets:

```

cltalloc: openx - errno=46 Device not ready
cltalloc: open failed with errno= 46 Device not ready

```

whereas when the operator moves the mount, the user gets:

cltalloc: SCRATCH is now mounted on /u/adsmres3/mytape
even if the device is not ready. The message shows up first on the operator GUI when NetTAPE sends an intervention required message. If the operator cancels the mount, the user sees:

```
adsmres3@bering> tar -cvf /u/adsmres3/mytape .  
a .  
a ./profile 1 blocks.  
a ./tfile 3316 blocks.  
tar: 0511-195 An error occurred while writing to the storage media.  
: There is an input or output error.
```

- Cancel

The mount request is discarded.

The three push buttons for performing these actions are on the operator GUI below the Mount Message Area (see Figure 23 on page 130).

Here are some examples of replies to an operator message through the command line interface:

- To cancel message number 15 and ask for confirmation:

```
cltopr -r 15 -a cancel
```

The result of this command is:

```
Do you want to cancel message 20? [yøn] n  
cltopr : Message 20 not delete.
```

- To cancel message number 20 with no confirmation:

```
cltopr -r 20 -a cancel -f
```

The result of this command is:

```
cltopr : Request sent to cancel message 20.  
cltopr : Cancel mount request for mount #20 sent to status daemon.
```

- To satisfy a mount request

```
cltopr -r 14 -a mount
```

The result of this command is:

```
Provide (6 characters) Volume Serial Number if mount request has specified.  
SCRATCH volume, otherwise hit ENTER to continue. >>  
cltopr : Request sent to mount message 14.  
cltopr : Mount reply for mount #14 sent to status daemon.
```

- To satisfy a scratch tape with a specific volume:

```
cltopr -v scr001 -r 22 -a mount
```

The result of this command is:

cltopr : Request sent to mount message 22.
 cltopr : Mount reply for mount #22 sent to status daemon.

6.1.3.4 Displaying Current Allocation Status

The current allocation status is permanently displayed on the upper section of the operator GUI. Refer to Figure 23 on page 130 to see the allocation status.

From the command line, use the `cltopr -q` command to display all of the pool names used in the NetTAPE configuration and control the state of the tape devices.

The output of the `cltopr -q` command is:

```
Current status for all known devices...
```

Pool_name	Host_name	Device_name	Dev_type	State	Op_mode
===== AINT8MM	sinop	/dev/mt2	MM8	Up	MANUAL
AINT8MM	bering	/dev/mt0	MM8	Down	MANUAL
ALIB4MM	sinop	/dev/mt1	4MM	Up	MANUAL
ALIB4MM	sinop	/dev/mt0	4MM	Up	MANUAL
DEV3490	cortez	/dev/rmt6	3490	Up	IBMACS
DEV3490	cortez	/dev/rmt7	3490	Up	IBMACS
DEV3490	cortez	/dev/rmt8	3490	Up	IBMACS
INT8MM	bering	/dev/rmt0	8MM	Down	MANUAL
INT8MM	sinop	/dev/rmt0	8MM	InUse	MANUL
LIB4MM	sinop	/dev/rmt1	4MM	Down	MANUAL
LIB4MM	sinop	/dev/rmt2	4MM	InUse	MANUAL

To return allocation status information for a specific device, such as `/dev/mt1`, use this command:

```
cltopr -d /dev/mt1 -q
```

The output of the command is:

```
Current status for all /dev/mt1 devices ...
```

Pool_name	Host_name	Device_name	Dev_type	State	Op_mode
===== ALIB4MM	sinop	/dev/mt1	4MM	Up	MANUAL

To return allocation status information for a specific device, such as `/dev/mt0`, on a specific host, such as `sinop`, use this command:

```
cltopr -d /dev/mt0 -h sinop -q
```

The output of the command is:

```

Current status for all /dev/mt1 devices ...
Pool_name      Host_name      Device_name    Dev_type  State  Op_mode
-----
ALIB4MM       sinop         /dev/mt0      4MM      Up    MANUAL

```

6.1.4 Occasional Operator Tasks

In this section we describe some of the occasional operator tasks that an operator would perform on an exception basis:

- Adding and deleting devices from NetTAPE
- Changing the state of a device
- Initiating a temporary 3590 ACF scan

6.1.4.1 Adding and Deleting Devices from NetTAPE

To add or delete devices from the NetTAPE configuration, you can use the command line interface or the GUI. The change can be either temporary or permanent. A temporary change is processed immediately but is valid only until NetTAPE stops. A permanent is valid even after a restart of NetTAPE.

To make a permanent change you would use the `-U` parameter on the `cltopr` command. Any daemons currently running are unchanged. For the currently running daemons to pick up the configuration changes, you must restart the NetTAPE status daemon.

Examples of a temporary change using the command line are:

```

cltopr -d /dev/rmt0 -h host2 -p 3480pool -t 3480 -o "IBMACL" -c add
cltopr -d /dev/rmt0 -h host2 -p 3480pool -c delete

```

Examples of a permanent change using the command line are:

```

cltopr -d /dev/rmt0 -h host2 -p 3480pool -t 3480 -o "IBMACL" -c add -U
cltopr -d /dev/rmt0 -h host2 -p 3480pool -c delete -U

```

You can use the operator GUI to change the device configuration. Figure 24 on page 135 shows the operator GUI. To add or delete a device, first select a device from the top part of the screen. Then select Config from the action bar and choose either add or delete a device from the drop-down menu (not shown). If you choose delete a device, the Delete Tape Device window (as shown in Figure 24 on page 135) is displayed.

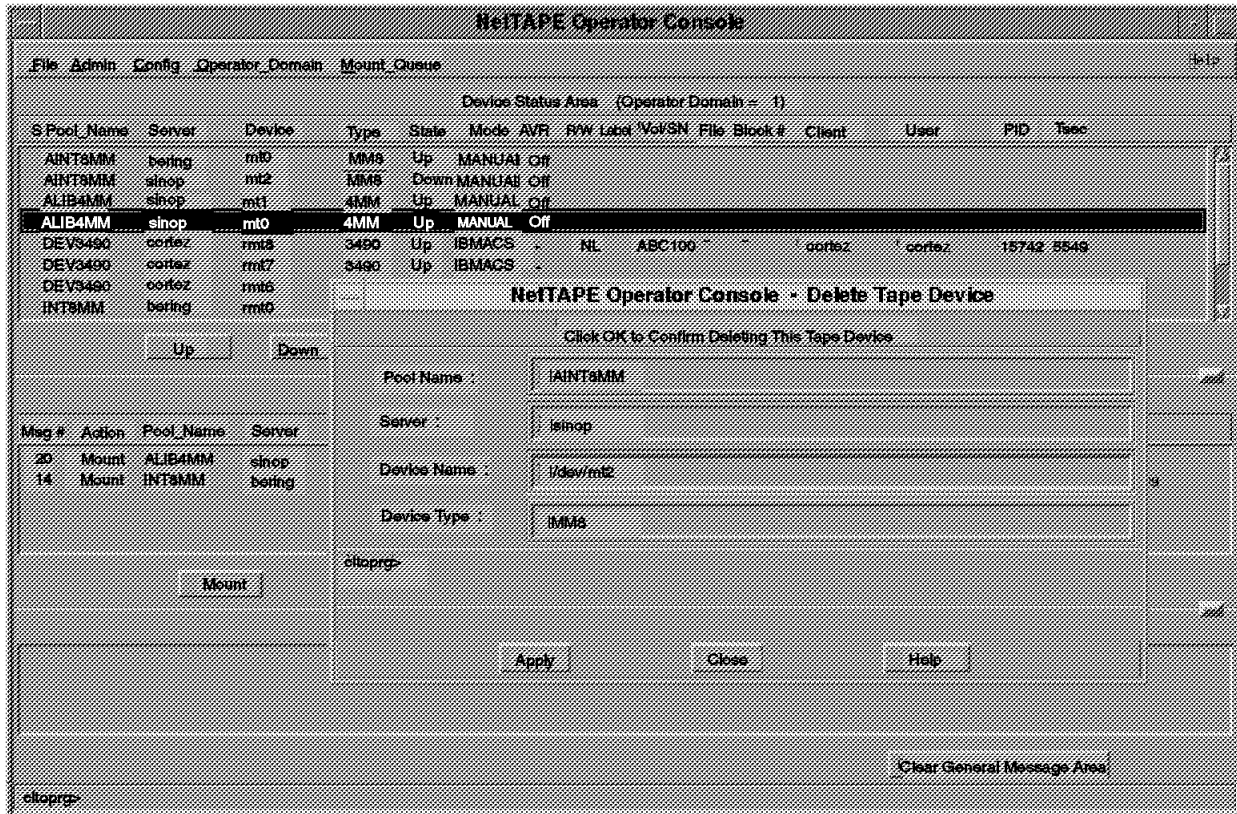


Figure 24. NetTAPE Operator GUI for Deleting a Device

6.1.4.2 Changing the State of a Device

You can change the state of a NetTAPE device to reserve, up, or down.

- Reserve

Reserving a device makes it unavailable for allocation and prevents the device from being assigned to a user. If the device is not in use, it goes immediately into the reserved state. If the device is in use, it is put into a *pending reserved* state and goes into a reserved state at the next end of tape or when the user invokes `ctunall` or `ctpunall` (see 6.2, “End-User and Application NetTAPE Use” on page 137). The reserved device can be used again by changing its state to up (that is, available for allocation). You might put a device into reserved state when maintenance is required.

Here is an example of reserving a tape device:

```
ctopr -d /dev/rmt0 -h host2 -c reserve
```

- Up

Here is an example of changing a device from the reserved state to the up state:

```
cltopr -d /dev/rmt0 -h host_name -c up
```

- Down

Here is an example of changing a device from the up to the down state (that is, not available for allocation):

```
cltopr -d /dev/rmt0 -h host_name -c down
```

6.1.4.3 Initiating a Temporary 3590 ACF Scan

You may want to scan the ACF magazine, for example, to put a new magazine in the tape drive. For more details see section 6.2.9, “Using ACL or ACF 3480, 3490, and 3590 Tape Libraries” on page 150.

Here is an example of specifying a scan of the ACF magazine on a 3590 in random volume access mode:

```
cltopr -d /dev/rmt0 -h host_name -s on
```

Here is an example of halting an ACF scan and placing the stacker in sequential mode:

```
cltopr -d /dev/rmt0 -h host_name -s off
```

6.1.5 Error Handling

Two operator tasks that would be performed only during error situations are resetting a tape device and stopping the NetTAPE client daemon.

6.1.5.1 Resetting a Tape Device

Generally you would reset a device to correct an error condition, such as a hardware problem or hanging job. Resetting a tape device stops the job running on the device and returns the device to the available state. Resetting a device forces the cancellation of the job running on it.

Here is an example of resetting a tape device:

```
cltopr -d /dev/rmt0 -h host_name -c reset
```

6.1.5.2 Stopping the NetTAPE Client Daemon

Under normal circumstances, NetTAPE starts and stops the NetTAPE tape client daemon as needed.

If NetTAPE terminates abnormally, some tape devices might remain allocated even though they are no longer in use. You have to manually stop

the tape client daemons that still have allocated devices. Follow these steps:

1. Find any tape client daemons that are still running:

```
ps -ef | grep cltclntd
```

2. Stop each tape client daemon:

```
cltterm -f /dev/pst0
```

The `cltterm` command signals the pseudo-device to terminate the local client, that is, the `cltclntd` daemon. You must have root authority to use the `cltterm` command.

6.2 End-User and Application NetTAPE Use

In this section we explain how an end user or application would use NetTAPE from the command line. There is no GUI for end users or applications. We review all of the end-user commands, going from the simple to the complex. We include many examples that you can use in EXECs.

The topics we cover are:

- Determining the available device pools for your allocation request
- Allocating a tape device or sending a tape mount request
- Unallocating a tape device
- Obtaining information about tape volumes used
- Using the VFILE
- Using multivolume data sets
- Using tape labels on 3480, 3490, and 3590s
- Setting an application block size
- Using ACL or ACF 3480, 3490, and 3590 tape libraries
- Writing special applications, using the NetTAPE API

6.2.1 Determining the Available Device Pools for Your Allocation Request

Before requesting a tape mount (that is, allocating a tape drive) with the `cltalloc` command, the end user or application can determine which device pools are available. You can get this information by using this command:

```
cltopr -q
```

The output of the command is:

```

Current status for all known devices...
Pool_name      Host_name      Device_name    Dev_type    State  Op_mode
=====
AINT8MM        sinop          /dev/mt2      MM8         Up     MANUAL
AINT8MM        bering        /dev/mt0      MM8         Down   MANUAL
ALIB4MM        sinop          /dev/mt1      4MM         Up     MANUAL
ALIB4MM        sinop          /dev/mt0      4MM         Up     MANUAL
DEV3490        cortez        /dev/rmt6     3490        Up     IBMACS
DEV3490        cortez        /dev/rmt7     3490        Up     IBMACS
DEV3490        cortez        /dev/rmt8     3490        Up     IBMACS
INT8MM         bering        /dev/rmt0     8MM         InUse  MANUAL
INT8MM         sinop         /dev/rmt0     8MM         Down   MANUAL
LIB4MM         sinop         /dev/rmt1     4MM         InUse  MANUAL
LIB4MM         sinop         /dev/rmt2     4MM         Down   MANUAL

```

You can check important information before starting an allocation. For example, this output shows you that some 4 mm and 8 mm tapes are not available (that is, they are in a down state). You can use this command in an EXEC to check whether there are any changes in your NetTAPE configuration before starting an allocation.

6.2.2 Allocating a Tape Device or Sending a Tape Mount Request

The allocation of a tape device, that is, the mounting of a tape device, is the most important command for end users and applications. In this section we explain the most important aspects about the command.

There are actually two steps in the allocation process:

1. Dynamically allocating a tape device from the device pool, esoteric device pool, or default device pool specified
2. Opening the tape file for reading, writing, or updating (reading and writing)

The `ctalloc` and `ctopen` commands correspond to the above steps in the process.

6.2.2.1 Allocating a Tape Device

We illustrate all of the allocation functions with examples, starting with the most basic way to use `ctalloc`.

If you issue:

```
ctalloc
```

the operator receives a message to mount a scratch tape and specify an external volume ID or label (VOI001). After the operator mounts the tape, you receive the following message:

Scratch Vo1001 is now mounted on /home/test/mytape

The application or user is locked from performing any other commands between the `ctalloc` and the receipt of the message.

NetTAPE looks in the `nettape.adm.config` file for a locally attached device in the default pool. If one is not available, NetTAPE assigns another device in the pool.

After the mount, the application or user can start any tape commands, for example:

```
tar -cvf mytape $HOME/tape_stuff
```

NetTAPE provides a device special file for the virtual tape and by default the name of the special file is `mytape`. In a normal AIX situation (without NetTAPE), the command would be:

```
tar -cvf /dev/rmt0 $HOME/tape_stuff
```

For hints and tips for using NetTAPE with the basic AIX backup commands (`tar`, `cpio`, `dd`, and `backup/restore`), see Appendix F, “NetTAPE Hints and Tips” on page 239.

It is important to understand the default parameters for the allocation command and the options to overwrite the default parameters. The defaults parameters are:

- The default name of the device special file for the virtual tape device you want NetTAPE to create is `$HOME/mytape` in the `$HOME` directory. You can overwrite this name using the `-f device` parameter. You can specify the name as either a fully qualified path name or as a name relative to the current directory, for example, `/home/test/sap_tape` or `-f sap_tape`.
- The label type is `NL` (no label). You can overwrite this parameter by using the `-l AL` parameter (for an ANSI label) or the `-l SL` parameter (for an EBCDIC standard label).
- You receive a scratch tape without an external label.

In our example, the operator chooses an external volume ID or label for the scratch tape. You obtain the same result if you use the `-v scratch` parameter (that is, you ask for a specific scratch tape).

You can overwrite the parameter by using the `-v volume` parameter. You specify the external volume serial number, which means that you

are asking the operator to mount a specific tape. See section 6.2.6, “Using MultiVolume Data Sets” on page 145 for more details.

- The tape record format is U (unformatted).

If you are using AIX tape commands, such as tar and cpio, do not overwrite the tape record format. The actual block size written to the tape is controlled by the AIX tape command or application through write() and pwrite() calls. NetTAPE ignores the SMIT block size and resets it to zero.

- The tape mode for the scratch tape is -Mw (write mode)

You can overwrite the default by specifying -Mr (read tape mode).

- The tape recording format (density and compression) is the format defined in SMIT when the device is configured.

You can overwrite this parameter by using -Fb (for basic, uncompact) mode or -Fc (for compacted) mode.

- You write or (read) the first file on tape. The default parameter is -s 1.

If you have to skip one or more files, you must specify the file sequence number in the -s parameter. For example, to skip the first file you would specify -s 2.

- No data set name is used.

You can specify a data set name by using the -d dsname parameter. The term data set comes from MVS, but in fact with AIX a data set represents a file.

The data set name parameter is mandatory for labeled tapes and when you issue an allocate with the VFILE parameter. For details about using VFILE, see section 6.2.5, “Using the Volume Information File” on page 144.

- The owner of the file written on the tape is the user ID executing the allocation command.

- NetTAPE does not use a special file to save the tape management information, that is, it does not use the VFILE.

You can use the -o “allocparms” parameter to specify the following information:

- OWNER=owner

Use this parameter to overwrite the owner (user or application) of the tape volumes.

- VFILE

This parameter asks NetTAPE to create a VFILE, when you write on a tape, for the file specified in the -d dsname parameter. By default, the VFILE is saved in the \$HOME/.nettape directory. The VFILE name in the directory is the file name provided by the -d dsname parameter. You can change the directory by specifying the VDIR=dir parameter.

- After the allocate command you wait for the tape to be mounted by the operator.

If you do not want to wait for the tape mount, for example, if the device is not immediately available, you can specify the -n parameter. If the operator has prepared a number of tapes for the night, you can directly check whether the correct tapes have been prepared by using the -n parameter.

Some default parameters are specific for MVS tape formats:

- The tape record format is U (unformatted).

You can overwrite this parameter by using -RF (for fixed format) or -RFB (for fixed block format). With -RF or -RFB you can specify the length (in bytes) for logical records on the tape by using the -L record_length parameter. You can also specify the size of the blocks or records by using the -B blocksize parameter.

- No expiration date

You can specify the expiration date for a *new* data set by using the -e yyddd or -r dddd parameters.

Now let us review some more complex allocation requests:

- Allocating a specific device on a specific host

To allocate a specific device (510) on the specified node (host_xx), you can use either of the following commands:

```
cltalloc -D /dev/510 -h host_xx
cltalloc -D /dev/510@host_xx
```

- Allocating a specific device on a specific host within a specific pool

For NetTAPE to look for the specified device on the specified host, the device must be defined as a member of the specified pool in the nettape.adm.config file. The allocate command is:

```
cltalloc -D /dev/510 -h host_name -P 3490
```

- Allocating any device on a specific host within a specific pool

NetTAPE looks for any device on the specified host that is defined in the specified pool in the nettape.adm.config file. The allocate command is:

```
cltalloc -h host_xx -P 3490
```

- Allocating any device on a specific host

NetTAPE looks for any device on the specified host that is defined in the default device pool in the `nettape.adm.config` file. The `allocate` command is:

```
cltalloc -h host_name
```

- Allocating any device within a specific pool

NetTAPE looks in the `nettape.adm.config` file for a locally attached device in the specified pool. If one is not available, NetTAPE assigns another device in that pool (local or remote). The `allocate` command is:

```
cltalloc -P 3490
```

6.2.2.2 Opening a Tape File

Use the `cltopen` command when you want to continue using the previously allocated tape drive to do additional tape I/O with modified parameters.

For example, use `cltopen` to:

- Specify a different file name
- Reposition the tape between each file on the same volume
- Change the allocation parameters
- Mount a different volume on the previously assigned tape drive

All `cltopen` parameters are identical to the `cltalloc` parameters, except for the allocation parameters used to select a tape: `-h` host-name, `-P` devpool, `-D` device, and `-n`.

Here is a sample `cltopen` requesting that a different tape volume, `myvol2`, be mounted on the previously assigned tape drive, `$HOME/mytape`:

```
cltopen -f $HOME/mytape -v myvol2
```

6.2.3 Unallocating a Tape Device

The `cltunall` command closes the allocation and releases the tape. It is important to unallocate the tape if you no longer need it. No other users can use the tape if you do not unallocate it.

Here are a couple of examples of unallocating a tape device:

- To unallocate device special file `$HOME/mytape` created by the `cltalloc` command:

```
cltunall
```

- To unallocate device special file `/home/sap/tape01`:

```
cltunall -f /home/sap/tape01
```

The `cltunall` command displays the following messages on completion:

```
cltunall: Tape on /dev/pst1 is now unallocated.
```

Here is an example of how you can put different files on the same tape. The virtual device is accessed with the device special file `mytape`. When issuing multiple `tar` commands to write multiple files on a tape, `cltopen` must be issued between each `tar` command. The steps to write the two files to tape and then to verify the files by reading them from the tape are:

1. Allocate a 3494 tape device in a specific pool with a file named `dataset1`.

```
cltalloc -P 3494 -f mytape -d dataset1 -s 1
```

2. Write the first file to the tape, using `mytape.1` to avoid a tape rewind.

```
tar -cvf mytape.1 file1
```

3. Open the tape with a second file named `dataset2`.

```
cltopen -f mytape -d dataset2 -s 2
```

4. Write `file2` to the tape.

```
tar -cvf mytape file2
```

5. Read the two files from the tape.

Open the tape to read the first file named `dataset1`.

```
clltopen -f mytape -d dataset1 s 1
```

6. Restore the first file from the tape.

```
tar -xvf mytape
```

7. Open the tape with a second file named `dataset2`.

```
cltopen -f mytape -d dataset2 s 2
```

8. Restore the second file from the tape.

```
tar -xvf mytape
```

9. Unallocate the tape device.

```
cltunall -f mytape
```

Please note that the data set name is not required unless you are using a labeled tape.

6.2.4 Obtaining Information about Tape Volumes Used

The `cltvols` command returns information about the tape volumes used by the file or data set that is currently in use. For example, if you ask for a scratch tape without an external label (sticker), the operator chooses the

external volume ID or label for the scratch tape and manually writes it on the sticker. The `cltvols` command enables you to know the external label written by the operator. The `cltvols` command writes information about the tape volumes used to standard output.

Table 1 shows when to use `cltvols` and why.

<i>Table 3. When to Use cltvols</i>	
When	Why
Reading from and writing to tapes	To confirm the volume ID
Writing to a scratch tape	To learn the volume ID
Writing to a multivolume tape	To confirm the volume ID because the write operation spanned a volume

Here is an example of using `cltvols` to back up a file to tape, obtain information about the volume used, and unallocate the device. In this example the operator has mounted a scratch tape with an external label, VOL001. The commands are:

1. `cltalloc -P 3494 -f my3494`
2. `tar -cvf my3494 large_backup_file`
3. `cltvols -f my3494`

The output of the command, displayed on your screen, is:

```
Total Number of volumes: 1
File Sequence Number: 1
External Volume IDs: VOL001
Internal Volume IDs: -----
```

4. `cltunall -f my3494`

6.2.5 Using the Volume Information File

To have information, such as the external label, written to a file instead of your screen, specify the `VFILE` option on the `cltalloc` command.

The `VFILE` contains a subset of the information usually found in a full-function tape catalog.

The `VFILE` parameter is very important for maintaining all of the information about tape usage. The `cltclose` command closes the `VFILE` that was created when the `VFILE` option was specified on the `cltalloc` and `cltopen` commands. When you execute the `cltclose` command, the `VFILE` is written

in the `$HOME/.nettape` directory; it is named using a file name that corresponds to the `dsname` entered on the `-d dsname` parameter.

If you write many data sets or files on the tape, you would use `cltopen` for each file to indicate the name of the file. After each write operation (such as AIX `tar` or `dd`), you must use `cltclose` to create a VFILE corresponding to each file written on the tape. The VFILE is used later when you read your tape data set or file. NetTAPE reads the external volume name and internal volume name (if used) from the VFILE. By specifying the data set or file name, you will automatically know the tape that will be used or mounted.

Here is an example using VFILE when backing up two files to a tape with the VFILE option and using `cltclose` to close the VFILE file. The steps are:

1. `cltalloc -P 3494 -f my3494 -d file_01 -s 1 -o "VFILE"`
2. `tar -cvf my3494 backup_file1`
3. `cltclose -f my3494`

After the `cltclose` command, the following message is displayed on the screen:

```
$HOME/.nettape/file_01 closed.
```

Here is the layout of the VFILE saved in the `$HOME/.nettape` directory, with our explanation for each line:

```
VERS001      # NetTAPE version number
adsmres2     # Name of file owner
AINT8MM      # The device pool name
NL           # Labeltype
00000001     # Number of volumes the file spans
00000001     # The file's starting sequence number
VOL003       # The list of EXTERNAL Volume IDs
NO_INT_VOL   # NO INTERNAL Volume IDs
```

If you are using internal volume IDs, the `NO_INT_VOL` field is replaced with the list of internal volume IDs.

6.2.6 Using MultiVolume Data Sets

Typically when the end of the first tape is reached, the user application receives an end of tape (EOT) or end of volume (EOV) condition code.

The `cltalloc` parameter `-o SUP_EOV` (suppress end of volume) changes the tape behavior for multiple volumes. The application views only one unique tape, but the operator sees many tapes.

Here are some common `cltalloc` parameters:

- SUP_EOV

SUP_EOV specifies that NetTAPE automatically mounts the next volume (if necessary), when the EOV or EOT is reached. If you omit the SUP_EOV keyword, NetTAPE returns the NetTAPE_EOV (-2) return code when it reaches the EOV on a read operation (that is, when it reads one tape mark) or the NetTAPE_EOT (-4) return code when it reaches the EOT on a write operation (that is, when no more space is available). When reading a multivolume tape, NetTAPE returns the NetTAPE_EOV (-2) return code when it reaches the EOV (that is, when it reads two tape marks) on the last volume in the volume list.

- SUP_EOF

SUP_EOF (suppress end of file) allows NetTAPE to read a tape with multiple data sets. For the read operation, SUP_EOF specifies that NetTAPE should ignore the EOF tape marks and continue processing the tape. If you omit the SUP_EOF keyword, NetTAPE stops reading when it reaches the first EOF tape mark.

- cltalloc -a

NetTAPE only requests one tape for each mount message. When you are using multiple volumes, it is interesting to send a list of tape volume IDs to the operator to mount on an ACL device. You can send this list by using the -a parameter on the cltalloc command. For more information about using the -a parameter, see section 6.2.9, "Using ACL or ACF 3480, 3490, and 3590 Tape Libraries" on page 150.

Here is an example using multiple volumes. Remember, to write to the tape you must specify the SUP_EOV allocation parameter. To read from the tape, you must specify the entire volume list, as well as the SUP_EOV and SUP_EOF parameters. The steps are:

1. Allocate a 3490 device, using a scratch tape and the SUP_EOV option.

```
cltalloc -P 3490B -f mytape -o "SUP_EOV"
```

2. Write a large directory to tape that will span multiple volumes.

```
tar cvf mytape /large_directory
```

3. Find out the volume serial numbers for the scratch volumes because the cltalloc command did not request any specific volumes.

```
cltvols -f mytape
```

4. Open the device.

Open the device with the volume list provided from the cltvols command and the SUP_EOV option.

```
cltopen -f mytape -v VOL1,VOL2,VOL3 -o "SUP_EOV"
```

5. Restore the large directory.

The restore asks the operator to remount the volumes.

```
tar -xvf mytape
```

6. Unallocate the device.

```
cltunall -f mytape
```

6.2.7 Using Tape Labels on the 3480, 3490, and 3590

NetTAPE supports two types of tape labels:

- The internal label written on the tape.

NetTAPE supports two types of internal labels:

- ANSI X3.27-1987 (ISO 1001-1986) label
- IBM EBCDIC standard label

- The external label on the tape sticker. This label is manually written by the operator.

The labels are specified when using the `cltalloc` and `cltopen` commands.

Use the following parameters on the commands:

- `-l SL`, for standard labels
- `-l AL`, for ANSI labels

The `-V vol1 [,vol2,...,voln]` parameter specifies the internal serial number for the volume in the list.

When you specify `-l SL` or `-l AL`, nothing happens at mount time. However, on the first write on the tape, NetTAPE writes the label before continuing. The internal serial number for the volume is written in this label.

The `volser` name in the `-v volser [,volser2,...,volsern]` parameter is not written on the tape, but generally the operator specifies the same external and internal label.

A `volser` name is one to six characters and is converted to upper case. The `-v` specifies the `volser` external serial number for one or more tapes. If you specify `-V` with no values, the default values are the corresponding external `volser`s. If you have to specify the internal volume with `-V`, you must also specify the external `volser` with `-v`.

The devices supported are the 3480, 3490, and 3590. In standard NetTAPE, tape labels for 4 mm, 8 mm, 1/4-inch, and 1/2-inch tape drives are not

supported. NetTAPE can support these devices if you specify (in the `nettape.adm.config` file) the following device types:

- `mm8` for the 8mm device type
- `mm4` for the 4mm device type
- `CIQ` for the QIC device type
- `TSO` for the 0ST device type
- `KRT9` for the 9TRK device type

Here is an example using tape labels. The steps are:

1. Allocate a 3494 tape device in a specific pool with a file named `dataset1`.

```
cltalloc -P 3494 -f mytape -d dataset1 -s 1 -l SL
```

2. Write the first file to the tape.

Before writing the first file, NetTAPE writes the label. We use `mytape.1` to avoid a tape rewind.

```
tar -cvf mytape.1 file1
```

3. Open the tape with a second file named `dataset2`.

```
cltopen -f mytape -d dataset2 -s 2
```

4. Write `file2` to the tape.

```
tar -cvf mytape file2
```

5. Read the two files from the tape.

Open the tape to read the first file named `dataset1`.

```
clltopen -f mytape -d dataset1 s 1
```

6. Restore the first file from the tape.

```
tar -xvf mytape
```

7. Open the tape with the second file named `dataset2`.

```
cltopen -f mytape -d dataset2 s 2
```

8. Restore the second file from the tape.

```
tar -xvf mytape
```

9. Unallocate the tape device.

```
cltunall -f mytape
```

Please note that if you use a labeled tape for `tar`, the tape may not be readable in a UNIX “no-label” environment.

6.2.8 Setting an Application Block Size

With NetTAPE, the application buffer size determines the actual block size written to the tape. NetTAPE sets the device driver block size to zero (zero means variable block size). Typically with NetTAPE, you do not have to change the block size of the pseudo-device driver. But you can specify the maximum block size that NetTAPE must support when it reads from a tape or writes to a tape. The maximum block size value must be greater than or equal to the largest data block read or written by your application.

To specify the maximum block size use the -o "MAXDATA" parameter on the `cltalloc` or `cltopen` command. The default is 65536 bytes.

Here is an example using the AIX backup and restore commands. If you think you will need multiple volumes, you must specify the SUP_EOV option on the allocations. The steps are:

1. Allocate a 3490 device using scratch tape.

```
cltalloc -P 3490B -f mytape
```

2. Back up everything in the current directory to tape.

```
find . -print 0 backup -vif mytape
```

3. Get the volume serial.

```
cltvols -f mytape
```

4. Unallocate the device.

```
cltunall -f mytape
```

5. Allocate the device for restore, using the volume from the backup.

```
cltalloc -P 3490C -f mytape -v VOL1
```

6. Restore the tape just created.

When backing up large amounts of data, specify the block size on the restore. In our example, we restore the data, specifying a block size of 51200. See the restore online help page for details.

```
dd if=mytape bs=51200 0 restore -xvgf -
```

7. Unallocate the device.

```
cltunall -f mytape
```

Here is an example using the AIX `cpio` command and specifying a block size.

We specify a block size because we are using `cpio` to back up large files. If you think you will need multiple volumes, you must also specify the

SUP_EOV option on the allocations. See section 6.2.6, “Using MultiVolume Data Sets” on page 145 for details. The steps are:

1. Allocate a 3490 device using scratch tape.

```
cltalloc -P 3490B -f mytape
```

2. Cpio the current directory.

The block size is specified with C32, which results in a block size of 16 KB.

```
find . -print 0 cpio -ovdC32 > mytape
```

3. Get the volume serial.

```
cltvols -f mytape
```

4. Open the device specifying the volume serial.

```
cltopen -f mytape -v VOL1
```

5. Restore the tape using block size 32.

```
cpio -idmvC32 < mytape
```

6. Unallocate the device.

```
cltunall -f mytape
```

6.2.9 Using ACL or ACF 3480, 3490, and 3590 Tape Libraries

The `cltalloc -a` parameter enables NetTAPE to send a list of tape volume IDs to the operator to mount on the ACL or ACF device. The ACL feature enables sequential access to tapes. ACL is supported only by 3480, 3490, and 3590 IBM tape devices. For 3590 tape devices you can also select the ACF feature, which allows random access mode, that is, it is a mini-tape library.

The ACF feature is supported for labeled tapes only. You can specify the `-a` parameter only for devices defined in the `nettape.adm.config` file with the `ibmac1` or `ibmacf` operation modes.

NetTAPE does not support the 3590 query inventory through the command line. You can get this information from the GUI by double-clicking on the 3590 device entry. This is the way you can request a specific 3590 volume ID.

6.2.10 Writing Special Applications Using the NetTAPE API

The `/usr/lpp/nettape/samples/api` file contains sample programs that use the NetTAPE APIs. You can use the NetTAPE API to create your applications.

Practical examples include:

- Creating labels
- Sending messages to an operator to mount a 7331 tape magazine
- Copying tape to tape
- Using the NetTAPE API directly from end-user applications

Chapter 7. ADSM Tape Management with NetTAPE

In this chapter we discuss how to integrate ADSM with NetTAPE by using the new External Library facility in ADSM. We explain how the External Library works, its advantages, and what you must do to make it work.

The topics we cover are:

- The ADSM External Library and NetTAPE ADSM External Manager
- Configuring the ADSM External Library for use with the NetTAPE ADSM External Manager
- Managing an ADSM NetTAPE environment

7.1 The ADSM External Library and NetTAPE ADSM External Manager

The ADSM External Library in ADSM V2 for AIX became available at the end of March 1996 in the V2R1M7 level. It is described in the latest *ADSTAR Distributed Storage Manager: Administrator's Guide*, SH35-0134-01. Appendix A of that guide documents the External Manager interface.

An ADSM External Library enables ADSM to decouple ADSM operations from the details of the tape drive and tape library configuration. Thus, instead of ADSM controlling the tape drives and libraries directly, as usual, ADSM sends volume mounts, volume dismounts, and volume releases (returns to scratch) to an external process, the *External Manager*, which handles the requests on behalf of the ADSM server.

External Libraries can be used with either automated or manual libraries. You can have several External Libraries, for example, for different types of tape devices defined for an ADSM server. You can have some libraries defined as external libraries and others managed directly by ADSM.

7.1.1 How Does an ADSM External Library Work?

For illustration purposes, we look at what happens when ADSM needs a tape volume and sends a mount request to the External Manager.

The mount request specifies, among other things:

- The volume name (or scratch)
- The access mode (readonly or readwrite)
- The device type

Notice that ADSM does not specify a drive address.

In this mount request example, the External Manager is responsible for:

- Verifying that the request is coming from the ADSM server
- Sending a mount request to either a human operator or an automated library
- Handling eventual mount problems, and if possible, recovering from them
- Verifying that the correct volume is mounted (ADSM volumes must have standard labels), but there is no way for ADSM to force the External Manager to do so.
- Passing back to ADSM, after a successful mount, the device address where the tape volume has been mounted

ADSM then opens the device, checks the tape label, and proceeds with writing or reading the tape volume.

With the External Library facility, the only thing that the ADSM server has to know about the tape device is its type. It also knows how many drives of this type it can utilize at a time, as defined by the `mountlimit` parameter in the device class definition.

7.1.2 The NetTAPE External Manager

NetTAPE provides an External Manager, `cladsmd`, that handles tape and library operations for ADSM External Libraries. The NetTAPE External Manager enables you to have your ADSM tape operations managed by NetTAPE and share tape drives between ADSM and other tape applications that use NetTAPE.

However, ADSM does not support remote tape access, even when going through NetTAPE.

7.1.3 Advantages of External Libraries

By removing direct device control from the ADSM server, the External Library gives you more flexibility for your ADSM operations.

7.1.3.1 ADSM Drive Sharing

The External Library enables you to share drives between ADSM and other applications.

ADSM utilizes drives only when they are needed, which in normal production environments is for only a few hours each day. With the External Library, the drives are now available to your other tape applications when ADSM does not use them.

Tape drive availability is important for applications running on the same system as the ADSM server but also provides advantages in an environment where your tape drives are attached to other hosts in addition to the ADSM server machine (that is, multitail attachments).

By not dedicating drives to an ADSM server, the External Library enables you to treat all of your (locally attached) tape drives as one resource. The External Manager selects (in the corresponding device pool) any drive that happens to be free at the time ADSM is requesting a volume. This process is much more efficient than a static partitioning of your drives between the ADSM server and other hosts.

The other applications running on the same or on different hosts can be other ADSM servers, also using External Libraries.

7.1.3.2 Additional Device Support

The External Library enables you to use devices that are not formally supported by ADSM, provided that they appear as devices that ADSM can handle. The STK drives and libraries are an example. These devices are supported by NetTAPE and look like IBM 3480 and 3490 devices. Thus, with the NetTAPE External Manager, it should be possible for ADSM to access these 3480- or 3490-like devices, provided that:

- They are supported by NetTAPE
- They are supported by the driver ADSM uses to access 3480 and 3490 devices (depending on whether they are attached by SCSI, parallel channel, or ESCON).

Caution

When using the ADSM External Library and NetTAPE External Manager, we did not test ADSM support for STK devices.

For a complete list of STK devices supported by NetTAPE, see Appendix D, “NetTape Supported Tape Drives and Libraries” on page 235. Please note that the list of devices is not identical to those supported by the ADSM for AIX StorageTek Tape Library Support Services Solution, as shown in Appendix E, “ADSM for AIX StorageTek Tape Library Support Services Solution” on page 237.

7.1.4 Sample Configurations

Figure 25 on page 156 and Figure 26 on page 157 are examples of how you can use ADSM with NetTAPE to achieve drive sharing between several

ADSM servers running on two hosts and other AIX applications. The hosts are accessing multitail drives, in this example 3590s in a 3494 ATLDS.

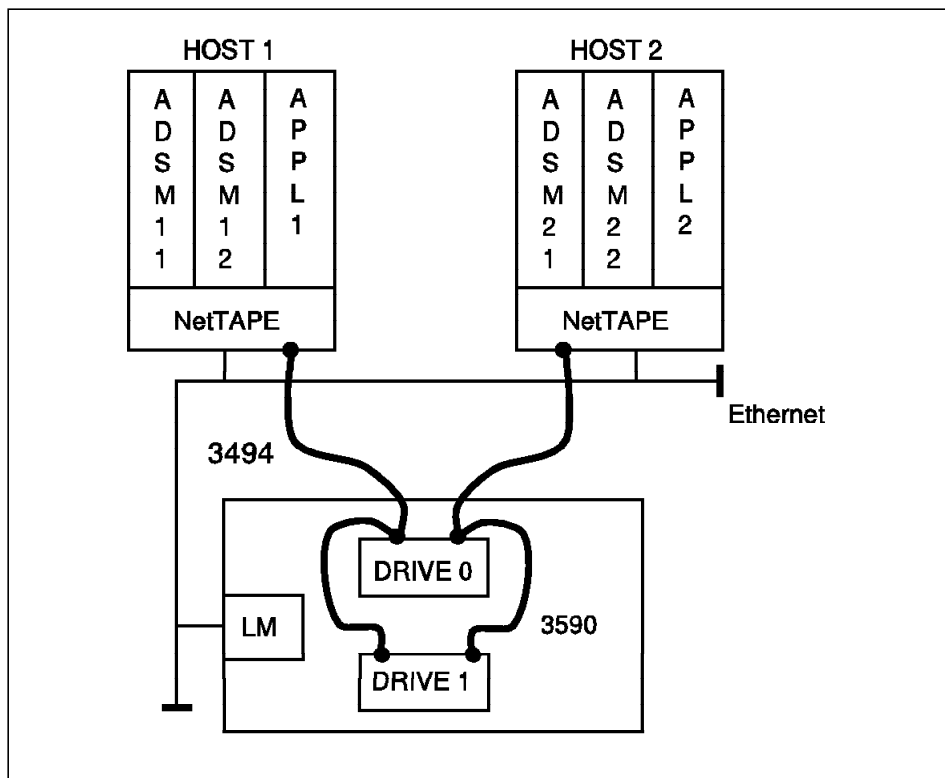


Figure 25. Drive Sharing with ADSM and NetTape: Two Hosts

In Figure 25 we have two ADSM servers on each host, as well as other AIX tape applications (APPL1 and APPL2). We have a 3494 tape library, with two 3590-B1A drives attached in twin-tail mode to both hosts through the two SCSI ports provided on 3590s. The tape library and drives are managed by NetTape, and the ADSM servers are using External Libraries through the NetTape ADSM External Manager.

With this setup, each ADSM server and AIX application can access the two tape drives at a given time, if they are free. Or, two ADSM servers and applications, on any of the two hosts, can each access one tape drive simultaneously. The two drives are dynamically allocated by NetTape to anyone who asks for them.

With this type of configuration, you will probably try to schedule the ADSM tape operations for each server so that they do not try to run at the same time.

Such a setup can be used with any tape drive that provides multitail capability, whether the tape drives are in a library or used as manual drives. Also, the hosts could be, for example, SP2 nodes (as shown in Figure 26 on page 157).

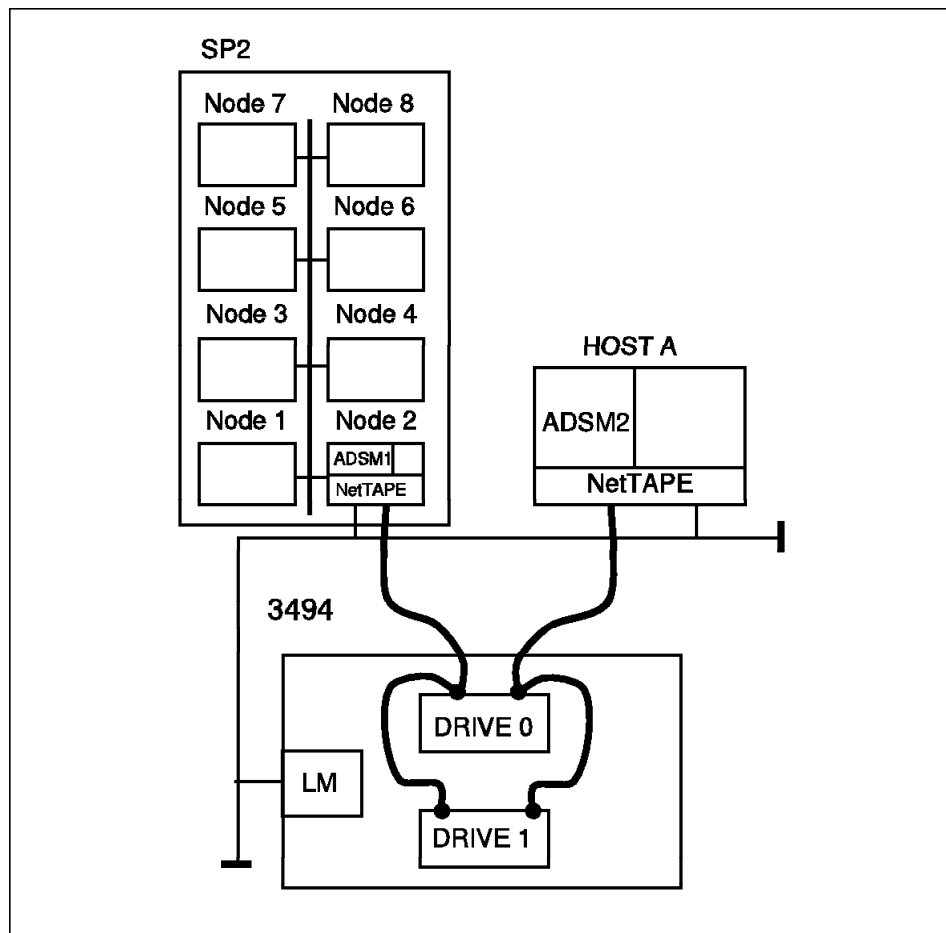


Figure 26. Drive Sharing with ADSM and NetTAPE: with an SP2

In this configuration, we have an eight-node SP2 system, where Node2 has a SCSI connection to two 3590 drives in a 3494; these drives, in twin-tailed mode, are attached to an external RISC System/6000 (HOST A). We have two ADSM servers, ADSM1 on Node2 and ADSM2 on HOST A.

With this configuration, you have drive sharing between the two ADSM servers and other applications running on these hosts, and remote tape access from all of the SP2 nodes through the SP2 High Speed Switch.

7.2 Configuring the ADSM External Library for Use with the NetTAPE ADSM External Manager

To configure an ADSM External Library for use the NetTAPE ADSM External Manager, you must:

- Run your ADSM server under a user ID that is not root
- Define the ADSM server user ID as a NetTAPE operator
- Define the External Library and External Manager in ADSM
- If you have an existing ADSM server in production that controls tape drives or libraries directly, you must change some of the ADSM definitions to make it use External Libraries instead.
- Put in place operational procedures to manage your volumes and libraries outside ADSM. We cover the operational procedures in 7.3.8, “Managing Library Volumes for ADSM” on page 172.

7.2.1 Running the ADSM Server Under Its Own User ID

There are many reasons for running the ADSM server under a user ID other than root. The main ones are:

- NetTAPE requires that you have a `netrc` file, which includes the password, for every NetTAPE user. If ADSM is running under root, the root password must be there, which is not very good for security.
- The NetTAPE operator interface shows the requesting user ID. It is much better to have the ADSM mount requests appearing under an *adsm* user ID than under root because a root user ID would not tell the operators what the mount is for.
- The External Manager has to check that ADSM mount requests are coming from the ADSM server. The only way to do this is for the ADSM server to have a unique user ID, rather than root.

If you are running your ADSM server under root, you must perform the following to run it under, in this example, user ID *adsm* in group *staff*:

1. Log in as root.
2. Create the *adsm* user ID in group *staff*, using SMIT, and give it an initial password.
3. Create a directory in which you will store the configuration files and scripts for the ADSM server, for example, `/home/adsm/adsmserve`.
4. Stop the ADSM server.
5. Copy the `dsmserv.opt` file into the `/home/adsm/adsmserve` directory. The `dsmserv.opt` file is usually in the `/usr/lpp/adsmserve/bin` directory.

6. Copy the `dsmserv.dsk` file from `/usr/lpp/admserv/bin` to the `/home/adsm/admserv` directory.

7. Change ownership of the directories and files that contain the ADSM database, log, and storage pool volumes to the new user ID. You change ownership under root by issuing the AIX `chown` command. For example, if your ADSM logical volumes are mounted under the `/adsm` directory, you would enter:

```
chown -R adsm:staff /adsm
```

The `-R` flag stands for recursive, which changes the ownership of all of the files and directories under `/adsm`.

8. Change the access rights to the `dsmreg.lic` file in `/usr/lpp/admserv/bin`. The `dsmreg.lic` file manages ADSM license keys. As distributed with ADSM, it is readable only by root, so you will want to change it so that the `adsm` user ID can read it. Use this command:

```
chmod a+r dsmreg.lic
```

9. Copy the `admserv.licenses` file from the `/usr/lpp/admserv/bin` directory into, in our example, the `/home/adsm/admserv` directory.

The `admserv.licenses` file contains the license keys that the ADSM administrator entered when installing ADSM.

10. Create a script that defines the appropriate environment variables and starts the server. For example:

```
#!/bin/ksh
# script name: start_dsmserv script
#Start the ADSM Server from a user ID other than "root"
#
# check that this script is running under the "adsm" user ID
#
me=$(whoami)
if [ $me != "adsm" ]
then
    echo "Server must be started under user ID adsm, exiting"
    exit
fi
# environment variable for ADSM Server code
export DSMSERV_DIR=/usr/lpp/admserv/bin
# environment variable for dsmserv.opt
export DSMSERV_CONFIG=/home/adsm/admserv/dsmserv.opt
cd /home/adsm/admserv
nohup /usr/lpp/admserv/bin/dsmserv quiet >> server.console 2>&1 &
```

11. Log in as `adsm` and start the server by issuing:

```
/home/adsm/admserv/start_dsmserv &
```

12. Update the `rc.adsmserve` script because it is started at system boot time, or update `/etc/inittab` to point to another script other than `rc.adsmserve` that will do the equivalent to this script, but from root.

7.2.2 Defining the ADSM Server to NetTAPE

To use ADSM with the NetTAPE External Manager, you must define the `adsm` user ID as a NetTAPE operator so that NetTAPE can control who can scratch a private volume. The ADSM server can issue release commands; therefore, defining the `adsm` user ID as a NetTAPE operator allows NetTAPE to control the operation.

After you have defined the ADSM server user ID, `adsm`, in AIX, you must add the `adsm` user ID to your list of defined NetTAPE operators in the `nettape.adm.config` file, for example:

```
stdsmn.operators    operator root adsm
```

Please refer to Chapter 5, “NetTAPE Installation and Configuration” on page 105 for details on NetTAPE configuration files.

7.2.3 Defining the External Library and External Manager

If you are installing a new ADSM server and decide to use the NetTAPE External Library and External Manager, you would enter the following ADSM definitions under an ADSM administrator session:

1. Define a library with a library type of *external*.

For example, if your tape devices are 8 mm tapes, and you decide to call your library `int8mm`, the ADSM command is:

```
define library int8mm libtype=external
        externalmanager=/usr/lpp/nettape/bin/cladsm
```

Note: The library name must be the name of the NetTAPE device pool that you want to use for the ADSM External Library.

2. Define a device class pointing to this library.

```
define devclass net8mm devtype=8mm library=int8mm mountretention=1
mountlimit=2
```

As you are going to share drives between ADSM and other applications, you probably do not want to use the ADSM default for `mountretention`, which is 60 minutes. In this example, we have set it to 1 minute.

We have also defined the `mountlimit` as 2, which means that ADSM will be able to utilize two tape devices at a time in this device class. Two is the minimum `mountlimit` you should define if you want ADSM to perform tape reclamation for the storage pool (assuming that you have at least two drives).

3. Define a storage pool that will use (tape) devices through NetTAPE.

```
define stgpool backup8mm net8mm
```

The backup8mm storage pool points to the net8mm device class, which points to the int8mm library, which is an External Library that is using the NetTAPE cladsmd External Manager.

4. Define the volumes in the storage pool that will be used through the External Library.

```
define volume BACKUPEXT VOL001
```

Note that you do not define any drive in ADSM when you use an External Library.

IMPORTANT: The NetTAPE device pool that you will use for ADSM must go through the ADSM tape driver, that is, the drives must be defined as /dev/mtx in the nettape.adm.config file, rather than as /dev/rmtx.

7.2.4 Migrating an Existing ADSM Server

If you already have an ADSM server in production, you have data on tape cartridges, and you have ADSM definitions for your drives, libraries, and device classes. In this section we explain the ADSM definitions you will have to add or update to use an External Library.

Let us first look at the logical structure of the relevant ADSM elements and how they are linked:

ADSM Element Link

Volumes	(Tape) volumes are defined in a storage pool.
Storage pool	A tape (or sequential) storage pool points to a device class.
Device class	A device class points to a library.
Library	If the library is an automated library (not a manual library), library volumes are associated with it.

Note that, for automated libraries, tape volumes are registered twice in ADSM:

- As volumes in a storage pool
- As library volumes (LIBVOL) in an automated library

ADSM needs the LIBVOL part of the volume information because it manages the automated library directly: It has to know which private and scratch categories to use, it has to maintain an inventory of its library volumes, and it must be able to eject (check out) volumes.

When you want to convert from a normal, ADSM-managed library, either manual or automatic, to an External Library, there are several conversion steps.

We use a real, albeit very simple, case as an illustration. As a starting point, here are a few queries and their output on our ADSM server, which uses one 3495 ATLDS:

```
q vol
```

Volume Name	Storage Pool Name	Device Class Name	Estimated Capacity (MB)	%Util	Volume Status
ABC100	BACKUPCART	ES3490	360.0	19.3	Filling
ABC104	BACKUPCART	ES3490	360.0	0.5	Filling

```
adsm>
q libvol
```

Library Name	Volume Name	Status	Last Use
LIBMGRC7	ABC100	Private	Data
LIBMGRC7	ABC104	Private	Data

The server has two tape volumes (ABC100 and ABC104), which show up as both storage pool volumes and library volumes in the LIBMGRC7 library.

```
adsm>
q stg backupcart
```

Storage Pool Name	Device Class Name	Estimated Capacity (MB)	%Util	%Migr	High Mig%	Low Mig%	Next Storage Pool
BACKUPCART	ES3490	720.0	9.9	100.0	90	70	

The server has one storage pool on tape, BACKUPCART, which points to device class ES3490.

```

adsm>
q devc1 es3490 f=d

      Device Class Name: ES3490
      Device Access Strategy: Sequential
      Storage Pool Count: 1
      Device Type: CARTRIDGE
      Format: DRIVE
      Est/Max Capacity (MB): 0.0
      Mount Limit: 2
      Mount Wait (min): 60
      Mount Retention (min): 1
      Label Prefix: ADSM
      Library: LIBMGRC7
      Directory:
      Last Update by (administrator): SERVER_CONSOLE
      Last Update Date/Time: 1996.04.02 14:31:43

```

The ES3490 device class points to our LIBMGRC7 library.

```

adsm>
q libr libmgrc7

Library Name  Library   Device          Private  Scratch  External
              Type                Category  Category Manager
-----
LIBMGRC7     349X     /dev/lmcp0      300     301

```

The LIBMGRC7 library controls the 3495 by using the /dev/lmcp0 special device and has the ADSM default categories defined for scratch and private volumes.

We now enter the ADSM commands to convert the server so that it does not manage the automated library directly but instead uses the NetTAPE External Library.

Figure 27 on page 164 is a schematic overview of the status before and after we migrated the server.

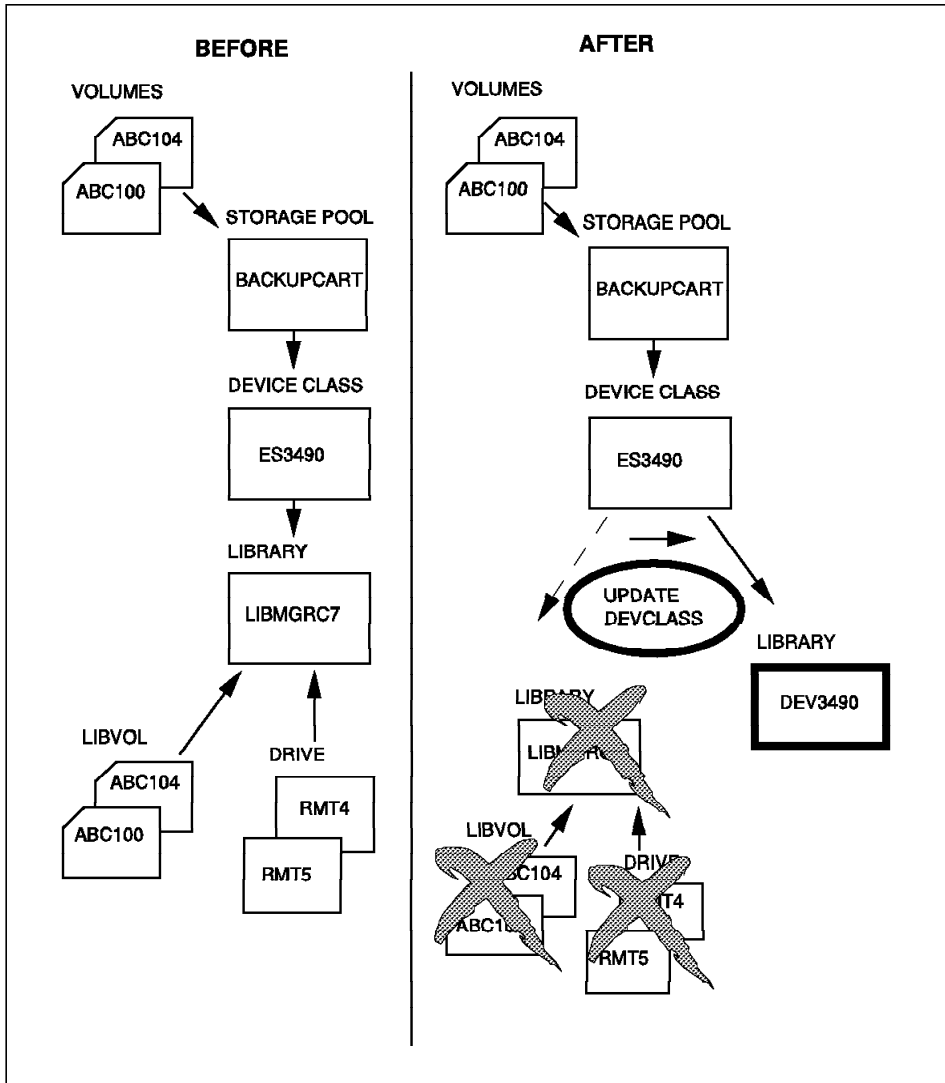


Figure 27. Migrating an Existing ADSM Server to Use an External Library

The migration steps are:

1. Define a new library with a library type of *external*.

```

adsm>
def libr DEV3490 libtype=external externalmanager=/usr/lpp/nettape/bin/cladsm
ANR8400I Library DEV3490 defined.

```

Important: The library name must match the NetTAPE device pool you are going to use for the External Library.

2. Update the existing device class to point to the new library.

```
adsm>
update devclass es3490 libr=dev3490
ANR2205I Device class ES3490 updated.
```

3. Delete all of the ADSM drives that were defined in the automated library.

```
adsm>
delete drive libmgrc7 rmt4
ANR8412I Drive RMT4 deleted from library LIBMGRC7.
adsm>
delete drive libmgrc7 rmt5
ANR8412I Drive RMT5 deleted from library LIBMGRC7.
```

4. Delete the library itself.

Note that when you delete the library you also remove all of the LIBVOL definitions for the library volumes. You do not have to perform checkout operations on the library volumes.

```
adsm>
delete library libmgrc7
ANR8410I Library LIBMGRC7 deleted.
adsm>
```

That is it!

All of your existing volumes still point to the same storage pool and device class, and the device class has been updated to point to the new External Library, DEV3490.

```
adsm>
q devcl es3490 f=d

      Device Class Name: ES3490
      Device Access Strategy: Sequential
      Storage Pool Count: 1
      Device Type: CARTRIDGE
      Format: DRIVE
      Est/Max Capacity (MB): 0.0
      Mount Limit: 2
      Mount Wait (min): 60
      Mount Retention (min): 1
      Label Prefix: ADSM
      Library: DEV3490
      Directory:
      Last Update by (administrator): SERVER_CONSOLE
      Last Update Date/Time: 1996.04.02 14:53:36
```

Note: The only automated libraries that NetTAPE supports are the 3494, 3495, 3590-B11, and STK libraries. NetTAPE does not support other libraries, most notably SCSI libraries such as the 7331.

7.3 Managing an ADSM NetTAPE Environment

In this section we describe some of the operational aspects that change when you use External Libraries. The changes affect only the system and ADSM administrator; ADSM end users do not see any changes.

7.3.1 Which Do You Start First—NetTAPE or ADSM?

When you start the ADSM server, ADSM tries to initialize its libraries, including the External Libraries. If NetTAPE has not been started, you will see error messages on the ADSM server console, because the `cladsmd` External Manager cannot contact NetTAPE.

As soon as NetTAPE is started, ADSM mount and volume release requests will work because they are preceded by another initialize request, and the NetTAPE interface will come up.

NetTAPE should be started before the ADSM server. If it is not, ADSM mount requests will fail, but will be handled by ADSM (for example, ADSM retries a storage migration every minute) until NetTAPE comes up.

7.3.2 Labeling ADSM Tapes with NetTAPE

If you are managing all of your tape operations with NetTAPE, you will want to use NetTAPE to label your cartridges for ADSM.

When you run ADSM with NetTAPE, you can label tape cartridges in two ways:

- Use the ADSM DSMLABEL utility
- Use the NetTAPE labeling capabilities

7.3.2.1 Using DSMLABEL

The DSMLABEL method involves two steps:

1. Issue the NetTAPE allocation command, `cltalloc`, to get the volume mounted.
2. Use DSMLABEL to put an ADSM label on the volume.

With the level of DSMLABEL we used, the labeling did not work because of an incompatibility between DSMLABEL and the NetTAPE pseudo-driver.

To use DSMLABEL with NetTAPE, you must mark the tape device as *reserved* in NetTAPE, for example, using the operator GUI, and then use the device special name, `/dev/rmtx`, on the DSMLABEL command.

To un-reserve the device in NetTAPE, you change the device state to *up*.

7.3.2.2 Using NetTAPE

NetTAPE can write standard labels on tape volumes (see 4.4.9, “Tape Labels” on page 99), so you can take advantage of this facility to label volumes that ADSM will use.

If you specify to NetTAPE that you want to work with standard labeled tapes, it writes the label for you dynamically when you write to the tape. The steps are:

1. Allocate a tape volume specifying standard label.

```
cltalloc -P INT8MM -v adsm06 -V adsm06 -l SL -d adsm06
```

The command asks NetTAPE to allocate a drive in the INT8MM device pool (which must be a pool going through the ADSM tape driver) and mount the volume with external label `adsm06`, which has an internal standard label, `adsm06`, and a data set name, `adsm06`.

2. Write something on the tape, for example:

```
dd if=dummy of=mytape obs=80
```

The dummy file contains just a few characters that can be anything.

3. Unmount the tape.

```
cltunall
```

`adsm06` now has a standard label so you can use the tape with ADSM.

4. Define the volume in ADSM.

```
DEFINE VOLUME BACKUPEXT ADSM06
```

5. The next time ADSM mounts the volume (through the External Manager), it will find a proper standard label on it, with VOL1, HDR1, and HDR2 records, and it will be able to use the volume.

The labeling process described above can be used with either manual drives or drives in an automated library.

7.3.3 Mounts

When a mount request is issued through the NetTAPE External Manager, the tape operator sees the mount message on the NetTAPE operator GUI (if the mount is for a manual drive; for mounts to automated libraries, the mount proceeds automatically). After the mount completes, the NetTAPE External

Manager passes the mount point information (the /dev/xxx name) to ADSM, which opens the device, checks the tape label, and then uses the volume.

If the library is an automated library, the tape operator sees the device allocation on the NetTAPE GUI (Figure 28):



Figure 28. NetTAPE GUI after an ADSM Mount through the External Manager

Here is what you see on the ADSM server console and activity log:

```

ANR1000I Migration process 2 started for storage pool BACKUPPOOL.
ANR8337I CARTRIDGE volume ABC104 mounted in drive DEV3490:/dev/rmt6.
ANR1001I Migration process 2 ended for storage pool BACKUPPOOL.
ANR8336I Verifying label of CARTRIDGE volume ADSM04 in drive
DEV3490:/dev/rmt6.
ANR8468I CARTRIDGE volume ABC104 dismounted from drive DEV3490:/dev/rmt6
in library DEV3490

```

ADSM generates a drive name, LIBRARY:/dev/xxx, when issuing messages.

Note: The device must be ready when the operator clicks on the **Mount** button on the NetTAPE operator GUI. If the drive is not ready, the following ADSM message appears:

```
ANR8779E Unable to open drive /dev/mt2, error number=46.
```

Our example shows a mount request for a storage migration. Other types of mounts work in the same way. We tested mount processing through the External Library for:

- Direct backup from clients (including two clients going to two devices)
- Storage pool migration
- ADSM database backup
- Export

However, when we tried to do two mounts for the same process, for example for a move data to the same tape storage pool, the second volume was never mounted. This is a known NetTAPE problem that is fixed in the NetTAPE AIX 3.2.5 refresh and the AIX 4.1 product.

7.3.4 Deadlocks

With the present design of the External Library interface, ADSM sends one tape mount at a time. The External Manager has no way of knowing that, in fact, ADSM requires two volumes for a given process, for example, for tape reclamation or move data. This can lead to two types of deadlock situations:

- You are sharing two drives with another application, and, if that other application (for example, another ADSM server) also needs two drives, you can get into a situation where:
 - The first server has one volume mounted.
 - Meanwhile, the second server has also mounted one volume on the other drive, and now both servers are waiting for their second volume to be mounted, which will never happen because both of the drives are busy.

This deadlock situation will clear itself when the mount requests time out, as defined by the mountwait parameter in the device class definition. The only way to avoid this deadlock situation is to ensure that processes requiring two drives are not started concurrently on both servers.

- You are using 3590-B11s, and ADSM asks for two cartridges, from the same 3590-B11 magazine, to be mounted. You can get out of this deadlock when the second mount request times out.

These deadlock situations are resolved after the mountwait time, so you should not specify too large a mountwait value. What “too large” is in your situation is for you to decide.

Deadlock Fix

A deadlock situation should not occur with the NetTAPE AIX 3.2.5 refresh or NetTAPE AIX 4.1 version because the NetTAPE ADSM daemon code no longer queues the ADSM mount request.

7.3.5 ADSM Database Restore

If you manage your tape devices with NetTAPE, you would want to do all of your tape operations through NetTAPE, including restoring the ADSM database. We tried to restore an ADSM database, but with the level of ADSM available at the time of writing, a stand-alone database restore through an External Library is not possible. Here is what happens:

```
> dmserv restore db devcl=aint8mm vol=dbb001 commit=yes
ANR7800I DSMSESV generated at 10:15:14 on Mar  5 1996.
ADSTAR Distributed Storage Manager for AIX-RS/6000
Version 2, Release 1, Level 0.7/0.7
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.
ANR7801I Subsystem (master) PID is 10259.
ANR0900I Processing options file dmserv.opt.
ANR0200I Recovery log assigned capacity is 112 megabytes.
ANR0201I Database assigned capacity is 396 megabytes.
ANR8498E Drive not defined for library INT8MM in device configuration
information file.
```

In our example, the devclass aint8mm points to a library that is defined as external.

The problem is that the DEVCONF file, which is where the restore db command gets information on the device definitions, is not correctly saved by ADSM. The DEFINE LIBRARY entry in the DEVCONF file contains:

```
DEFINE LIBRARY INT8MM LIBTYPE=external
```

The EXTERNALMANAGER=/usr/lpp/nettape/bin/cladsmd entry is missing!

This problem will be fixed by a future ADSM PTF. Until then, you can correct this problem manually by adding the EXTERNALMANAGER definition to the library definition in your DEVCONF files. Adding the definition ensures that the External Manager is known to ADSM at start time, so that ADSM will be able to call the External Manager for a restore db command.

Another way of correcting this problem is to define your drives not to use the External Manager. You define your drives in a device class that is not being used but will be saved in the DEVCONF file:

```
define library dummy libtype>manual
define devclass dummy devtype=cartridge
define drive dummy drive1 device=/dev/rmt1
```

When you perform a restore db command, you would use the command:

```
dsmserv restore db devclass=dummy volumes=...
```

after changing the tape drive to not being under the control of NetTAPE by putting it in a reserved state in NetTAPE.

You put the device into a reserved state either by using the operator GUI (click on the device and then on the **Reserve** button) or by issuing the NetTAPE command:

```
cltopr -c reserve -d device
```

7.3.6 What If NetTAPE Assigns ADSM a Remote Tape Drive?

ADSM does not support remote tapes. However, if the NetTAPE device pool that ADSM uses contains drives on other hosts, and the drives on the local host are down, NetTAPE allocates a remote drive to ADSM, and it will not work! The NetTAPE External Manager “sees” that the drive is not local and therefore does not accept the mount request. This is what the ADSM server console shows:

```
ANR8324I 8MM volume ADSM04 is expected to be mounted (R/W).
sinop/cladsmd 09:58:08 (pid=31548): - GetDevName(): No Device in Pool
(INT8MM) attached to host - sinop
sinop/cladsmd 09:58:08 (pid=31548): - MountRequest(): Mount command failed...
ANR1156W Move data process terminated for volume ADSM04 -
internal server error detected.
```

The messages prefixed by sinop/cladsmd are error messages sent by NetTAPE External Manager cladsmd.

If the operator, for some reason, moves the mount request from a local to a remote drive, using the NetTAPE GUI, it will not work either, and you will see the following in the ADSM server log:

```
ANR8324I 8MM volume ADSM04 is expected to be mounted (R/W).
ANR9999D mmsect.c(628): Unexpected device type 0, 6
ANR8779E Unable to open drive /dev/mt0, error number=46.
ANR1401W Mount request denied for volume ADSM04 - mount failed.
```

If you want to avoid these error situations, use, for ADSM operations, a NetTAPE device pool that contains only tape drives directly attached to the ADSM server host.

Note

The AIX 4.1 version of NetTAPE does not assign a remote tape to an ADSM server. The NetTAPE code checks for this situation.

7.3.7 Stopping ADSM While a Volume Is Mounted

With the levels of ADSM and NetTAPE we used, if you stop the ADSM server while it has a tape volume mounted, by issuing the HALT command, the volume does not get dismounted, and the External Manager process (c1adsm) does not terminate.

To dismount the volumes, you must reset the devices from the NetTAPE operator session and then kill the c1adsm processes.

To avoid this situation, ensure that a tape is not mounted on ADSM before stopping the ADSM server (which you can easily check by issuing the query mount command).

7.3.8 Managing Library Volumes for ADSM

When you use an External Library, ADSM does not directly control tape libraries, so you cannot use the ADSM functions to move volumes in and out of automated libraries. You must move volumes by using NetTAPE functions.

7.3.8.1 Inserting Volumes into an Automated Library

Inserting volumes into an automated library is normally done in ADSM by using the `checkin libvol` command. With an External Library, ADSM does not manage the library inventory, but you, or NetTAPE, must!

If you are inserting new volumes into a 349X library and want to make the volumes available to ADSM, you must:

1. Physically insert the volumes into the library (or have the operator do it for you). The volumes are placed in the insert category, X'FF00'.
2. Change the category of the volumes to either the scratch or private category you are using for the ADSM server. (You defined the categories in the `nettape.adm.config` file.)

You can change volume categories with NetTAPE by using the `cltalloc` command. For example, to change volume ABC099 from the Insert category (FF00) to category 301, you would issue:

```
cltalloc -P dev3490 -v ABC099 -S ff00 -T 301
```

You do not have to specify the source category (-S), but it is good practice because a check will occur to verify that the volume was in the category in which you thought it was.

Note that using `cltalloc` to change a volume category causes the volume to be mounted. If you want to avoid the mount, use the `mtlib` utility instead. For example:

```
mtlib -l /dev/lmcp0 -C -t 301 -s ff00 -V ABC099
```

Both commands put volume ABC099 in target category 301, which is, in our case, the scratch category for the ADSM server.

Notes:

- You may have to be root on the host controlling the library to issue the `mtlib` command.
- The `mtlib` command does not cause a volume mount (contrary to the NetTAPE `cltalloc` command), so there is no interference with NetTAPE operations.
- You do not have to specify the source category, but it is safer to specify it: `mtlib` checks that the source category of the volume is the category specified, and it fails the operation if it is not.

3. Define the volumes to ADSM.

You must define the volumes to ADSM only if you are using private volumes. If you are using scratch volumes for ADSM, you have nothing else to do.

For private volumes, you must define the volumes with the normal ADSM define volume command.

Your new volumes are now ready for use by your ADSM server (if they have been labeled; if not, please refer to 7.3.2, “Labeling ADSM Tapes with NetTAPE” on page 166).

You may want to automate this process by writing a shell script that would issue the `mtlib` commands followed by the ADSM `define volume` commands. You would get the list of volumes from an external file, using a method similar to the shell script for labeling cartridges, which we describe in 3.3.3.2, “DSMLABEL in Automatic Mode” on page 52, and include in Appendix G, “Scripts” on page 243.

7.3.8.2 Ejecting Volumes from an Automated Library

You eject volumes from an automated library because they must be physically sent to another site or they are empty and you need the cartridge slot in the library.

When you check out a nonempty storage pool volume from an automated library, if ADSM needs the volume, it asks that it be checked into the library again. You should eject a nonempty storage volume from an automated library only if you are desperate for cartridge slots, which is most likely to happen on small SCSI libraries.

The nonempty volumes you would typically want to eject, and optionally send to an offsite location, are:

- Copy storage pool volumes
- Database backup volumes
- Export volumes

The eject operation involves two steps:

1. Ejecting the cartridges from the automated library
2. Telling ADSM that these cartridges are not available for mounting

In a typical ADSM environment, the eject operation is done by using the ADSM `checkout libvol ... remove=yes` command. However, ADSM no longer manages the physical volumes, so you must perform the ejection externally.

You can process the ejection externally by using the `mtlib` command:

```
mtlib -l /dev/lmcp0 -C -t ff10 -V ABC099
```

The command changes the category to target category `X'ff10'`, the eject category, which causes the library manager to physically put the volume in the output station of the library.

To tell ADSM that a copy storage pool volume is no longer available for mounting, you change the volume attribute to *offsite*. For example:

```
update volume ABC098 access=offsite
```

Notes:

- access=offsite is valid only for copy storage pool volumes.
- The ADSM server never spontaneously asks for database backup and export volumes, so you do not have to tell ADSM that those volumes are offsite.

New DRM Function

ADSM for AIX server PTF 9 plans to include a DRM enhancement to generate a command file that contains the `mtlib` commands for all database backup and copy storage pool volumes that have been successfully processed by DRM and marked as *unavailable* or *offsite*.

Here is a sample DRM command:

```
move drmedia * wherestate=mountable tostate=vault  
cmdfilename=/usr/lpp/admserv/nettape/eject.cmds  
cmd="mtlib -l /dev/lmcp0 -C -t ff10 -V &vol"  
format=cmd
```

This DRM command scans the ADSM database for copy storage pool and database backup volumes that contain valid data and therefore must be moved offsite.

For each volume, DRM:

- Marks the volume as offsite in the ADSM database
- Updates the volume's location field in the ADSM database to reflect the customer's offsite vault location
- For automated libraries (not external libraries), ejects (checks out) the volume from the library. With NetTAPE you are using an external library, so checkout is not supported. Instead an entry with the required `mtlib` command to change the volume's category is added to the command file.

When you are ready to remove the volumes from the library, execute the `eject.cmds` command.

7.3.8.3 Returning Volumes to Scratch

Even with an External Library, you still have the choice of using private or scratch volumes for ADSM. However, you now define your scratch and private volume categories in NetTAPE instead of ADSM. Library volume categories in NetTAPE are defined by the `stdmn.devpool.cat.scratch` and `stdmn.devpool.cat.private` statements in the `nettape.adm.config` file. You must have categories defined if you use scratch volumes in ADSM.

For example, to define the same categories as the default ADSM categories for scratch and private volumes, you would define in the `nettape.adm.config` file for the NetTAPE device pool you use with ADSM (`dev3490` in our example):

```
# Default Scratch/Private Category
stdmn.devpool.cat.scratch dev3490 301
stdmn.devpool.cat.private dev3490 300
```

When ADSM sends a mount for a scratch volume, the NetTAPE External Manager sends (to a 3495 in our tests) a mount for a volume in source category 301 and target category 300 to the library. The mount command sent to the library:

- Asks for a volume in category 301
- Changes the category of the volume to category 300

When ADSM releases the volume, because it has become empty, it sends a release request to the External Manager, which then changes the volume category back to 301 in the library.

7.3.8.4 Auditing a Library Volume Inventory

The ADSM `audit library` command is usually used after you have had serious problems, either in an automated library or the ADSM database, and you have to reconcile the two inventories.

Using External Libraries, ADSM does not keep an inventory of the automated library (it does not even know if the volumes in the External Library are used on manual drives or in an automated library), so ADSM's inventory cannot be wrong—because there is no inventory! Therefore, you no longer have to audit libraries. If your automated library is not working correctly, your IBM hardware customer engineer will fix it for you.

Chapter 8. CLIO/S Tape Management

In this chapter we describe CLIO/S support for tape management. Several books document how AIX workstations can use CLIO/S to access tape devices on MVS. Therefore in this book, although we provide an overview of all of the CLIO/S functions including MVS tape access, we discuss only AIX-to-AIX tape access in detail. Note that NetTAPE replaces CLIO/S as the product of choice for AIX-to-AIX environments.

The CLIO/S topics we cover are:

- Overview
- Benefits
- Functions
- Typical CLIO/S environments
- Supported hardware and software
- Configuring CLIO/S for AIX workstations
- Configuring and managing the tape pseudo device driver
- Administrative commands
- End-user commands
- CLIO/S positioning

8.1 Overview

CLIO/S is a set of commands and APIs that can be used for high-speed communication and to access tape devices on a network of AIX workstations and MVS mainframes. CLIO/S makes it easier to distribute work and data across a network of mainframes, workstations, and 9076 Scalable POWERparallel (SP) systems. CLIO/S also provides an API to tape drives anywhere in your network. CLIO/S can be used to:

- **Quickly move data between your MVS/ESA system and your workstation (or SP).** For example, you can store large volumes of seismic data on tape and manage it using a mainframe acting as a data server to multiple workstations. This solution retains tape management as the responsibility of a single mainframe system, while permitting seismic processing capacity to increase by distributing the work.
- **Transfer very large files.** For example, you can use applications on AIX to update customer files during the day, then use CLIO/S for fast backups to take advantage of MVS as a file server with extensive data

management capabilities. Using CLIO/S for frequent file copying can mean shorter interruptions to your ongoing applications.

- **Transfer files using familiar workstation commands.** The CLIO/S CLFTP subcommands are similar to those of TCP/IP's FTP command, so there is no need for you to learn a new interface. You can even access tape data on MVS with the CLFTP subcommands.
- **Access a tape drive on MVS or AIX from your workstation as though it were a local tape drive.** For example, you can store data on MVS controlled tape drives and access it using CLIO/S connections to the compute servers.
- **Start servers on other workstations and mainframes in your network to create a parallel processing environment.** For example, CLIO/S can be used to schedule work on several workstations running in parallel. It also provides high data transfer rates and low processor utilization, permitting very high parallel efficiency.
- **Use AIX named pipes and MVS batch pipes.** For example, you can access data on MVS (either on DASD or on tape) with an AIX named pipe. Or, an MVS program can use an MVS batch pipe to send its output to AIX, where another program using an AIX named pipe can do further processing.
- **Tape driver transparency.** CLIO/S provides a standard AIX interface to the CLIO/S tape server (TAPESERV), enabling users to access tape data on remote MVS and RISC System/6000 systems through standard UNIX commands (for example, tar, cpio, or backup).

8.2 Benefits

Let us summarize the CLIO/S benefits by grouping them into three categories: flexibility, performance, and transparency.

8.2.1 Flexibility

Flexibility benefits include:

- Communication between mainframes and workstations using the Parallel Channel Adapter (PCA), also known as BMCA, and the ESCON connection
- Communication between two RISC/System 6000 workstations on any hardware supported by TCP/IP, including Fiber Distributed Data Interface (FDDI), FCS, Ethernet, or Token Ring
- File and data serving using distributed processing

- Access to remote tape drives on MVS, RISC System/6000s, or the 9076 SP
- Interfaces available in C and Fortran
- InterMix can use CLIO/S for fast transfer.

InterMix (program number 5765-223) is a UNIX-based graphical user interface for submitting jobs and accessing data on MVS and AIX workstations.

8.2.2 Performance

Performance benefits include:

- Reduced CPU use on MVS, compared to TCP/IP data transfers
- Much faster transfer rates than TCP/IP—90% of channel speed on MVS-to-workstation transmissions, and 80% of channel speed on workstation-to-MVS transmissions
- Striping, which allows a single data transfer to be distributed across two or more channel adapters to a single MVS host

8.2.3 Transparency

Transparency benefits include:

- File transfer transparency—workstation (or SP) users can issue CLFTP subcommands to access data residing on MVS (even on tape)
- Tape driver transparency—the CLIO/S tape pseudo device driver lets workstation (or SP) users access tape data on remote MVS, RISC System/6000, or SP systems through standard UNIX commands (like tar)
- Piping transparency—the CLIO/S cross system pipe link lets you pipe data directly to and from remote MVS, RISC System/6000, or SP systems
- Application programming transparency—programs using fast transfer or tape routines work regardless of where the data resides
- Bidirectional transparency—with the exception of the tape pseudo device driver (which works only on AIX), you can use the same methods to transfer data from MVS to AIX and from AIX to MVS.

8.3 Functions

The CLIO/S functions can be divided into two categories: the tape server and fast data transfer.

8.3.1 The Tape Server

The CLIO/S tape server provides two ways of accessing any tape device on a RISC System/6000, 9076 SP, or MVS/ESA system:

- Standard UNIX commands and programming calls
- Special CLIO/S tape programming interface

The CLIO/S tape pseudo device driver is the component of the CLIO/S tape server that lets you use standard UNIX commands and programming calls to access tapes on remote systems. For example, you can use the UNIX tar command to back up files to a tape on an MVS system.

The CLIO/S tape server lets you access all standard tape operations through a special CLIO/S programming interface. The tape server has a read-ahead function and supports routing to reach a tape drive by going through multiple machines. The tape interface also cooperates with tape management systems that run on MVS. The tape interface is available in both C and Fortran.

8.3.2 Fast Data Transfer

The fast data transfer feature is included in every CLIO/S package. The tape server is an optionally orderable feature. See your IBM marketing representative for more information.

Select the interface that best meets your needs for fast data transfer. One major advantage of CLIO/S fast transfer interfaces is that they can automatically select the fastest technology available in your CLIO/S network. All of the interfaces support the following high-speed communication technologies:

- System/390 (S/390) channel to BMCA. BMCA is also known as the PCA.
- S/390 ESCON channel to a RISC System/6000 ESCON adapter

TCP/IP communications are also supported. TCP/IP enables you to create CLIO/S networks that include workstations or 9076 SPs. If your TCP/IP network is configured to use a high-speed attachment such as FDDI or FCS, CLIO/S can provide good performance over TCP/IP.

8.3.2.1 File Transfer (CLFTP)

CLFTP lets you transfer files between hosts (without programming) over high-speed channels. It also supports *routing*, the ability to reach files by going through multiple machines. Routing is especially valuable in an SP environment, as it enables you to reach files on different nodes.

Because the CLFTP subcommands are similar to those of TCP/IP's FTP command, there is no need for you to learn a new interface. You can also access tape data on MVS with the CLFTP subcommands.

8.3.2.2 Cross System Pipe Link (CLPLINK)

CLPLINK lets you and programs pipe data to and from remote locations through AIX named pipes and, optionally, MVS batch pipes.

8.3.2.3 Sockets

The CLIO/S sockets interface provides communication channels that enable unrelated processes to exchange data locally and across networks. The sockets interface runs a combination of TCP/IP and CLIO/S CMI calls and is available as C subroutines.

The CLIO/S sockets interface lets you convert an existing TCP/IP sockets program into a CLIO/S sockets program with little or no change to the application.

8.3.2.4 Communication Manager Interface (CMI)

The CMI supports high-speed communications. It provides functions to start programs on a given node, establish connections between processes, and enable fast data transfer. The CMI is a lower level interface than sockets and provides more control with more functions from which to choose. The CMI is available as C subroutines.

8.3.2.5 Distributed Processing Interface (DPI)

The DPI provides functions for distributed process control as well as high-speed communications. It allows programs on different systems to communicate synchronously or asynchronously over high-speed links. The DPI provides the most control. It is available in both C and Fortran.

8.4 Typical CLIO/S Environments

In this section we describe some typical CLIO/S environments including how you can access data with the tape server and the tape pseudo device driver.

8.4.1 Accessing Tapes with the Tape Server

Here is an overview of how to use the tape server to access MVS tapes from AIX or AIX tapes from MVS or AIX.

8.4.1.1 MVS Tapes from AIX

You can use the tape server to access tapes on MVS from one or more RISC System/6000 systems, as shown in Figure 29 on page 182. Parallel channel attachments or ESCON connections provide fast data transfer. You

can use either individual or shared connections, directly or through an intermediate AIX system.

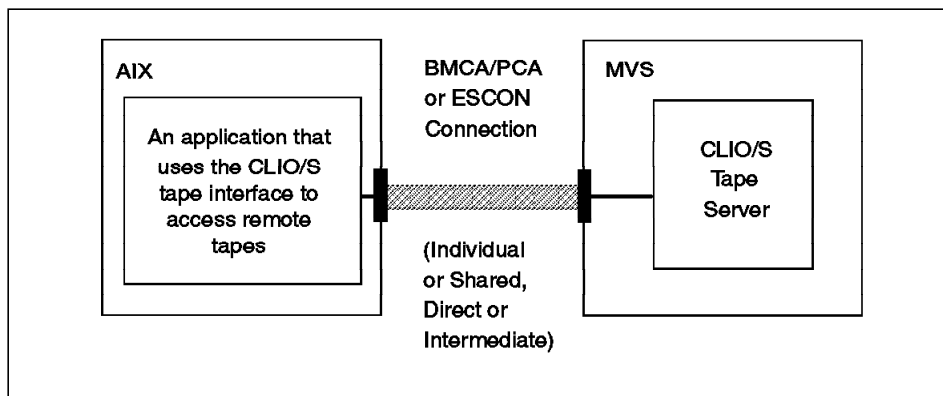


Figure 29. Accessing MVS Tapes from AIX with the Tape Server

The sample command sequence shows how easily the tape can be accessed.

Sample Command Sequence

Define tape device, open, rewind, read, close, then unallocate:

```
tpalloc(tplu1, kn1mvst, 32768, DD=TAPE93 DSN=CLIO TP.TEMP \
VOL=CLIO UNIT=CTAPE NEW NL, CLIO)
topen(tplu1, CLIO, 3)
tprew(tplu1)
tpread(tplu1, read1, 100)
tpclose(tplu1, 0)
tpunall(tplu1)
```

8.4.1.2 AIX Tapes from MVS or AIX

You can use the tape server to access RISC System/6000-attached tapes from MVS or RISC System/6000 systems, as shown in Figure 30 on page 183. Using this scheme, MVS can access tapes stored on a RISC System/6000. Parallel channel attachments and ESCON connection provide fast data transfer. For example, you can back up RISC System/6000 tapes onto MVS tapes. Accessing AIX tapes from an AIX machine requires TCP/IP.

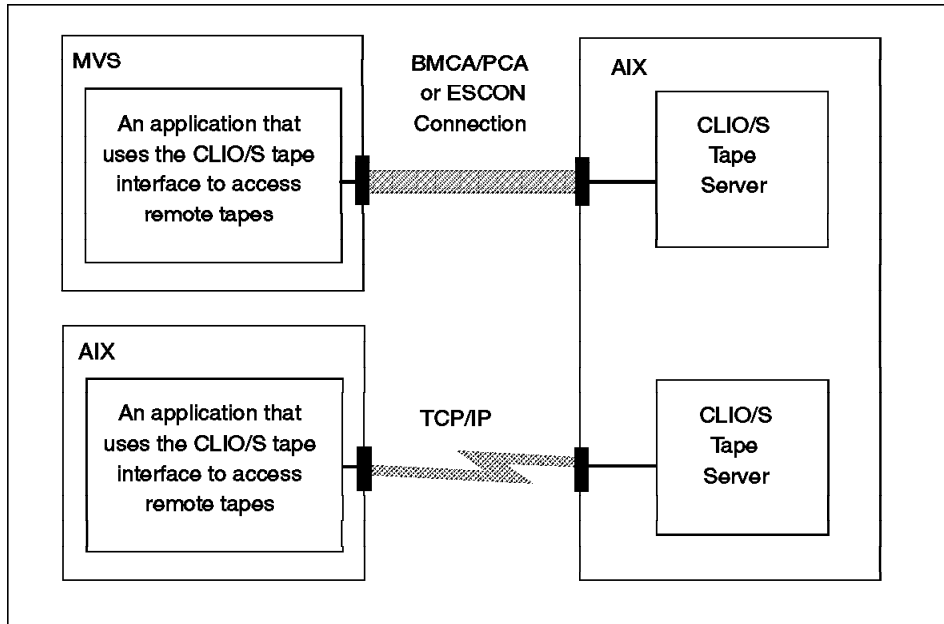


Figure 30. Accessing AIX Tapes from AIX or MVS with the Tape Server

The sample command sequence shows how easily the tape can be accessed.

Sample Command Sequence

Define tape device, open, rewind, read, close, then unallocate:

```

talloc(tplu1,zootv,150,/dev/rmt0.1,dummy)
topen(tplu1,dummy,3)
tprew(tplu1)
tpread(tplu1,mvsfile.read1(ones),50)
tpclose(tplu1,0)
tpunall(tplu1)

```

8.4.2 Accessing Tapes with the Tape Pseudo Device Driver

Here is an overview of how to use the tape pseudo device driver to access MVS or AIX tapes from AIX.

8.4.2.1 MVS Tapes from AIX

With the tape pseudo device driver, an application can transparently access tapes on MVS, with no extra programming required to account for the use of the CLIO/S tape server, as shown in Figure 31 on page 184. You can use

either individual or shared connections, directly or through an intermediate AIX system.

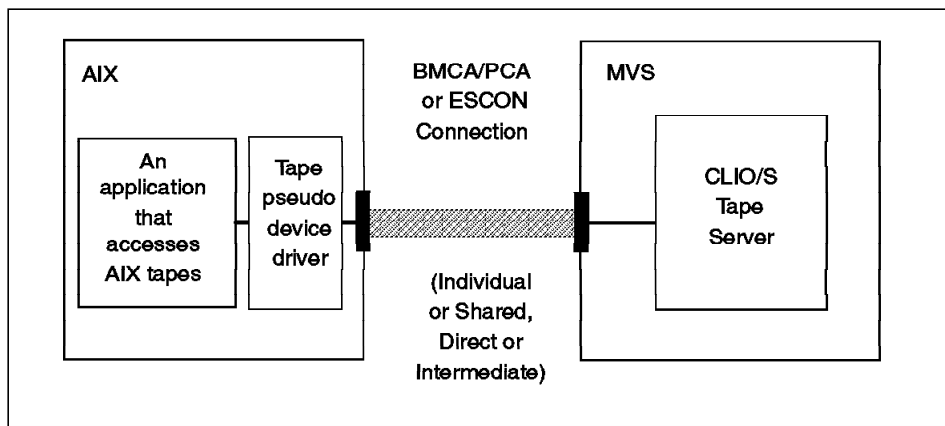


Figure 31. Accessing MVS Tapes from AIX with the Tape Pseudo Device Driver

The sample command sequence shows how easily the tape can be accessed.

Sample Command Sequence

Define tape device, then create a tar file:

```
cltalloc -v vol=12345 -f /dev/pst2
tar -cvf /dev/pst2 /bin
```

8.4.2.2 AIX Tapes from AIX

You can use the tape pseudo device driver to access AIX tapes from another AIX system, as shown in Figure 32 on page 185. Note, however, that you cannot use the tape pseudo device driver to access AIX tapes from MVS. The tape pseudo device driver works only on an AIX system.

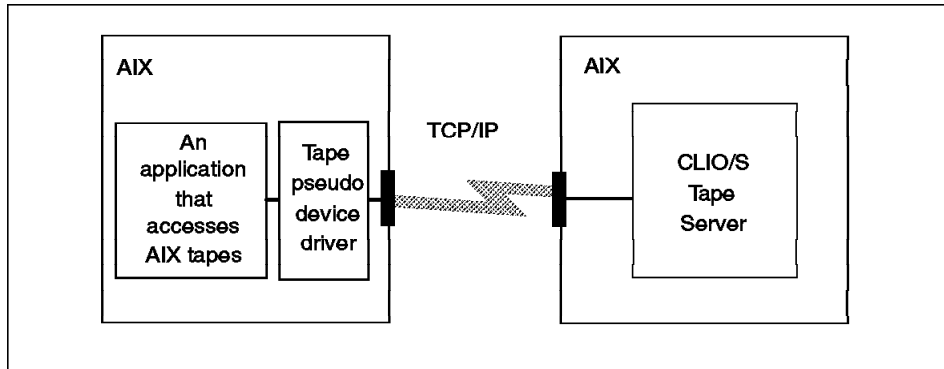


Figure 32. Accessing AIX Tapes from AIX with the Tape Pseudo Device Driver

The sample command sequence shows how easily the tape can be accessed.

Sample Command Sequence

Define tape device, then create **tar** file:

```
cltalloc -f /dev/pst0
tar -cvf /dev/pst0 /bin
```

8.5 Supported Hardware and Software

The hardware you need to use CLIO/S depends on whether you transfer data from the mainframe to a workstation or from one workstation to another workstation (including the tape server). In the sections that follow we list the hardware and software requirements for mainframe and workstation data transfer.

Note: A standard TCP/IP interface between the systems is needed for initialization and control purposes. CLIO/S can be used with TCP/IP as the only interface between systems, with performance limited to the data transfer rate of TCP/IP. If TCP/IP uses a high-speed attachment, CLIO/S can provide high-speed data transfer.

8.5.1 Mainframe to RISC System/6000 Workstation Data Transfer

CLIO/S requires hardware and software on both the mainframe and the workstation to enable mainframe-to-workstation data transfer. The requirements for each platform follow.

8.5.1.1 Mainframe Hardware

Mainframe-to-workstation data transfer requires the following mainframe hardware:

- System/390 mainframe
- *One or both* of the following:
 - Block Multiplexer Channel and Channel Cables
 - ESCON channel and fiber cable

You can use CLIO/S without a channel to channel (CTC), but performance is limited to the data transfer rate of TCP/IP.

8.5.1.2 Mainframe Software

To run CLIO/S on MVS/ESA, check the prerequisite products with your IBM representative.

8.5.1.3 RISC System/6000 Hardware

The following workstation hardware is required for workstation-to-mainframe data transfer:

Machine Type: You need either:

- RISC System/6000 machine type 7009 Model C10, 7012, 7013, or 7015, with one microchannel slot for BMCA or two microchannel slots for ESCON
- or
- IBM 9076 SP systems

Communications Devices: If you have either a BMCA or an ESCON channel adapter, the following appropriate adapters and cables are required.

The BMCA requirements for a RISC System/6000 are:

- BMCA card (feature code 2755)
- BMCA cable (feature code 2757)
- BMCA Interface Assembly (feature code 2758)

The BMCA requirements for the IBM 9076 SP systems are:

- BMCA card (feature code 2755)
- BMCA cable (feature code 2753)
- Block Multiplexer Node-to-Node Channel Adapter cable (feature code 2752) if more than one adapter card is being daisy-chained in a 9076 frame.

The ESCON Channel Adapter requirements for a RISC System/6000 or IBM 9076 SP systems are:

- ESCON Channel Adapter card (feature code 2756)

- ESCON Channel Adapter cable (see the *IBM ESCON Channel Adapter User's Guide and Service Information*, SC23-2474, for details.)

Notes:

- See the *IBM Block Multiplexer Channel Adapter User's Guide and Service Information*, SC23-2427, and the *IBM ESCON Channel Adapter User's Guide and Service Information*, SC23-2474, for information about supported configurations.
- See the *RISC System/6000 SP Site Planning Guide*, GC23-3905, for BMCA and ESCON cabling details for the 9076 SP.
- A maximum of eight 9076 SP nodes can be daisy-chained to a host processor S/390 channel. The Block Multiplexer Node-to-Node Channel Adapter cable connects a 9076 SP BMCA to a host processor S/390 channel through a BMCA cable to form a daisy-chain.
- ESCON is supported on the 9076 SP.

8.5.1.4 RISC System/6000 Software

The following software is required on the workstation for workstation-to-mainframe data transfer:

- IBM AIX Version 3.2.4 or 3.2.5 (program number 5756-030). For availability on AIX Version 4.1, see your IBM representative.

To use BMCA or ESCON, CLIO/S requires the appropriate software on the workstation:

- BMCA 3088 Emulation device driver and software (feature code 5055)
- ESCON 3088 Emulation device driver and software (feature code 5056)
- For AIX V4.1
 - For BMCA: IBM SNA Server for AIX and Channel Attach for SNA and TCP/IP, 5765-604
 - For ESCON: IBM SNA Server for AIX and Channel Attach for SNA and TCP/IP, 5765-603

8.5.2 RISC System/6000 to RISC System/6000 Data Transfer

You can use TCP/IP for workstation-to-workstation transfer. The TCP/IP connection can use any hardware supported by TCP/IP, including FDDI, FCS, Ethernet, or Token Ring.

Workstation-to-workstation data transfer requires the following software:

IBM AIX Version 3.2.4 or 3.2.5 (program number 5756-030). For availability on AIX Version 4.1, see your IBM representative.

8.6 Configuring CLIO/S for AIX Workstations

In this section we describe how to configure CLIO/S for two or more RISC/System 6000 workstations or SP nodes. The topics we cover are:

- How CLIO/S works
- Basic configuration
- Creating and customizing CLIO/S configuration files
- The CLIO/S PARSRV server
- Ensuring password security

8.6.1 How CLIO/S Works

We describe the programs that comprise CLIO/S as well as its main servers.

8.6.1.1 The View from 50,000 Feet

A typical CLIO/S command or application consists of a *master program* that you start on your local system and one or more *servant programs* that can run anywhere in your distributed CLIO/S network.

When you invoke one of CLIO/S's command line interfaces, the master program is the command, that is, the tape pseudo device driver. The command does all of the work for starting and managing servant programs and transferring data. The servant programs are included in the CLIO/S product.

The tape subroutines call code internal to CLIO/S that starts a servant program on the machine or machines to which your tape drives are physically attached. The servant program, the TAPESRV tape server, is internal to CLIO/S.

A typical example could have the tape pseudo device driver (master program) running on an AIX master application node and starting the remote tape server, TAPESERV, on a remote AIX servant node.

8.6.1.2 Under the Covers of CLIO/S

Under the covers of all of the interfaces is the CLIO/S PARSRV server. When a CLIO/S master wants to start a servant, it sends a request to the PARSRV on the remote host, using a control connection over TCP/IP. PARSRV creates the link between the hosts and then starts the servant program.

If you are using one of CLIO/S's command line interfaces (that is, the tape pseudo device driver) or the tape programming interface, subroutines internal to CLIO/S cause PARSRV to start the appropriate servant program on your behalf.

8.6.2 Basic Configuration

The basic configuration for the tape server or file transfer consists of two RISC System/6000 hosts (workstation-to-workstation). You use TCP/IP for workstation-to-workstation data transfer. The TCP/IP connection can use any hardware supported by TCP/IP, including FDDI, FCS, Ethernet, or Token Ring. Here are details about the hosts' functions:

- The CLIO/S PARSRV server, as well as the servants, have tape devices attached.
- The second host runs the master program. All users connected on the second host can access the tape connected on the CLIO/S PARSRV server by using the tape pseudo device driver.

For portability and flexibility, CLIO/S applications do not contain the specific information needed to start communications. This information is in the PARCONF and NETRC configuration files.

For details on how to install CLIO/S, refer to the *CLIO/S User's Guide and Reference V2.1*, Chapter 3, "Configuring and Starting CLIO/S Networks," and Chapter 4, "Configuring the CLIO/S Tape Pseudo Device Driver."

Figure 33 on page 190 shows the CLIO/S network we use in the configuration file examples throughout this document.

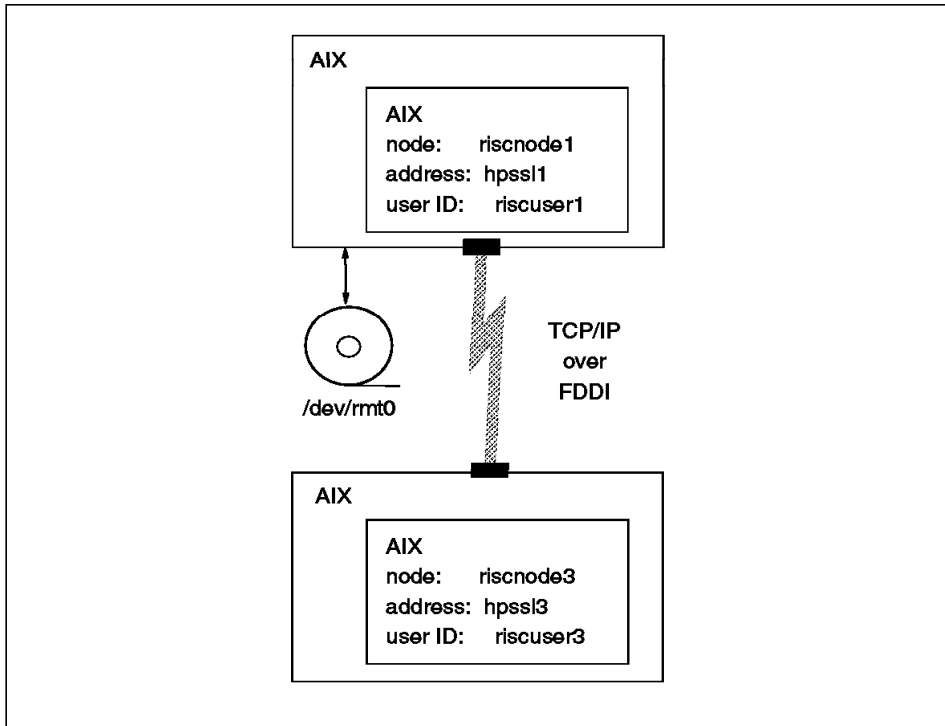


Figure 33. Sample CLIO/S Configuration

8.6.3 Creating and Customizing CLIO/S Configuration Files

CLIO/S configuration files are used so that applications using CLIO/S functions do not have to know the details of the network connections, hardware technologies, or software protocols used to support the high-speed data transfer CLIO/S provides.

CLIO/S uses the configuration files shown in Table 4:

File Name	Created by	Resides on	Purpose	Notes
PARCONF	Administrator, user	MVS and AIX	Used to establish a connection	Each user requires access
NETRC	User	MVS and AIX	Makes a connection	Also used by other non-CLIO/S applications

8.6.3.1 The PARCONF File

The PARCONF file is the first place CLIO/S looks for configuration information. PARCONF contains the information CLIO/S uses to locate nodes and to know how to connect to them. Each user ID in a CLIO/S network must have access to a PARCONF file on the host that starts the client program.

PARCONF is found in \$HOME/PARCONF or /etc/PARCONF on AIX. If both files exist, CLIO/S uses \$HOME/PARCONF.

The PARCONF file consists of a series of one-line entries, each defining a unique nickname. You must include at least one entry for each remote host you want to access. Each system's own entry is optional. You can have multiple entries for any remote host so that you can have different nicknames for different CLIO/S protocols. Hosts that are intermediate nodes for the tape server or tape pseudo device driver should have an entry defining the route. You do not need routing nicknames for intermediate nodes for CLFTP or CLPLINK. The syntax of each entry is:

```
nickname type protocol nodeaddress extra
```

Here are the PARCONF entries for each of the nodes in the configuration shown in Figure 33 on page 190:

- riscnode1: **/etc/PARCONF** or **/u/riscuser1/PARCONF**
riscnode3 1 1 hpss13
- riscnode3: **/etc/PARCONF** or **/u/riscuser3/PARCONF**
riscnode1 1 1 hpss11.kgn.ibm.com

Each PARCONF file reflects the network configuration of its node. In many cases, different user IDs on the same or different processors can share a single PARCONF file (by either copying it or using a shared file system).

8.6.3.2 The NETRC File

The NETRC file is not unique to CLIO/S but is used by other services such as rsh. CLIO/S uses NETRC for the same purpose as rsh, to provide user ID and authorization information. NETRC is found in \$HOME/.NETRC on RISC System/6000s.

Here are the NETRC files for the nodes in the configuration shown in Figure 33 on page 190:

- riscnode1: **/u/riscuser1/.NETRC**
machine hpss13.kgn.ibm.com login riscuser3 password secret3

- riscnode3: **/u/riscuser3/.NETRC**
machine hpssl1.kgn.ibm.com login riscuser1 password secret1

8.6.4 The CLIO/S PARSRV Server

Most CLIO/S applications, with the exception of the CLIO/S socket interface, use the CLIO/S PARSRV server to start jobs on remote hosts. In this section we describe the PARSRV program, the environment variables that PARSRV uses to control CLIO/S operations, how to start and stop the PARSRV server, and how you customize the CLIO/S environment by modifying the default values for the environment variables.

PARSRV is the CLIO/S server; it starts CLIO/S servant applications. PARSRV listens at a known socket waiting for applications to contact it. An administrator can start PARSRV and leave it running for others to use. Or, users can start their own PARSRV. There can be one or more PARSRV servers on a host. There must be a PARSRV server started on each host on which a user wants to start CLIO/S remote applications. You can run CLIO/S master programs on hosts in your CLIO/S network that do not have a PARSRV server running, provided that you do not want to start servants on that host.

One PARSRV server can be shared among multiple users (and applications) and by users on multiple hosts. Each CLIO/S application needs PARSRV only to establish the initial communication. After the client and server programs are communicating directly over the CTC or TCP/IP connection, PARSRV is not involved.

The steps to configure a PARSRV server are:

1. Decide which port to use.
2. Set environment variables.
3. Start the PARSRV server.

Table 5 on page 193 lists the customizable programs you use to start CLIO/S.

<i>Table 5. Programs for Starting CLIO/S</i>				
Program Name	Function	Resides on	Purpose	Notes
FCFPARSR, parsrv	The CLIO/S server	MVS and AIX	Submits CLIO/S jobs	If there is more than one parsrv, or if you are not using the default port number, each user needs to know the port number.
fcfconf.c	Some CLIO/S defaults	MVS and AIX	Default environment variables and global variables	Compile and relink CLIO/S programs after changing.

8.6.4.1 Selecting a Port Number

By default, PARSRV listens on TCP/IP port 6667 for a CLIO/S application to request a connection. Use the CLIOPSPT and CLIOPSID environment variables to override the default PARSRV port. Be sure to set the environment variables before starting the server, as described in 8.6.4.2, "Setting CLIO/S Environment Variables" on page 194. You can also run multiple PARSRVs on a single host, each with its own port number.

If your installation chooses to have a single PARSRV on each host, all applications can use the default port number defined in fcfconf.c. The CLIO/S product defaults to port 6667; you can change the port as described in 8.6.4.5, "Changing Defaults for CLIO/S Environment Variables" on page 195.

This is the algorithm that CLIO/S uses to determine the port used by PARSRV:

```

Get environment variable CLIOPSPT;
Get environment variable CLIOPSID;
if (CLIOPSPT != 0) {
    port = CLIOPSPT;
} else {
    service = getservbyname(CLIOPSID, "tcp");
    if (service == NULL) {
        port = Cliopspt; /* default=6667,
                           from clioconf.c */
    } else {
        port = service->s_port;
    }
}

```

```
}
```

8.6.4.2 Setting CLIO/S Environment Variables

When your CLIO/S servant program is started, PARSRV passes it the value of the master program's CLIO/S environment variables. The master's values override any defaults established on the server system.

The way the master process initializes environment variables depends on whether you start your master program on AIX or MVS. In the AIX environment, use the `setenv` (C shell) or `export` (Korn shell) commands to set the environment variables before you start your master application. For example, a C shell user who wants to see all CLIO/S debug messages would issue:

```
setenv CLIODEB -1
```

A Korn shell user who wants to see only messages relating to opening and closing a CTC connection would issue:

```
export CLIODEB=8192
```

See the *CLIO/S User's Guide and Reference* for more information about the environment variables.

8.6.4.3 Starting the PARSRV Server

On AIX, PARSRV is started with the `parsrv` command, which is installed in the `/usr/lpp/fsf/bin` directory. The AIX `parsrv` submits jobs in the directories specified by the `$CLIOBIN:$CLIOSYS:$CLIOPATH` path. In AIX, normal processing messages are written to `stdout`; error messages are written to `stderr`. You can redirect `stdout` and `stderr` to a file to log the activities of the PARSRV server. For example:

```
parsrv 1> parsrv.log 2>&1
```

For both AIX and MVS, if you try to start the PARSRV program on a port where another PARSRV is already running or has recently stopped, you get the following message:

```
CLIO/S parallel server failed to start on nodeid
```

```
Probable reason:
```

```
Another parallel server is either running or has been  
stopped very recently.
```

```
Please wait a few seconds for TCP/IP to release port nnnn
```

8.6.4.4 Stopping the PARSRV Server

Servant jobs already submitted by PARSRV are not affected when you stop the PARSRV server.

8.6.4.5 Changing Defaults for CLIO/S Environment Variables

The `fcfconf.c` file shipped with CLIO/S lets an installation customize the defaults for certain CLIO/S environment variables. On AIX, the file is in the `/usr/lpp/fsf/sample/` directory. The environment variables whose defaults can be specified are `CLIOBIN`, `CLIOSYS`, `CLIOPATH`, `CLIOPSID`, and `CLIOPSPT`. To change these defaults, modify the appropriate values in `fcfconf.c`, compile `fcfconf.c` with the sample makefile, and link all CLIO/S programs with the resulting object code.

8.6.5 Ensuring Password Security

CLIO/S masks passwords before sending them over the network. IBM provides source code for the password lookup, masking, and unmasking subroutines used by CLIO/S. You can change these routines to meet the security requirements of your installation. Be sure to make corresponding changes on all hosts in your CLIO/S network.

The AIX source code is in `/usr/lpp/fsf/sample/fcffindp.c`. The password functions provided are:

- **find_pw(*node*)**

This function returns a structure containing the user ID and password associated with that node name. The default is to find the information in the user's NETRC file.

- **encr_pw(*arg*)**

The argument is the structure returned by `find_pw`. The function returns a structure containing a masked version of the argument.

- **decr_pw(*arg*)**

This function converts the masked structure back to the original structure containing user ID and password.

The master program calls `find_pw` and `encr_pw`. `PARSRV` calls `decr_pw`.

8.7 Configuring and Managing the CLIO/S Tape Pseudo Device Driver

Note

Before using the procedures in this chapter, you must have installed CLIO/S on MVS, downloaded its files to the AIX workstation, and installed the optional CLIO/S tape server feature on AIX.

CLIO/S provides a standard AIX interface to the CLIO/S tape server, `TAPESRV`, enabling users to access tape data on remote MVS and RISC

System/6000 systems through standard UNIX commands, for example, tar, cpio, or backup.

You make this interface available to users on an AIX host by configuring a CLIO/S tape mount daemon for each tape device you want to make available. This is an optional step. If you do not want to provide the standard UNIX interface to CLIO/S tapes, skip this section. If CLIO/S's optional tape server feature is installed, the CLIO/S tape API is available, regardless of whether you perform the tasks in this section to provide access through UNIX commands.

You do not have to do any special installation or configuration on MVS, even if the tape drives you want to share are physically attached to the MVS host.

Note: CLIO/S supports the “norewind” tape special file (*/dev/pstn.1*). However, you have to create only the */dev/pstn* tape pseudo device driver; you specify whichever device is appropriate, */dev/pstn* or */dev/pstn.1*.

8.7.1 Configuring and Starting the CLIO/S Tape Mount Daemon

You configure and start one CLIO/S tape mount daemon for each tape device you want to share between hosts. The tape mount daemon runs on the AIX host on which you want to create a standard UNIX interface, even if the tape device is physically attached to an MVS host.

The sample CLIO/S configuration in Figure 33 on page 190 shows one tape device physically attached to one AIX system. In this section we describe how to configure riscnode3 to access the tape device. The steps to configure and start the tape mount daemon are:

1. Login with root authority.
2. If your installation has started more than one CLIO/S PARSRV server on the host to which the tape drive is attached, set the CLIOPSPT environment variable to the appropriate port number. For example, if you want the tape mount daemon to use a PARSRV server that is listening on port 1955 and you are running on the Korn shell, set the environment variable with the command:

```
export CLIOPSPT=1955
```

If you are using the C shell, use:

```
setenv CLIOPSPT 1955
```

Multiple tape mount daemons can use the same PARSRV.

Note: PARSRV must be running at this point. If you have not already started PARSRV, refer to 8.6.4, "The CLIO/S PARSRV Server" on page 192.

If PARSRV is registered in /etc/services on AIX, you must set the CLIO/SID environment variable to the name corresponding to the server and port combination you want to use.

3. Invoke SMIT to perform device configuration:

```
smit
```

Our example shows SMIT panels from AIX 3.2.5. If your operating system is at a different level, the panels you see may be slightly different.

4. On the Systems Management panel, select Devices.
5. On the Devices panel, select Tape Drive.

Note: You can skip directly to this menu by using the `smit tape` command.

6. On the Tape Drive panel, select Add a Tape Drive.
7. In the Single Selection List of device types, select `cliotape`.

Note: If your menu does not list the `cliotape` type, the CLIO/S tape pseudo device driver was not installed correctly. Exit from SMIT and try installing again, as described in the Program Directory.

An Add a remote tape menu similar to that in Figure 34 is displayed. The fields you fill in or modify are shown in boldface.

Add a remote tape	
Type or select values in entry fields. Press Enter AFTER making all desired changes.	
Tape drive type	cliotape
Tape drive interface	remote
Description	CLIO remote tape
Remote host	riscnode1
Allocation parameter / device name	/dev/rmt0
Maximum block size	32768
Unallocate on close ?	no
Delayed open ?	yes
Default volser for allocation	DUMMY

Figure 34. Adding a Remote Tape for CLIO/S

8. Change the following fields on the Add a remote tape panel:

- a. Update the Remote host field with the name of the remote host to which the tape drive is attached. The remote host used in our example is riscnode1.

If you are using a routed nickname, it must be defined in the PARCONF file on the local AIX system. Section 8.6.3.1, "The PARCONF File" on page 191 explains how to specify nicknames for intermediate hosts.

Note: For best performance, do not use CLIO/S tape support over an Ethernet or Token-Ring TCP/IP connection. If you specify a PARCONF nickname in this field, it should not be a name that specifies the TCP/IP protocol. If you specify an Internet name:

- When connecting MVS and AIX systems, ensure that an ESCON or parallel channel attachment adapter is available
- When connecting AIX machines, ensure that a TCP/IP connection is available to use a fast connection such as FDDI.

- b. Update the Allocation parameter / device name field.

AIX For tape drives physically attached to AIX, supply the name of the tape device on the remote host, in our example, /dev/rmt0.

MVS For tape drives physically attached to MVS, supply the MVS allocation parameters separated by underscores. For example, to define the DD name of TAPE01 on a 3480 tape drive using no label (NL) processing, enter DD=TAPE01_UNIT=3480_NL_DSN=*dataset-name* in this field. Users will want to know the default MVS data set name.

Ensure that the machine type field in the PARCONF file accurately defines the physical host type of the system to which the device is physically attached. If the field and type do not match, you can complete this configuration, but users will not be able to allocate the device using the cftalloc command.

- c. Update the Unallocate on close ? field, specifying *yes* or *no*.

An administrator must unallocate tapes using SMIT. With NetTAPE, the allocation and unallocation are done dynamically by the application or end user.

- yes When a command (for example, tar) issues a close, the tape device is unallocated (except for AL and SL tapes).
 - no When a command (for example, tar) issues a close, the tape device is not automatically unallocated; the tape stays mounted.
- d. Update the Delayed open ? field, specifying *yes* or *no*.
- yes The tape device waits for a read, write, or `ioctl` before the device is opened. This is useful for backup operations.
 - no The device will be opened when allocated without waiting for a read or write.
- e. Update the Default volser for allocation field.
- AIX This field is ignored on AIX.
 - MVS Specifies the default volser for tape allocation.
9. Press Enter or select "Do" to continue.

After you complete these steps, SMIT automatically configures the CLIO/S tape pseudo device driver and starts the CLIO/S tape mount daemon. To configure additional tape drives, repeat steps 6 through 9.

SMIT automatically names the tape pseudo device drivers, starting with `/dev/pst0` for the first and continuing, if defined, up to `/dev/pst99`. The UNIX file permissions for the CLIO/S tape pseudo device driver allow all users to read and write to the device. To restrict access to the tape pseudo device driver, use standard UNIX access control methods, for example, `chmod` and `chown`.

8.7.2 Monitoring Tape Mount Daemon Activity

The tape mount daemon records information about its activities by:

- Writing to a log file, one file per daemon
- Using the AIX error logging facility

The tape mount daemon writes informational and error messages to the `/tmp/cltclntd.pstn` log file. For example, the tape mount daemon associated with tape pseudo device driver `/dev/pst1` records activities in the `/tmp/cltclntd.pst1` log file. When you stop a daemon, the log remains on the system; when you restart the daemon, the new log replaces the old one.

Using the AIX error logging facility, the tape mount daemon puts records into the system error log. Read the system error log by using the `errpt` command. This log contains information about all CLIO/S tape mount daemons and is not affected by stopping or restarting a daemon.

8.7.3 Stopping the CLIO/S Tape Mount Daemon

Stopping the tape mount daemon prevents users from using standard UNIX commands to access CLIO/S tapes. You do not have to stop the daemon during normal system operation. Circumstances that might require you to stop the tape mount daemon include:

- You want to change the characteristics of the tape pseudo device driver.
- There are hardware problems with the physical device.

Stop the tape mount daemon with the `cltterm` command by following these steps:

1. Login with root authority.
2. Ensure that the device is not being used by any users on your system. If necessary, use `cltunall` to unallocate it.

The `/tmp/cltclntd.pstn` file shows whether the device is currently allocated.

3. Use `cltterm` to stop the tape mount daemon.

```
cltterm -f /dev/pst0
```

To restart the daemon, follow the procedures in section 8.7.5, “Reconfiguring the Tape Pseudo Device Driver and Restarting the CLIO/S Tape Mount Daemon” on page 201.

8.7.4 Changing the CLIO/S Tape Pseudo Device Driver

If you want to change the default allocation parameters, change the CLIO/S tape pseudo device driver. To override the defaults for a specific tape, use `cltalloc`'s command line flags. Permanently change the daemon by following these steps:

1. Login with root authority.
2. Make sure the tape device is not being used by any users on your system. If necessary, use `cltunall` to unallocate it.

The `/tmp/cltclntd.pstn` file shows whether the device is currently allocated.

Changing the characteristics of the tape pseudo device driver while it is allocated to a user has unpredictable results.

3. Use `cltterm` to stop the tape mount daemon, as shown in section 8.7.3, “Stopping the CLIO/S Tape Mount Daemon.”

4. Use SMIT panels to change the tape pseudo device driver:

```
smit tape
```

5. On the Tape Driver panel, select Change / Show Characteristics of a Tape Drive.

Verify that the tape drive is listed as “Defined” (rather than “Available”). If the device is not “Defined,” change it to “Defined” and then return to this step and continue.

6. Select your tape drive from the Single Selection List that is displayed.
7. On the Change / Show Characteristics of a Remote Tape Drive panel, make the desired changes.
8. Press Enter or select “Do” to change the characteristics of the daemon.
9. Reconfigure the tape pseudo device driver and restart the tape mount daemon as described in section 8.7.5, “Reconfiguring the Tape Pseudo Device Driver and Restarting the CLIO/S Tape Mount Daemon.”

8.7.5 Reconfiguring the Tape Pseudo Device Driver and Restarting the CLIO/S Tape Mount Daemon

When you add a new tape pseudo device driver, SMIT automatically configures the pseudo device driver and starts the tape mount daemon. You have to configure and restart after you have stopped the daemon or unconfigured the pseudo device driver, following these steps:

1. Login with root authority.
2. If your installation has started more than one CLIO/S PARSRV server on the host to which the tape drive is attached, set the CLIOPSPT environment variable to the appropriate port number. For example, if you want the tape mount daemon to use a PARSRV server that is listening on port 1955 and you are running on the Korn shell, set the environment variable with this command:

```
export CLIOPSPT=1955
```

If you are using the C shell, use:

```
setenv CLIOPSPT 1955
```

Multiple tape mount daemons can use the same PARSRV.

If PARSRV is registered in /etc/services on AIX or in TCPIP.profile.TCPIP on MVS, you must set the CLIOPSID environment variable to the name corresponding to the server and port combination you want to use.

3. Invoke SMIT to perform device configuration:

```
smit tape
```
4. On the Tape Drive panel, select Configure a Defined Tape Drive.
5. Select your tape drive from the Single Select List menu that appears.

You can also restart the daemon by using the `cltclntd` command. For example, restart the daemon associated with tape pseudo device driver `/dev/pst0` with these commands:

```
su root
cltclntd -f /dev/pst0
```

8.7.6 Removing the CLIO/S Tape Pseudo Device Driver

You can remove the CLIO/S tape pseudo device driver. Typically removal is necessary only if you physically remove the tape device from the remote host.

8.8 Administrative Commands

For AIX to AIX environments, administrators only have two commands to control CLIO/S: `cltclntd` and `cltterm`.

8.8.1 `cltclntd`—Starting the CLIO/S Tape Mount Daemon

We describe the format of the `cltclntd` command, its parameter, purpose, usage, and authorization.

8.8.1.1 Format

The format of the `cltclntd` command is:

```
cltclntd -f device
```

8.8.1.2 Parameter

The `-f device` parameter specifies the tape device. For both AIX and MVS, this is the pseudo device associated with the desired physical device (for example, `/dev/pst0`).

8.8.1.3 Purpose

The `cltclntd` command starts the tape mount daemon. It is automatically issued by SMIT when you configure the tape pseudo device driver. The tape pseudo device driver signals the tape mount daemon whenever one of the supported I/O action occurs. The tape pseudo device driver signals the tape mount daemon by issuing the appropriate API function, such as open, read, write, and close.

In addition, the tape pseudo device driver passes requests to the API to obtain tape device parameters defined in the ODM database and to service `ioctl` requests, such as STFSF, STREW, and STEOF.

8.8.1.4 Usage

The configuration manager automatically restarts the tape mount daemon when the system is rebooted, using configuration files `/etc/PARCONF` and `/.NETRC`.

8.8.1.5 Authorization

You must have root authority to use the `cltclntd` command.

8.8.2 cltterm—Stopping the CLIO/S Tape Mount Daemon

We describe the format of the `cltterm` command, its parameter, purpose, and authorization.

8.8.2.1 Format

The format of the `cltterm` command is:

```
cltterm -f device
```

8.8.2.2 Parameters

The `-f device` parameter specifies the tape device associated with this daemon's instantiation (for example, `/dev/pst0`).

8.8.2.3 Purpose

The `cltterm` command signals the pseudo device driver to terminate the local tape client, that is, the `cltclntd` daemon.

8.8.2.4 Authorization

You must have root authority to use the `cltterm` command.

8.9 End-User Commands

Only two commands exist for the end user: `cltalloc` and `cltunall`. We describe the overall steps an end user must perform to allocate and unallocate a tape device. We then describe the details of the `cltalloc` and `cltunalloc` commands.

8.9.1 Accessing CLIO/S Tapes with AIX Commands and Subroutines

The CLIO/S tape command interface lets you access CLIO/S tapes through standard AIX commands or programs. Before you can use the the CLIO/S tape command interface, an administrator must install and configure the CLIO/S tape pseudo device driver and start the tape mount daemon, as described in section 8.7, "Configuring and Managing the CLIO/S Tape Pseudo Device Driver" on page 195. Here are the steps you perform to access CLIO/S tapes with UNIX commands and function calls. Before you

can read from or write to a tape device, you must establish a connection between your host and a CLIO/S tape server on the remote host.

1. Use the `ctalloc` command to specify the tape device you want.

`ctalloc` establishes a connection with the remote host, enabling you to access it through the tape mount daemon. If delayed allocation was specified when a tape mount daemon was configured, the tape device will be allocated on a systemwide basis. Anyone on the system with appropriate file permissions can access that tape device. If immediate allocation had been specified, the device will be opened for the specific user who issues the `ctalloc` command. Only that user can access that tape device. See section 8.7.1, "Configuring and Starting the CLIO/S Tape Mount Daemon" on page 196 for more information about configuration.

If the tape drive you want is physically attached to MVS, CLIO/S on your host talks to a PARSRV program on the remote MVS system. PARSRV on the remote MVS host starts a tape server on that same MVS system.

If the tape drive you want is physically attached to a remote AIX host, CLIO/S on your host uses the `rexec` command to start a tape server on the remote host. If the tape drive is physically attached to your local AIX system, CLIO/S does not require a tape server.

2. Use standard UNIX commands or function calls to access the tape device.

For example, you can use `tar` to back up a file system, or you can write a UNIX application program that sends `ioctl` requests to the tape device.

The tape mount daemon on your host intercepts the tape I/O requests that result from your command or program and translates them into CLIO/S calls to the tape server on the remote host.

The remote tape server sends tape I/O commands to the tape drive and returns status and data to the daemon.

The daemon passes the response back to you.

3. Issue additional AIX commands or I/O calls, by repeating step 2.

4. Release the tape drive, using the `ctunall` command.

When you no longer need access to the tape drive, use the `ctunall` command to unallocate the tape and stop the remote tape server if one was created for you.

You cannot create AL tape on AIX. You can create the AL tape with the IEHINITT MVS program. After the label is written, you can write data to the tape, using either the tape API subroutines or the AIX tape pseudo device

driver interface. An AL (or SL) tape rewinds but does not unload when the AIX command completes. The tape is unloaded when you issue `cltunall`.

8.9.2 `cltalloc`—Allocating a CLIO/S Tape Device

We describe the format of the `cltalloc` command, its parameters, purpose, usage, and examples.

8.9.2.1 Format

The format of the `cltalloc` command is:

```
cltalloc  [-v [vol=]volser] [-d [dsn=]dsname] [-p new | old] [-c]
           [-l nl | al | blp | sl] [[-e [ex=]yyddd] | [-r [ret=]nnnn]]
           -f device
```

8.9.2.2 Parameters

The person who configures the tape device daemon can specify default allocation parameters when the tape device daemon is configured. You can accept the defaults or use the parameters to override them. Your system administrator can tell you which defaults were set, or you can use SMIT to examine them.

Here are the parameters available under AIX:

-f device Specifies the tape device. For both AIX and MVS tape devices, this is the pseudo device associated with the desired physical device (for example, `/dev/pst0`).

This is a required parameter.

If the tape drive is physically attached to MVS and you are using cataloged MVS data sets, do not use the tape in no-rewind mode on AIX (that is, do not specify the device as `/dev/pst0.1`). Writing multiple files in this mode has unpredictable results. To write multiple files to a cataloged MVS data set using the tape pseudo device driver, use a separate `cltalloc` and `cltunall` for each file.

-l nl | al | blp | sl For MVS tapes, indicates the tape label type. If omitted, the value specified in the allocation data field in SMIT is used. If the label attribute was not specified when the device was configured, the default is `sl`.

You may type `nl`, `al`, `blp`, and `sl` in upper, lower, or mixed case; it is converted to upper case.

Notes:

- At the completion of the UNIX command, al and sl tapes rewind automatically but do not unload.
- For al tapes, CLIO/S supports ANSI version 3 standard labels.

-v [vol=] *volser* For AIX or MVS tapes, specifies the volume serial number of the tape volume. If omitted, the default is the *volser* the system administrator specified when configuring the daemon, by default, DUMMY.

The value you specify for *volser* is converted to upper case.

8.9.2.3 Purpose

Use the `cltalloc` command to specify which tape drive you want to use. The tape pseudo device driver uses the parameters from the command to set up internal control blocks and start the remote tape server. For MVS tapes, the parameters refer to the data definition (DD) statement used to describe tape devices on MVS. Up to 10 tape units may be allocated with successive calls. MVS tapes are dynamically allocated.

The command waits until the operator has mounted the requested tape on the requested device. You must wait until the command completes and you receive a message that the `cltalloc` command is complete before you can issue commands that use the tape drive.

8.9.2.4 Usage

CLIO/S supports using AIX tapes as special files `/dev/pstn` and `/dev/pstn.1` (no rewind mode). You must have a NETRC file defined. See section 8.6.3.2, "The NETRC File" on page 191 for information about creating a NETRC file.

8.9.2.5 Examples

Here are some `cltalloc` examples:

- To allocate AIX tape device `/dev/pst0` when no *volser* is required, issue:

```
cltalloc -f /dev/pst0
```
- To allocate the MVS tape device associated with `/dev/pst0`, mounting a previously created tape with *volser* `mytape` and data set name `backup.stuff`, issue:

```
cltalloc -p old -v mytape -d backup.stuff -f /dev/pst0
```
- To allocate the same tape device as in the previous example, back up the `$HOME/mydata` from AIX to MVS, then unallocate the tape device:

```
cltalloc -p old -v mytape -d backup.stuff -f /dev/pst0
tar -cvf /dev/pst0 $HOME/mydata
cltunall -f /dev/pst0
```

- The following command allocates the new cataloged data set 'galina.recent', using AIX tape device /dev/pst1. The tape uses standard labels and the data set expires on March 19, 1996 (Julian date 96078). Because `-v` is not specified, the default is the volume entered on the SMIT panel when the tape pseudo device driver was configured.

```
cltalloc -p new -c -l sl -e 96078 \
-d 'galina.recent' -f /dev/pst1
```

8.9.3 cltunall—Unallocating a CLIO/S Tape Device

We describe the format of the `cltunall` command, its parameter, purpose, and an example.

8.9.3.1 Format

The format of the `cltunall` command is:

```
cltunall -f device
```

8.9.3.2 Parameters

The `-f device` parameter specifies the tape device being unallocated. For both AIX and MVS, this is the pseudo device associated with the desired physical device (for example, /dev/pst0).

8.9.3.3 Purpose

The `cltunall` command stops the remote tape server and unallocates the tape device.

8.9.3.4 Example

To unallocate the CLIO/S tape device /dev/pst1, issue:

```
cltunall -f /dev/pst1
```

8.10 CLIO/S Positioning

CLIO/S provides price and functional advantages over other high-speed data transfer products. It is a simple, channel-attached protocol requiring no offload boxes, hubs, or other hardware. It provides multiple APIs to choose from, for example, a standard FTP-like command, BSD sockets-like interface, named and batch pipes, and other distributed programming interfaces.

A customer's application development effort is minimized by using the AIX tape commands or named pipes, for example. Regardless of where the

data is located (for example, on tape or DASD), CLIO/S has a method to move it.

Appendix A. ADSM Supported Tape Drives and Libraries

The ADSM for AIX server supports a wide range of tape drives and libraries. Here is the latest list at the time of writing. Please check the ADSM server README file, contact your IBM representative, or access the ADSM home page on the World Wide Web (<http://www.storage.ibm.com/adsm/>) for the most current product information.

With the purchase of a base license, the following device support is provided:

- Disk drives
- 4 mm and 8 mm tape
- QIC
- Optical drives

Additional device support upgrades (Modules 1-4) are also available. Device support modules enable an ADSM for AIX server to support storage devices that are not supported in the base product (B). Device modules are inclusive. For example, Device Module 2 includes support for devices in Modules 1 and 2. Device Module 3 includes support for devices in Modules 1, 2, and 3. Device Module 4 includes support for devices in Modules 1, 2, 3, and 4.

Reference the *ADSM for AIX Price Flyer* (G225-6639) for pricing information.

Device Description	Capacity (in GB or max slots)	Device Module
Direct Access Storage		
ADSM for AIX should support any disk supported by the given operating system		B
4 mm Tape Drives		
Hewlett-Packard 35470A	2 GB	B
Hewlett-Packard 35480A	2 GB ¹	B
Hewlett-Packard C1533A	4 GB ¹	B
Hewlett-Packard JetStore 2000e	2 GB	B
Hewlett-Packard JetStore 5000e	2 GB ¹	B
Hewlett-Packard Jetstore 6000e	4 GB ¹	B

Device Description	Capacity (in GB or max slots)	Device Module
IBM 7206-001	2 GB	B
IBM 7206-005	4 GB ¹	B
Sony SDT-5000	4 GB ¹	B
Sony SDT-7000 ⁴	4 GB ¹	B
WangDAT 3300DX	2 GB	B
WangDAT 3400DX	4 GB ¹	B
4 mm Autochangers¹		
ADIC 1200D ⁵	48 GB	1
ADIC VLS 4mm	60 GB	2
AIWA AL-D210 ⁵	68 GB	1
AIWA AL-D220	68 GB	2
Diverse Logistics Libra 8 ⁵	32 GB	1
Diverse Logistics Libra 16 ⁵	64 GB	1
Exabyte 018/218	72 GB	2
Hewlett Packard C1553A ⁵	24 GB	1
Hewlett Packard C1561A ⁵	24 GB	1
Hewlett Packard SureStore 12000e ⁵	24 GB	1
IBM 7332-005 4 or 12 slot	16/48 GB	1
IBM 7336-205 ⁴	112 GB	2
Spectrallogic 4000/20	80 GB	2
Spectrallogic 4000/40	160 GB	2
Spectrallogic 4000/60	240 GB	3
StorageTek 9704	100 GB	2
8 mm Tape Drives		
Andataco Encore 8205	2.3 GB ¹	B

Device Description	Capacity (in GB or max slots)	Device Module
Andataco Encore 8505	5G B ¹	B
Dynatek HSB 2300	2.3 GB ¹	B
Dynatek HSB 5000	5 GB ¹	B
Dynatek HSB 10.0	5 GB ¹	B
Exabyte EXB-8200	2.3 GB	B
Exabyte EXB-8205	2.3 GB ¹	B
Exabyte EXB-8205XL	3.5 GB ¹	B
Exabyte EXB-8500	5 GB	B
Exabyte EXB-8500C	5 GB ¹	B
Exabyte EXB-8505	5 GB ¹	B
Exabyte EXB-8505XL	7 GB ¹	B
IBM 7208-001	2.3 GB	B
IBM 7208-011	5 GB ¹	B
SUN 8505XL	7 GB ¹	B
Transitional Technologies, Inc. CTS-8000H	5 GB ¹	B
Transitional Technologies, Inc. CTS-8519H	5 GB or 10 GB	B
8 mm Autochangers¹		
Andataco Encore 10e⁵	50 GB	1
Andataco Encore 210	70 GB	2
Andataco Encore 440	287 GB	2
Andataco Encore 120	580 GB	3
Andataco Encore 480	567 GB	3
BoxHill Bread Box⁵	50 GB	1
BoxHill Cube Box	287 GB	2
BoxHill Light Box	70 GB	2
BoxHill Freezer Box	300 GB	2

Device Description	Capacity (in GB or max slots)	Device Module
BoxHill Borg Box	567 GB	3
BoxHill Ice Box	580 GB	3
Exabyte EXB-10i ⁵	50 GB	1
Exabyte EXB-10e ⁵	50 GB	1
Exabyte EXB-10h ⁵	70 GB	1
Exabyte EXB-60	300 GB	2
Exabyte EXB-210	70 GB	2
Exabyte EXB-440	287 GB	2
Exabyte EXB-120	580 GB	3
Exabyte EXB-480	567 GB	3
IBM 7331-205	140 GB	2
LAGO LS/380L DataWheel	270 GB	2
StorageTeK 9708	270 GB	2
StorageTeK 9711	378 GB	2
Transitional Technologies, Inc. Series 10 ⁵	50GB	1
QIC Tape Drives		
IBM 7207-012	1.19 GB	B
Sony DTF Tape Drives¹		
Cybernetics CY-9000 ⁴	12 GB or 42 GB	2
DLT Tape Drives		
DEC TZ87/TZ87N ⁴	10 GB ¹	1
DEC TZ88/TZ88N ⁴	20 GB ¹	1
Quantum DLT 2000	10 GB ¹	1
Quantum DLT 4000	20 GB ¹	1
DLT Tape Libraries¹		
ATL Products (Odetics) ACL 2/28	280 GB	2

Device Description	Capacity (in GB or max slots)	Device Module
ATL Products (Odetics) ACL 4/52	520 GB	3
ATL Products (Odetics) ACL 9/88	880 GB	4
ATL Products (Odetics) ACL 6/176	1.7 TB	4
ATL Products (Odetics) ACL 2640	2.5 TB	4
DEC TZ875/TZ875N ^{4, 5}	50 GB	2
DEC TZ877/TZ877N ^{4, 5}	70 GB	2
DEC TZ885/TZ885N ^{4, 5}	100 GB	2
DEC TZ887N/TZ887N ^{4, 5}	140 GB	2
Quantum DLT 2500 ⁵	50 GB	2
Quantum DLT 2700 ⁵	70 GB	2
Quantum DLT 4500 ⁵	100 GB	2
Quantum DLT 4700 ⁵	140 GB	2
Transitional Technologies, Inc. Series Q7 ⁴	280 GB	2
Transitional Technologies, Inc. Series Q47 ⁴	600 GB	3
Transitional Technologies, Inc. Series Q47 ⁴ (more than 2 drives)	1.2 TB	4
SCSI-Attached Tape Transports¹		
IBM 3490E Model C10	800 MB	2
IBM 3490E Model C11	800 MB	2
IBM 3490E Model C1A	800 MB	2
IBM 3490E Model C22	800 MB	2
IBM 3490E Model C2A	800 MB	2
IBM 3490E Model E01	800 MB	2
IBM 3490E Model E11	800 MB	2
IBM 3590 Model B11 (Manual)	10 GB	2
IBM 3590 Model B11 (Random Mode)	10 GB	3
SCSI-Attached Tape Library Dataservers		

Device Description	Capacity (in GB or max slots)	Device Module
IBM 3494 Model L10 Single frame	240 slots	3
IBM 3494 Model L12 Single frame	240 slots	3
IBM 3494 Model L10 Multi frame	3,040 slots	4
IBM 3494 Model L12 Multi frame	3,040 slots	4
Parallel-Attached Tape Transports¹		
IBM 3480 Model B11	200 MB	2
IBM 3480 Model B22	200 MB	2
IBM 3490/3490E Model B02	200 MB	2
IBM 3490/3490E Model B04	200 MB	2
IBM 3490/3490E Model B20	800 MB	2
IBM 3490/3490E Model B40	800 MB	2
IBM 3490/3490E Model C10	800 MB	2
IBM 3490/3490E Model C11	800 MB	2
IBM 3490/3490E Model C1A	800 MB	2
IBM 3490/3490E Model C22	800 MB	2
IBM 3490/3490E Model C2A	800 MB	2
IBM 3490/3490E Model D31	200 MB	2
IBM 3490/3490E Model D32	200 MB	2
IBM 3490/3490E Model D41	800 MB	2
IBM 3490/3490E Model D42	800 MB	2
Parallel-Attached Tape Library Dataservers		
IBM 3494 Model L10 Single frame	240 slots	3
IBM 3494 Model L10 Multi frame	3,040 slots	4
IBM 3495 Model L20	6,440 slots	4
IBM 3495 Model L30	10,590 slots	4
IBM 3495 Model L40	14,750 slots	4
IBM 3495 Model L50	18,910 slots	4
IBM 3495 Model M10	100,000 slots ²	4
ESCON-Attached Tape Transports¹		

Device Description	Capacity (in GB or max slots)	Device Module
IBM 3490/3490E Model B20	800 MB	2
IBM 3490/3490E Model B40	800 MB	2
IBM 3490/3490E Model C10	800 MB	2
IBM 3490/3490E Model C11	800 MB	2
IBM 3490/3490E Model C22	800 MB	2
IBM 3490/3490E Model C1A	800 MB	2
IBM 3490/3490E Model C2A	800 MB	2
ESCON-Attached Tape Library Dataservers		
IBM 3494 Model L10 Single frame	240 slots	3
IBM 3494 Model L10 Multi frame	3,040 slots	4
IBM 3495 Model L20	6,440 slots	4
IBM 3495 Model L30	10,590 slots	4
IBM 3495 Model L40	14,750 slots	4
IBM 3495 Model L50	18,910 slots	4
IBM 3495 Model M10	100,000 slots ²	4
Optical Drives		
HP Model 1300T	1.3 GB	B
HP Model 1716T	1.3 GB	B
IBM 0632-C1/AAA ³	650 MB	B
IBM 0632-C2A/ACA ³	1.19 GB	B
IBM 0632-C2B/CCA ³	1.19 GB	B
IBM 0632-CHA/CHB/CHC ³	1.19 GB	B
IBM 7209 Model 001	650 MB	B
IBM 7209 Model 002	1.19 GB	B
IBM 3510 Model 001	1.19 GB	B
Kodak System 2000 ⁴	14 GB	B
Philips LMS LD4100 ⁴	5.6 GB	B
Philips LMS LD6100 ⁴	12 GB	B

Device Description	Capacity (in GB or max slots)	Device Module
Sony SMO-F541 ⁴	2.6 GB	B
Optical Jukeboxes (Native SCSI Rewritable, and WORM)		
ATG Cygnet Series 1602 ⁴	29 Max slots	2
ATG Cygnet Series 1802 ⁴	61 Max slots	3
ATG Cygnet Series 1803 ⁴	141 Max slots	4
DISC DocuStore D150U-1	170 Max slots	3
DISC DocuStore D245-1	247 Max slots	4
DISC DocuStore D255-1U	247 Max slots	4
DISC DocuStore D280U-1	290 Max slots	4
DISC DocuStore D350-1U	340 Max slots	4
DISC DocuStore D510-2	514 Max slots	4
DISC DocuStore D525-1	517 Max slots	4
DISC DocuStore D525-1U	517 Max slots	4
DISC DocuStore D1050-2	1054 Max slots	4
HP Model 20T ³	16 slots	1
HP Model 40T	32 slots	2
HP Model 120T	88 slots	3
HP Model 200T	144 slots	3
IBM 3995 Model A63 ⁵	16 slots	1
IBM 3995 Model C60 ^{4, 5}	20 Max slots	2
IBM 3995 Model 063	32 slots	2
IBM 3995 Model C62 ⁴	52 Max slots	2
IBM 3995 Model 163	144 slots	3
IBM 3995 Model C64 ⁴	104 Max slots	3
IBM 3995 Model C66 ⁴	156 Max slots	3
IBM 3995 Model C68 ⁴	258 Max slots	4
Philips LMS 6600 RapidChanger ⁴	6 Max slots	1

Device Description	Capacity (in GB or max slots)	Device Module
Philips LMS 6602 RapidChanger ⁴	12 Max slots	2
<p>Note:</p> <p>1 These devices support compression. Capacities given here are noncompressed.</p> <p>2 This is a stand-alone library manager.</p> <p>3 JukeBox drives, not available as stand-alone drive</p> <p>4 Requires 3.2.5 PTF IP20758 and 4.1.4 PTF U443880</p> <p>5 <u>Single Drive Library</u></p> <p>Note: There are some functional limitations when using libraries with only one drive. Some operations (such as space reclamation) require multiple drives.</p> <p>Note: Two tape transports are strongly recommended, to function as one logical library under one tape mounting mechanism (manual or automated), in order to support the reclamation feature.</p>		

Appendix B. Records at the Beginning of a Labeled Tape

For a named volume with volume serial number FCT101, there would be three EBCDIC-encoded 80-byte records at the beginning of tape. For example:

```
VOL1FCT101
HDR1ADSM.BFS.V0000004      0001      96019 993650000000ADSM
HDR2U3276800000 0
```

IBM standard labels have the following fields:

- Entries marked with “->” are checked for existing values before being altered. If the field is in use, the original contents are left unaltered.
- Fields marked with “>>” are overwritten.
- Fields marked with “--” may be examined but not changed.
- Fields marked with “BB” are set to blanks.

VOL1 record:

```
char      labelId[3];          /* -- */
char      labelNum[1];       /* -- */
char      volSer[6];        /* -- */
char      reserved_1[1];    /* BB */
char      vtocPointer[10];  /* BB */
char      reserved_2[13];   /* BB */
char      recTechnique[2];  /* >> */
char      reserved_3[5];    /* BB */
char      ownerName[10];    /* -- */
char      reserved_4[29];   /* BB */
```

HDR1 record:

```
char      labelId[3];          /* >> */
char      labelNum[1];       /* >> */
char      dsn[17];          /* -> */
char      dsSerialNum[6];    /* BB */
char      volSeqNum[4];     /* BB */
char      dsSeqNum[4];      /* >> */
char      genNum [4];       /* BB */
char      versionNum[2];    /* BB */
char      createDate[6];    /* -> */
char      expireDate[6];    /* -> */
char      security[1];      /* -> */
char      blockCount[6];    /* -> */
```

```

char          systemCode[13];      /* >> */
char          reserved_1[7];      /* BB */

```

HDR2 record:

```

char          labelId[3];          /* >> */
char          labelNum[1];         /* >> */
char          recfm[1];            /* >> */
char          blockLength[5];      /* >> */
char          recordLength[5];     /* >> */
char          tapeDensity[1];      /* >> */
char          dsPosition[1];       /* >> */
char          jobStep[17];         /* BB */
char          recTechnique[2];     /* >> */
char          ctlChar[1];          /* >> */
char          reserved_1[1];        /* BB */
char          blockAttr[1];        /* >> */
char          reserved_2[8];       /* >> */
char          ckptId[1];           /* >> */
char          reserved_3[32];      /* BB */

```

Appendix C. Single Drive Reclamation Process

Here is a copy of the script to perform single drive reclamation. It is provided AS-IS with no support. It can also be found on the ADSM ftp server as SDRECLM AIXBIN. You can download the file in binary to a file named sdreclaim.

```
=====
#!/bin/ksh
#
# sdreclaim - script that automates manual reclamation for
#             single drive sequential access storage pools
# Adapted from:
# mrecsv - Manual Reclamation of Sequential access Volumes
# Written By  : Van Stokes, Jr 11/13/1995
# Last Updated : Mike Kaczmariski, IBM 1/3/1996
#
# DESCRIPTION:
# This script was made available by Van Stokes Jr, and modified slightly
# by IBM to provide a means for reclamation in ADSM single-drive
# sequential storage pools.
#
# SUPPORT:
# This script is being provided AS IS - no warranties or support is
# provided. There are no warranties, express or implied, including
# the warranties of merchantability and fitness for a particular
# purpose.
#
# ***** Work Flow - What is happening? *****
#
# The ADSM server, both version 1.0+ and 2.0+ are not capable of doing a
# reclamation on a sequential access devices (tape, optical, etc) that only
# have one drive. Nor, to my knowledge, do they have any plans to make it
# possible to do so. Because I was in a bind and I didn't have the time to
# manually move data between volumes all the time, I wrote this script to
# automate this procedure - Manual Reclamation of Sequential Access Volumes.
#
# Requirements:
# Sequential Access Media - tape, optical, etc.
# Random Access Media    - Device Class of Disk
# Storage Pool for Manual Reclamation
#
# How it works.....
# The script finds volumes on your sequential storage pool (VSP) whose
# occupancy is *LOWER* than 100-(Reclamation Threshold) and are in
# the 'filled state'.
# When the script encounters such a volume, it then proceeds to move the
```

```

# data off of the VSP volume on to the manual reclamation storage pool (MRSP).
#
# If the MRSP gets full, ADSM will (if set) automatically migrate the data
# back to the VSP thus consolidating the data.
#
# The MRSP must have enough available space to hold the reclaimable
# data that need to be moved to empty a volume.
#
# If the MRSP doesn't get full, that's OK. The script will automatically move
# the data back to the VSP - again, consolidating the data.
#
# And the process repeats itself until there are no more volumes in the VSP
# below the "Reclamation Threshold".
#
# It's really a "round-robin". Data is moved from a partially full sequential
# access media volume (tape, optical, etc) to a random access media volume
# (disk) and then back to the sequential access media (tape, optical, etc).
#
# The goal is to get sequential access media volumes that are not
# completely full and consolidate the data on to fewer volumes.
#
# !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
# YOU MUST DO THE FOLLOWING to ensure the proper operation of this script:
#
# In this Script, you must set:
#   ADSM_ID      - The Admin ID to log into ADSM. Ensure proper permissions.
#   ADSM_PASSWD - The Admin ID's password to use
#
#   MRSP         - The *DISK* storage pool name to use for reclamation of VSP.
#                 - This is where the data will be moved to temporarily.
#                 - This storage pool *CAN NOT* be used by anything else.
#                 - This storage pool should migrate to your VSP (see next item).
#                 - Caching should be turned off.
#                 - Suggested Migration Thresholds: Low: 0%, High: 98%
#
#   VSP          - The name of the Storage Pool *TO* Reclaimate.
#                 - i.e. Your Tape or Optical storage pool name.
#
# On the ADSM server, you must set:
#   Reclamation Threshold - This must be set on your VSP storage pool.
#
#   Next Storage Pool    - This must be set on your MRSP storage pool.
#                         - This should be your VSP, but doesn't have to be.
#
# In this script, Other settings to check:
#   SLTF                - Send logs(status) to the log file (Y or N)?
#   SLTS                - Send logs(status) to the screen (Y or N)?
#   LOG                 - Location of the log file (check path).

```

```

# LCKF                - Location of the lock file (check path).
# REC_TABLE           - Reclamation Table (check path).
#
# ***** Macro's that you change per system *****
#
# Login and Password to use when issuing ADSM commands.
ADSM_ID="adminuser"
ADSM_PASSWD="adminpassword"
#
# Manual Reclamation Storage Pool name (where to move the data to).
MRSP="MRSP"
#
# Storage Pool Name to Reclaimate
VSP="FILEPOOL"
#
# Send logs(status) to the log file (Y or N)?
SLTF="Y"
#
# Send logs(status) to the screen (Y or N)?
SLTS="Y"
#
# Location and Name of the Log File
LOG="/tmp/adsm.rec.log"
#
# Lock file to keep another instance of this script from running.
LCKF="/tmp/LCK.ADSMREC"
#
# Reclamation Table - a list of VSP volumes to reclaimate (auto generated).
REC_TABLE="/tmp/adsm.rectbl"
#
# ***** Macro's that should not be changed manually *****
#
# Default command that is normally issued.
DSMADMC="dsmadm -ID=$ADSM_ID -PASSWORD=$ADSM_PASSWD -NOCONFIRM"
#
# Manual Reclamation Process ID (must be zero, will be set by script)
MR_PROC_ID=0
#
# Reclamation Threshold (must be zero, will be set by script).
# This item is retrieved from the VSP Reclamation Threshold field in ADSM.
RTH=0
#
# Volume Threshold (must be zero, will be set by script).
# This is the volume threshold used to choose volumes eligible for
# reclamation. The value is the maximum volume occupancy for full
# volumes, below which volumes will be reclaimed. This is equivalent
# to 100-RTH
VTH=0

```

```

#
# Maximum Scratch Volumes Allowed (must be zero, will be set by script)
# This item is retrieved from the VSP Max Scratch Volumes field in ADSM.
MSVA=0
#
# Current Number of Scratch Volumes (must be zero, will be set by script)
CNSV=0
#
# Current Number of Volumes being filled (must be zero, will be set by script)
CNVF=0
#
# Manual Reclamation Storage Pool's (MRSP) next storage pool to send data to.
MRSP_NSP=""
#
# ***** Define Macro Functions *****
#
# Function to load information about VSP for Reclamation.
# Returns 0=success, 1=failed.
Get_VSP_Info ()
{
|Rtn=0
|#
|$DSMADMC q stg $VSP f=d " tail +8 " while read line
|do
|field=echo "$line" " awk ' FS=":" { print $1 } '
|var=echo "$line" " awk ' FS=":" { print $2 } '
|case $field in
|'|Reclamation Threshold')
|    |RTH=echo "$var" " awk ' { print $1 } '
|    |;;
|'|Maximum Scratch Volumes Allowed')
|    |MSVA=echo "$var" " awk ' { print $1 } '
|    |;;
|esac
|done

#
# Did we get everything we need to have?
#
|if $RTH -lt 1 -
|then
|Log "! Storage Pool $VSP \"Reclamation Threshold\" value is invalid."
|Rtn=1
|fi
#
|if $MSVA -lt 1 -
|then
|Log "! Storage Pool $VSP \"Maximum Scratch Volumes Allowed\" value is

```

```

invalid."
|Rtn=1
|fi

|VTH=0

|# Calculate the volume threshold (for occupancy) that will
# be used to determine if it can be reclaimed. Note that this
# is 100 - the reclamation threshold, since the threshold is
# specified in terms of percentage of empty space on the volume
|if $Rtn -eq 0 -
|then
|  VTH=expr 100 - $RTH
|  fi

|#
|return $Rtn
|}
|#
|# Function to load information about MRSP
|# Returns 0=success, 1=failed.
Get_MRSP_Info ()
{
|Rtn=0
|#
|$DSMADMC q stg $MRSP f=d " tail +8 " while read line
|do
|field=echo "$line" " awk ' FS=":" { print $1 } '
|var=echo "$line" " awk ' FS=":" { print $2 } '
|case $field in
|'|Next Storage Pool')
|'|MRSP_NSP=echo "$var" " awk ' { print $1 } '
|'|;;
|esac
|done

|#
|# Did we get everything we need to have?
|#
|if "$MRSP_NSP" = "" -
|then
|Log "! Storage Pool $MRSP must have a \"Next Storage Pool\" value."
|Rtn=1
|fi
|#
|return $Rtn
|}
|#

```

```

# Function to load the VSP volume tables for reclamation.
Load_Rec_Table ()
{
# Remove any existing tables
rm -f $REC_TABLE
#
$DSMADMC q vol " tail +12 " while read line
do
| if "$line" != "" -
| then
| #
| vol=echo "$line" " awk '{ print $1 }'"
| stgpool_name=echo "$line" " awk '{ print $2 }'"
| storage_class=echo "$line" " awk '{ print $3 }'"
| util=echo "$line" " awk '{ print $5 }'"
| status=echo "$line" " awk '{ print $6 }'"
| #
| # Does this volume belong to VSP?
# | echo "$VSP"
# | echo "$VSP" " grep $stgpool_name
| Rtn=echo "$VSP" " grep $stgpool_name
| if "$Rtn" != "" -
| then
| # If you are here, then this is a valid storage pool to work with.
| if "$status" = "Full" -
| then
| # Volume is in a status of full, might be able to reclaimate.
| echo "$util:$vol" >> $REC_TABLE
| fi
| #
| # Are we filling this volume? We need to know for CNSV
| if "$status" = "Filling" -
| then
| # yep, were filling this volume
| CNVF=expr $CNVF + 1
| fi
| fi
| fi
done
#
Rtn=?
if "$Rtn" = "0" -
then
| # got a good table, sort it and continue.
| cat $REC_TABLE " sort -n > $REC_TABLE
| return 0
else
| # Opps, something bit it

```

```

|rm -f $REC_TABLE
|return 1
fi
}
#
# Function that finds the current number of available scratch volumes
# to make sure we have enough to do a reclamation.
Calculate_CNSV ()
{
|CNUV=0
|#
|# Load the volumes into the Reclamation Table from ADSM
|Load_Rec_Table
|if $? != 0 -
|then
|Log "! Failed to load the reclamation table."
|return 1
|fi
|#
|# Make sure there is a table to go by.
|if ! -f $REC_TABLE -
|then
|# Opps, no table - how this happen?
|Log "! Reclamation Table missing. Unable to Calculate_CNSV"
|return 1
|fi
|#
|# Make sure MSVA has been set
|if $MSVA -lt 1 -
|then
|# Opps, no table - how this happen?
|Log "! MSVA is missing! Unable to Calculate_CNSV"
|return 1
|fi
|#
|# Get the number of volumes being uses from REC_TABLE
|CNUV=cat $REC_TABLE " wc -l " awk '{ print $1 }'
|# Calculate the current number of scratch volumes available
|CNSV=expr $MSVA - $CNVF - $CNUV
|# Any scratch volumes available?
|if $CNSV -lt 1 -
|then
|# Nope, out of scratch volumes to use for reclamation.
|return 1
|fi
|#
|# If you are here, all is well, lets continue.
|return 0
}

```

```

}
#
# Function that controls the reclamation
Reclamation_Loop ()
{
# Make sure there is a table to go by.
if ! -f $REC_TABLE -
then
# Opps, no table - how this happen?
return 1
fi
#
RL_RET=1
#
cat $REC_TABLE " while read line
do
vsp_util=echo "$line" " awk ' FS=":" { print $1 } '
vsp_vol=echo "$line" " awk ' FS=":" { print $2 } '
#
# Volume utilization greater than the Reclamation Threshold (VTH)?
if $vsp_util -gt $VTH -
then
# Yep, Nothing to migrate
# All volumes packed beyond the RTH.
# We'll exit the program with this return so we don't try another pack.
return $RL_RET
fi
#
# If you are here, then there is something to migrate.
#
# Lets ensure the MRSP is empty for reclamation
Empty_MRSP
#
# Lets make sure there are scratch volumes to use for reclamation.
Calculate_CNSV
if $? != 0 -
then
Log "! Insufficient scratch volumes available for reclamation."
Log "= Storage Pool: $VSP, Scratch Volumes Available: $CNSV."
return 1
fi
#
# Begin the moving of data off of this volume.
RL_RET=0
MR_PROC_ID=0
Move_Data $vsp_vol $MRSP
Rtn=$?
if $Rtn != 0 -

```



```

then
| if $Rtn = 2 -
| then
| |Log "= No data on volume $vsp_vol to move to stgpool $MRSP."
| else
| |Log "! Failed to begin a MOVE DATA on volume $vsp_vol to stgpool $MRSP."
| fi
| # We'll try another volume just in case this one is checked out.
else
|Log "+ Move Data started on vol $vsp_vol, process id is $MR_PROC_ID."
| #
| # Monitor the process, when its finished, then continue with
| # the next volume.
| while true
| do
| |sleep 120
| |# See if the move data process is still running
| |Check_Proc_Status $MR_PROC_ID
| |if $? != 0 -
| |then
| | |# Nope, lets break out and do the next volume.
| | |break
| | fi
| |# If you are here, our move data process is still running.
| done
| fi
done
#
# Sigh - that's all done. But, lets do it again to make sure we
# can't pack it even better.
return $RL_RET
}
#
# Function that moves the data from $1 (volume) to $2 (storage pool)
# $1 must be the SOURCE volume to that contains data to be moved
# $2 must be the DESTINATION storage pool
# Returns:
# 0 = Move process was successfully started
# 1 = Failed to start the move process
# 2 = No Data to move
Move_Data ()
{
| $DSMADMC move data $1 stg=$2 " tail +5 " while read line
| do
| |# Find our process id
| |msg=echo "$line" " awk ' { print $1 } '
| |#
| |# Successfully started a Move Data Job?

```

```

| if "$msg" = "ANS5104I" -
| then
| # yes, Get our process ID.
| MR_PROC_ID=echo "$line" " awk ' { print $4 } '
| return 0
| fi
| #
| # Was there any data to move?
| if "$msg" = "ANR2209W" -
| then
| # No there wasn't any data to move
| return 2
| fi
| #
done
#
# Something went wrong.
return 1
}
#
# Function to check the status of a process
# $1 must be the process number to check
Check_Proc_Status ()
{
|Rtn=0
|#
|$SMADMC q proc $1 " tail +5 " while read line
|do
|msg=echo "$line" " awk ' { print $1 } '
|if "$msg" = "ANR0942E" -
|then
| # Process is no longer running
| |Rtn=1
|fi
|done
|#
|return $Rtn
}
#
# This function checks $1 to ensure it isn't in migration
# $1 must be the storage pool to check for migration on
Check_For_Migration ()
{
|$SMADMC q stg $1 f=d " grep 'Migration in Progress' " read line
|#
|status=echo "$line" " awk ' FS=":" { print $2 } ' " awk ' { print $1 } '
|#
|if "$status" = "Yes" -

```

```

then
| # Yes, migration has commenced.
| return 1
else
| # Nope, migration is idle
| return 0
fi
}
#
# Function that ensures the MRSP is empty
Empty_MRSP ()
{
| Log "= Attempting to empty storage pool $MRSP for use in reclamation."
| #
| # Make sure the system isn't Migrating MRSP
while true
do
| Check_For_Migration $MRSP
| if $? = 0 -
| then
| # Nope, not in progress, yet.
| break
| fi
| #
| # Yep, there is a migration going. Wait until it's finished.
| Log "= Migration in progress on storage pool $MRSP, waiting."
| sleep 300
done
| #
| #
$DSMADMC q vol stg=$MRSP " tail +12 " while read line
do
| if "$line" = "" -
| then
| break
| fi
| #
| # Get the volume(s) associated with MRSP.
| mrsp_vol=echo "$line" " awk ' { print $1 } '
| #
| # Begin the moving of data off of this MRSP volume.
| MR_PROC_ID=0
| Move_Data $mrsp_vol $MRSP_NSP
| Rtn=$?
| if $Rtn != 0 -
| then
| if $Rtn = 2 -
| then

```

```

| | | | Log "= No data on volume $mrsp_vol to move to stgpool $MRSP_NSP."
| | | | else
| | | | | Log "! Failed to begin a MOVE DATA on volume $mrsp_vol to stgpool
$MRSP_NSP."
| | | | fi
| | | | # We'll try another volume just in case this one is checked out.
| | | | else
| | | | | Log "+ Move Data started on volume $mrsp_vol, process id is $MR_PROC_ID."
| | | | #
| | | | # Monitor the process, when its finished, then continue with
| | | | # the next volume.
| | | | while true
| | | | do
| | | | | sleep 120
| | | | | # Is the move data process is still running?
| | | | | Check_Proc_Status $MR_PROC_ID
| | | | | if $? != 0 -
| | | | | then
| | | | | | # Nope, lets break out and do the next volume.
| | | | | | break
| | | | | fi
| | | | | # If you are here, our move data process is still running.
| | | | done
| | | | fi
| | | done
| | #
| | return 0
| }
#
# Log information to the log file
# $1 needs to be the information to log
Log ()
{
| # Following line sends information to the LOG file.
| if "$SLTF" = "Y" -
| then
| | echo "date +%y/%m/%d date +%H:%M $1" >> $LOG
| fi
| #
| # Following line sends information to the screen.
| if "$SLTS" = "Y" -
| then
| | echo "date +%y/%m/%d date +%H:%M $1"
| fi
| fi
| }
#
# ***** Main script area *****
#

```

```

cd /usr/lpp/adsm/bin
#
# Make sure we are the only process running
if -f $LCKF -
then
|echo "A reclamation process is already running, exiting."
|exit 1
else
|echo "date" > $LCKF
|Log "+ Started."
fi
#
while true
do
|# Load MRSP information just in case it changed while we were working.
|Get_MRSP_Info
|if $? != 0 -
|then
|break
|fi
#
|# Load VSP information just in case it changed while we were working.
|Get_VSP_Info
|if $? != 0 -
|then
|break
|fi
#
|Log "= Storage Pool: $VSP, Reclamation Threshold: $RTH."
|Log "= Filled volumes with $VTH % occupancy or less will be reclaimed."
|Log "= Storage Pool: $VSP, Maximum Scratch Volumes Allowed: $MSVA."
#
|# Load the volumes into the Reclamation Table from ADSM
|Load_Rec_Table
|if $? != 0 -
|then
|Log "! Failed to load the reclamation table."
|break
|fi
#
|# Begin the reclamation
|Reclamation_Loop
|Rtn=$?
|# Can we pack any more volumes?
|if $Rtn != 0 -
|then
|break
|# Nope, volumes packed beyond RTH, Reclamation is over.

```

```
| #  
| # Lets ensure the MRSP is completely empty.  
| Empty_MRSP  
| #  
| # And that's it for this time.  
| break  
| fi  
| #  
| # If you are here, Lets try one more reclamation to make sure we can't  
| # pack the volumes even better.  
done  
#  
# Cleanup  
rm -f $REC_TABLE  
#  
Log "+ Ended."  
#  
# remove lock file  
rm -f $LCKF  
#  
# (end script)
```

Appendix D. NetTape Supported Tape Drives and Libraries

NetTAPE supports the following tape drives and libraries when they are connected to an AIX workstation. The Ampex product listed below is supported but not warranted by IBM.

- The Ampex DST 310 tape drive
- The IBM 3420 magnetic tape unit
- The IBM 3480 magnetic tape subsystem, including support for the ACL in serial mode
- The IBM 3490 magnetic tape subsystem enhanced capability models C10, C11, and C22, including support for the ACL in serial mode
- The IBM 3490 magnetic tape subsystem enhanced capability models E01 and E11, including support for the cartridge stack loader in serial mode
- The IBM 3490E magnetic tape subsystem, including support for the ACL in serial mode
- The IBM 3590 high performance tape subsystem, including support of the ACF in random or serial mode

NetTAPE also provides limited tape processing support for devices that are supported by the standard AIX SCSI tape device driver. NetTAPE does not support tape label processing or record format handling for these devices. It only allows access to these devices as no-label tapes (that is, standard UNIX format).

The following types of tape drives are currently supported by the AIX SCSI tape device driver. You can use SMIT to list all of the supported tape drives and to configure them to AIX.

- 4 mm tape drives of various capacities, such as the IBM 7206 Model 005 external 4 mm tape drive
- 8 mm tape drives of various capacities, such as the IBM 7208 Model 011 external 8 mm tape drive
- 1/4-inch tape drives of various capacities, such as the IBM 7207 Model 012 external 1/4-inch cartridge tape drive
- 1/2-inch, 9-track tape drives, such as the IBM 9348 Model 012 tape drive

The NetTAPE Tape Library Connection product adds support for the following tape libraries when they are attached to an AIX workstation. The StorageTek products listed below are supported but not warranted by IBM.

- The IBM 3494 Tape Library Dataserver
- The IBM 3495 Tape Library Dataserver
- The StorageTek 4400 Automated Cartridge Subsystem
- The StorageTek PowderHorn 9310/9311 Automated Cartridge Subsystem
- The StorageTek WolfCreek 9360 Automated Cartridge System

Appendix E. ADSM for AIX StorageTek Tape Library Support Services Solution

This solution allows you to extend the use of automated tape operations to the local area network through ADSM and the StorageTek tape libraries. Tape libraries can be effectively shared among RISC System/6000 or SP2 servers, or with an MVS host. The provided tape drivers support StorageTek tape drives in native mode and do not emulate IBM 3490 tape drives. Support of StorageTek tape drives in native mode allows the use of mixed tape media in the same library and the accurate reporting of drive types in reporting systems.

The software includes a set of utilities and features that interact with StorageTek's ACSLS server or Library Station software on an MVS host. The utilities query the status of volumes, drives, and scratch pools, allow for mounting and dismounting of tape volumes, allow the definition of scratch pools, and the assignment of tape volumes to scratch pools. The *auto checkin* feature allows you to check in large numbers of tape volumes into the library at one time.

The solution contains the following components:

- ADSM for AIX with StorageTek Tape Library Support software
 - Enables ADSM for AIX to use StorageTek tape libraries and associated tape drives as supported storage devices
- IBM Services
 - Installation
 - Defect support and maintenance
 - Installation and configuration documentation

Highlights:

- Allows cost-effective sharing of tape libraries among RISC System/6000, SP2, and MVS servers
- Extends automated tape operations and their efficiencies to local area network environments
- Allows the use of mixed media in the same tape library
- Provides a single automated backup and restore solution to ensure integrity and security across both mainframe and local area network environments
- Protects your investment by utilizing currently installed tape libraries

- Retains the benefits of ADSM for AIX product support by utilizing the standard IBM software support structure

StorageTek Device Support:

Library	Control Unit/Drive
4410 LSM, or 9310 Powderhorn, or 9360 Wolfcreek	4781 18-track SCSI and/or 4791 36-track SCSI and/or 9490 Timberline and/or SD-3 Redwood
9710 Timberwolf	4890 Twin Peaks and/or Quantum DLT 4000

The StorageTek 9710 Timberwolf may contain Twin Peaks 4890 36-track tape drives, Quantum DLT 4000 tape drives, or both in the same library. The tape drives are supported with SCSI-2 Fast and Wide connectivity. The library control interface is through StorageTek's ACSLS or Library Station software.

Appendix F. NetTAPE Hints and Tips

This appendix contains hints and tips for using NetTAPE with the basic AIX backup commands: tar, cpio, dd, and backup/restore.

For specific information about using NetTAPE, please see the *IBM Network Tape Access and Control System for AIX* and *IBM NetTAPE Tape Library Connection User's Guide and Reference* in the `/usr/lpp/nettape/doc` directory.

The topics we cover are:

- Using tar with labeled tapes
- Using tar, cpio, dd, and backup/restore with multivolume data sets
- Using the backup and restore commands
- Using the cpio command
- Using DB2 PE database backup and restore commands

F.1 Using tar with Labeled Tapes

When issuing multiple tar commands to write multiple files to a single labeled tape, `cltopen` must be issued between each tar command.

Here is an example:

```
# allocate a 3494 tape device with dataset name d1 and file sequence 1
# standard label tape
# tar file1 to the tape
cltalloc -P 3494 -f mytape -d d1 -s 1 -l s1
tar -cvf mytape.1 file1

# now open the tape with dataset name d2 and file sequence 2
# tar file2 to the tape
cltopen -f mytape -d d2 -s 2
tar -cvf mytape file2

# now let's get the 2 files off the tape
# open the tape to dataset name d1 and file sequence 1
# restore file1 from tape
cltopen -f mytape -d d1 s 1
tar -xvf mytape

# open the tape to dataset name d2 and file sequence 2
# restore file2 from tape
```

```
cltopen -f mytape -d d2 s 2
tar -xvf mytape

# unallocate the tape device
cltunall -f mytape
```

F.2 Using tar, cpio, dd, and backup/restore with Multivolume Data Sets

Certain NetTAPE parameters are required when using multiple volumes with the tar, dd, and cpio commands. The parameters vary according to whether or not a labeled tape is used.

F.2.1 Without a Labeled Tape (NL)

To write to the tape, you must specify the SUP_EOV allocation parameter. To read from the tape, you must specify the entire volume list, as well as the SUP_EOV and SUP_EOF parameters.

Here is an example using tar:

```
# allocate a 3490 device using the suppress end of volume option
# using a scratch tape
cltalloc -P 3490B -f mytape -o "SUP_EOV"

# tar a large directory to tape which I know will span volumes
tar cvf mytape /large_directory

# find out the volume serial numbers since I didn't request
# any specific volumes and used scratch tapes
cltvols -f mytape

# open the device with the volume list and the suppress end
# of volume and suppress end of file options
cltopen -f mytape -v VOL1,VOL2,VOL3 -o "SUP_EOV SUP_EOF"

# restore the large directory
tar -xvf mytape

# unallocate the device
cltunall -f mytape
```

Although the example is for tar, you must use the same parameters for cpio, dd, and backup/restore.

F.2.2 With a Labeled Tape (AL or SL)

To write to the tape you must specify the SUP_EOV allocation parameter. To read from the tape you must specify the entire volume list, as well as the SUP_EOV parameter.

Here is an example using tar:

```
# allocate a 3490 device using the suppress end of volume option
# and a standard label scratch tape
cltalloc -P 3490B -f mytape -o "SUP_EOV" -l s1

tar cvf mytape /large_directory
cltvols -f mytape
cltopen -f mytape -v VOL1,VOL2,VOL3 -o "SUP_EOV"
tar -xvf mytape
cltunall -f mytape
```

Although the example is for tar, you must use the same parameters for cpio, dd, and backup/restore.

F.3 Using the backup and restore Commands

When backing up large amounts of data, you should specify the block size on the restore. If you think you will need multiple volumes, you must also specify the SUP_EOV parameter on the allocations.

Here is an example:

```
# allocate a 3490 device using scratch tape
cltalloc -P 3490B -f mytape

# backup everything in the current directory to tape
find . -print | backup -vif mytape

# get the volume serial
cltvols -f mytape

# unallocate the device
cltunall -f mytape

# allocate the device for restore using the volume from the
# backup
cltalloc -P 3490C -f mytape -v VOL1

# restore the data specifying blocksize of 51200
# dd is required when using restore with tape, see the
# restore manpage
dd if=mytape bs=51200 | restore -xvzf -

# unallocate the device
cltunall -f mytape
```

F.4 Using the cpio Command

You should specify the block size when using cpio to back up large files. If you think you will need multiple volumes, you must also specify the SUP_EOV parameter on the allocations.

Here is an example:

```
# allocate a 3490 device using scratch tape
cltalloc -P 3490B -f mytape

# cpio the current directory
# the blocksize is specified with C32. This will result
# in a blocksize of 16k (32 x 512).
find . -print | cpio -ovdC32 > mytape

# get the volume serial
cltvols -f mytape

# open the device specifying the volume serial
cltopen -f mytape -v VOL1

# restore the tape, again using the blocksize 32
cpio -idmvC32 < mytape

# unallocate the device
cltunall -f mytape
```

F.5 Using DB2 PE Database backup and restore Commands

DB2 PE database backup and restore commands to *mytape* (a symbolic link to the NetTAPE tape pseudo device driver created by cltalloc) only works if you specify the:

- Full path name of the NetTAPE device (for example, /u/curly/mytape)
- Buffer parameter that corresponds to the NetTAPE device's MAX_DATA size

Here is an example:

```
db2 backup database sample to /u/curly/mytape buffer 16
db2 restore database sample from /u/curly/mytape buffer 16
```

NetTAPE's MAX_DATA size defaults to 64 KB, therefore the buffer parameter is 16 (16 * 4 KB buffer size = 64 KB). If you change the MAX_DATA size (for example, through cltalloc -o), the buffer parameter must correspond to the new MAX_DATA size.

Appendix G. Scripts

This appendix contains two scripts. The first script provides automatic volume labeling using an automatic library. The second script provides automatic checkin of volumes into a library.

G.1 Automatic Labeling of a Library

This script enables you to automatically label volumes by using library devices in automatic mode (that is, an automated library). The script searches for volumes in all of the storage slots in the library and tries to label each volume that it finds. By default, each volume is returned to its original location.

- For 349x libraries the script:
 - Scans only the INSERT category
 - Does not provide prompts
 - Uses volume labels from the external barcode
- For SCSI libraries, you would normally be prompted to enter the name for each volume that is encountered in the library.

The script is especially useful for SCSI libraries because it does not require you to enter the name for each volume; instead, it gets the name from a file that contains the list of the names. In the script we use two tape drives to increase performance. If you have only one tape drive, specify only one.

```
#!/bin/ksh
#set -x
LABEL="/home/adsm/back_label_xxxx"
PASSWD=aixtape

echo
echo "This procedure will create new tape labels."
echo "Make sure that you will use the correct cartridges."
echo "The file $LABEL must be edited to reflect the labels\nyou want to create.\n"
echo "Enter .y" to begin"
read Answer
if . "$Answer" = y "
then
# Put your correct drive definitions.
cat $LABEL | dsmlabel -drive=/dev/mt0,32768 -drive=/dev/mt1,32769 -library=/dev/lb0
-overwrite -keep -search
fi
echo
```

```
#  
exit
```

Here is an example of the file referred to by the script variable *LABEL*. You put the list of labels in this file:

```
ADSM01  
ADSM02  
ADSM03  
ADSM04  
ADSM05  
ADSM06  
ADSM07
```

Note

Be careful when using the `-overwrite` option on the `dsmlabel` command. If you have a tape that for some reason has not been ejected, the tape could be relabeled, and you would never know. It might be safer to issue the command without the `-overwrite` option and then make a decision on each tape that was already labeled.

G.2 Automatic Checkin of Volumes into a Library

This script file provides automatic checkin of volumes into a library. ADSM automatically searches the library for new volumes that have not already been added to the library volume inventory. The server always attempts to read the media label for a volume in a 349x library and only searches for new volumes in the following categories:

- Private category
- Scratch category
- Insert category

```
#!/bin/ksh  
PASSWD=aixtape  
LABEL="/home/adsm/back_label_XXXX"
```

```
dsmadm -id=admin -password=$PASSWD checki libv lib_4mm stat=pri search=yes
```

```
exit
```

Appendix H. IBM Tape Drive and Library Attachment

Here are some details about attaching tape drives and libraries to a RISC System/6000. We look at internal and external tape drives and external tape libraries. Note that not all RISC System/6000 models support all of the devices we describe.

H.1 Internal Tape Drives

The kinds of internal drives supported on a RISC System/6000 depend on each RISC System/6000, but generally RISC System/6000s support QIC, 4 mm, and 8 mm internal tape drives connected through SCSI attachment.

All internal tape drives are connected through an internal SCSI cable. Therefore, the tape drives cannot be shared between different machines. For performance reasons, avoid connecting the tape drive on the same SCSI bus as your disks.

H.2 External Tape Drives and Libraries

External tape drives support three connectivity methods:

- SCSI adapter
- Parallel channel adapter
- ESCON adapter

We describe each type of attachment and look at some special considerations for 7331 and 3590-B11 tape libraries.

H.2.1 SCSI Attachment

The most common method used to attach tape drives is SCSI. The SCSI bus can be connected on the IBM Micro Channel and on the standard PCI-Bus with the newer RISC System/6000 machines. The standard SCSI PCI-bus is limited to seven SCSI devices. When a SCSI bus is shared between different hosts, the hosts are counted as device addresses. Therefore the number of devices that can be attached is reduced.

There are five standard exits with SCSI attachments, each with different characteristics and limitations:

- SCSI-1 Single Ended

The main characteristics of this type of SCSI are:

- The bandwidth of the bus is limited to 4-5 MB/sec.

- The SCSI bus length is limited to 6 m.
- Sharing of devices is not allowed between different hosts.

Generally, you can connect any QIC, 4 mm, or 8 mm tape drive with this type of SCSI connection. Some typical drives include the 7207-012, 7206-005, and 7208-011.

- SCSI-2 Single Ended

The main characteristics of this type of SCSI are:

- The bandwidth is limited to 10 MB/sec.
- The SCSI bus length is limited to 3 m.
- Sharing of devices is not allowed between different hosts.

Generally, you can connect any QIC, 4 mm, or 8 mm tape drive with this type of SCSI attachment. An example is the 7332-005 4 mm library.

- SCSI-2 Single Ended and Fast Wide

The main characteristics of this type of SCSI are:

- The bandwidth is limited to 20 MB/sec.
- The SCSI bus length is limited to 3 m.
- Sharing of devices is not allowed between different hosts.

The 7332-005 4 mm library is an example of a device that can be attached by using SCSI-2 single ended and fast wide.

- SCSI-2 Differential (typically adapter #2420)

The main characteristics of this type of SCSI are:

- The bandwidth is limited to 10 MB/sec.
- The SCSI bus length is limited to 19 m.
- Sharing of devices between different hosts is allowed.

The old SCSI-2 Differential High-Performance I/O controller #2420 is replaced by the new SCSI-2 Differential Fast Wide.

Typical devices that support attachment using SCSI-2 Differential are:

- 3490 Model C11 and C22

The 3490 Model C11 contains one drive, and the Model C22 contains two drives. The 3490-C11 is provided with a six-cartridge ACL. The 3490-22 has two six-cartridge ACLs. Because you can have one or two differential SCSI features on each model, the tape devices can be shared among multiple RISC System/6000 servers.

- 3490 Model E01 and E11

The 3490 Model E01 is a table-top version. The 3490 Model E11 is a rack-mounted version. Both models include a seven-cartridge cartridge stack loader (CSL). The 3490E Models E01 and E11 are lower-priced, smaller, full-function versions of the 3490E that are designed for the RISC System/6000.

Although multiple RISC System/6000 systems can be linked to a 3490 E01 or E11 tape drive, they cannot use the drive simultaneously. Sharing is not allowed.

- 3590 Model B11

The 3590 Model B11 includes a 10-cartridge random access ACF that can be used as a mini-library. The Model B11 requires a 7202 rack, 7015-R00 rack, or space in an existing rack.

- The 3490-B1A in a 3494 and 3495 Tape Library Dataserver

This support is replaced by a SCSI-2 Differential Fast Wide attachment.

- SCSI-2 Differential Fast Wide

The main characteristics of this type of SCSI are:

- The bandwidth is limited to 20 MB/sec.
- The SCSI bus length is limited to 25 m.

There are many advantages of this type of SCSI, such as better performance and the ability to share tape drives among different RISC System/6000s.

Note: Although multiple systems may be attached to a tape drive, the systems cannot use the drive simultaneously. The tape drive can only be varied online to one system at a time.

There are two RISC System/6000 SCSI-2 Differential Fast Wide adapters:

- The SCSI-2 Differential Fast Wide Adapter/A #2416
- The new SCSI-2 Enhanced SCSI-2 Differential Fast Wide Adapter/A #2412, which is recommended for a faster data rate.

The following tapes or tape libraries are supported on all models that support the RISC System/6000 SCSI-2 Differential Adapter:

- The 7331-205 8mm Tape Library
- The 3490 Model C11 and C22
- 3490 Model E01 and E11
- 3590 Models B11
- 3494 Tape Library Dataserver Model L2 (SCSI only)

- The 3490-B1A in a 3494 and 3495 Tape Library Dataserver
The 3490-B1A can be attached to host systems using one or two SCSI adapters. The 3590-B1A has its own integrated tape control unit. A maximum of two 3590-B1As can be in a control unit frame, and a maximum of six 3590-B1As can be in a drive unit frame.
- 3494 Tape Library Dataserver Model L10 (SCSI, parallel, and ESCON)
- The 3490-CxA in a 3494 and 3495 Tape Library Dataserver
The 3590-C1A has its own integrated tape control unit. Up to two Differential SCSI-2 adapters can be installed on the 3490E C1A or C2A. The differential SCSI-2 adapters in a 3590-CxA can coexist with ESCON or parallel channel adapters on a 3490E.

H.2.2 Parallel Channel and ESCON Attachments

By expanding the availability of channel products on RISC System/6000 systems, including the SMP Micro Channel Family and Scalable POWERparallel System computers, additional connectivity is possible between those systems and mainframe systems. Parallel and ESCON connections to a RISC System/6000 enable customers to move equipment and data from legacy systems to the RISC System/6000 in an evolutionary way.

- S/370 Parallel Channel attachment

S/370 Parallel Channel attachment supports the following devices:

- 3494 and 3495 Tape Library Dataservers
Only the 3494 Model L10 is supported for parallel, ESCON, and SCSI attachments. However, you can intermix the attachments.
- The 3490-CxA in a 3404 Model L10
One 3490-CxA in a control unit frame and one 3490-CxA in a drive unit frame are supported. Therefore, the maximum number of 3490-CxAs supported is eight because there is a maximum of eight frames.
The 3490-C1A contains one tape drive, and the 3490-C2A contains two tape drives.

Note: The IBM 3590 does not support parallel attachment.

The S/370 Parallel Channel attachment for the RISC System/6000 requires the S/370 Channel Emulator/A adapter (feature code 2759), the 3494 AIX Parallel Channel Tape Attachment/6000 (3494 feature code 5224), and the ADSTAR Programming RPQ #P83033.

The S/370 parallel channel attachment mode is rarely used because of the adapter price and the performance obtained.

Using the 3490E's Improved Data Recording Capability (IDRC) data compression, the 3490E Models E01 and E11 can provide an effective throughput, as high as 6.5 MB/sec for compressible data. In comparison, the 3490E C models (when using IDRC) support an effective throughput of up to 3.5 MB/sec with SCSI and up to 3.8 MB/sec with parallel channel attachment. The actual throughput depends on the system processor and the data block size.

- ESCON Attachment

ESCON attachment supports the following devices:

- The 3490 Model C11 and C22
- The 3494 and 3495 Tape Library Dataservers
- The 3490-CxA in a 3494 Model L10

ESCON provides connectivity over longer distances than 61 m (200 ft), achieves higher performance than parallel, and provides device sharing with ES/9000 processors that use ESCON channels. ESCON provides performance up to 9 MB/sec at distances up to 43 km.

You must order PRPQ 8A1016 to obtain the necessary software device drivers for the ESCON Channel Emulator. The drivers are no longer available as part of feature code 2759.

Note: The IBM 3590 does not have ESCON attachment.

H.2.3 7331 and 3590-B11 Tape Library Considerations

Here are some special considerations for the 7331 and 3590-B11 tape libraries.

H.2.3.1 The 7331 Tape Library

The 7331 has two configurations:

- Base library configuration is intended for attaching the 7331 to a single host, where the library is accessible to the host through one SCSI port.
- Split library configuration is intended for attaching the 7331 to two hosts, where the 7331 appears as two separate half-size libraries on two SCSI buses.

The 7331 has three modes of operation:

- Manual mode is controlled by the operator panel switches and allows complete operator control.

- Sequential mode automatically loads the cartridges into the drives sequentially for reading and writing.
- Random mode allows tapes to be selected randomly by the host for reading or writing.

Each mode will operate in either configuration. In the split library configuration, each half of the library can operate in random or sequential mode independently.

Note: The second drive feature, #0702, is a prerequisite if you want to use random mode on both sides.

H.2.3.2 The 3590-B11 Tape Library

You can use a 3590-B11 like a small tape library. You can support the 3590 ACF in random mode. The AIX device driver feature #9603, must be used.

Appendix I. Product Levels

Here are the AIX, ADSM, and NetTAPE product levels used in our environment.

I.1 AIX

- AIX Version 3.2.5.1
- Dcephreads Version 1.3

I.2 ADSM

- Server: ADSM Version 2, Release 1, test version of Level 0.7/0.7
- Client: ADSM Version 2, Release 1, Level 0.2

I.3 NetTAPE

The NetTAPE products were Version 1, Release 1, with PTF 1.

Appendix J. Special Notices

This publication is intended to help IBM, customer, vendor, and consultant personnel manage tapes in an AIX environment by using the native AIX operating system, ADSM for AIX, NetTAPE, and CLIO/S. The information in this publication is not intended as the specification of any programming interfaces that are provided by ADSM, NetTAPE, and CLIO/S. See the PUBLICATIONS section of the IBM Programming Announcement for ADSM, NetTAPE, and CLIO/S for more information about what publications are considered to be product documentation.

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 IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

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, NY 10594 USA.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM ("vendor") products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific

information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

ADSTAR	AIX
AIX/6000	DATABASE 2
DB2	ESCON
IBM	Magstar
MVS	MVS/ESA
POWERparallel	RISC System/6000
S/370	S/390
Scalable POWERparallel Systems	SP
SP1	SP2
9076 SP1	9076 SP2

The following terms are trademarks of other companies:

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.

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

Ethernet	X/Open Company Limited
SCSI	Security Control Systems, Inc.

Other trademarks are trademarks of their respective companies.

Appendix K. Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

K.1 International Technical Support Organization Publications

For ordering information on these ITSO publications see “How To Get ITSO Redbooks” on page 261.

- ADSM redbooks
 - General topics
 - *ADSM Presentation Guide*, GG24-4146
 - *ADSM Version 2 Presentation Guide*, SG24-4532
 - *ADSM Implementation Examples*, GG24-4034
 - *ADSM Advanced Implementation Experiences*, GG24-4221
 - Specific server books
 - *ADSM for MVS: Recovery and Disaster Recovery*, SG24-4537
 - *ADSM for MVS: Using Tapes and Tape Libraries*, SG24-4538
 - *Getting Started with ADSM/2*, GG24-4321
 - *ADSM for OS/2: Advanced Topics*, SG24-4740
 - *Getting Started with ADSM/6000*, GG24-4421
 - *ADSM for AIX: Advanced Topics*, SG24-4601
 - *ADSM/6000 on 9076 SP2*, GG24-4499
 - *ADSM/VSE Implementation*, GG24-4266
 - *Setting Up and Implementing ADSM/400*, GG24-4460
 - Specific client books
 - *Getting Started with the NetWare Client*, GG24-4242
 - *Hierarchical Storage Management for NetWare: Implementing ADSM and Avail HSM*, SG24-4713
 - *Getting Started with the AIX/6000 Client*, GG24-4243
 - *ADSM Hierarchical Storage Management for AIX Clients*, SG24-4631
 - *ADSM API Examples for OS/2 and Windows*, SG24-2588

- ADSM with other products
 - *Using ADSM to Back Up Databases*, SG24-4335-01
 - *Using ADSM to Back Up Lotus Notes*, SG24-4534
 - *Using ADSM to Back Up OS/2 LAN and Warp Servers*, SG24-4682
 - *Backup, Recovery, and Availability with DB2 Parallel Edition on RISC/6000 SP*, SG24-4695
- Other topics that include ADSM
 - *AIX Storage Management*, GG24-4484
 - *AIX Storage Management Products Comparison*, GG24-4495
- In press
 - *ADSM for NT*
 - *Client Disaster Recovery: Bare Metal Restore*
 - *ADSM Technical Overview*
 - *Tivoli TME 10 AIX Examples*
 - *Tivoli TME: Software Distribution and Configuration Management*
- CLIO/S redbook
 - *RISC/6000 to Mainframe Using S/370 Channel Connections*, SG24-4589
- Related ITSO Sold Bills of Form
 - *Storage Software: Distributed Storage Management*, SBOF-6311
 - *AIX Operating System/Systems Management and High Availability*, SBOF-6338
 - *RISC System/6000 Hardware*, SBOF-6340

A complete list of International Technical Support Organization redbooks, with a brief description of each, may be found in:

International Technical Support Organization Bibliography of Redbooks, GG24-3070.

K.2 Other Publications

These publications are also relevant as additional information sources.

- ADSM Publications
 - *ADSM Online Product Library CD-ROM*, SK2T-8714

All of the ADSM publications are available in online readable format on the CD-ROM listed above. The ADSM library is also available on the following CD-ROMs:

- *MVS Base Collection Kit*, SK2T-0710
- *VM Base Collection Kit*, SK2T-2067
- *OS/2 Base Collection Kit*, SK2T-2176
- *AIX Base Collection Kit*, SK2T-2066
- *AS/400 Base Collection Kit*, SK2T-2171
- *IBM SystemView for AIX*, SK2T-1451

Please note that the ADSM AIX and MVS server books and the client books listed are the Version 2 books; the library is restructured for Version 2.

- *ADSM: General Information*, GH35-0131
- *ADSM: Licensed Program Specification for AIX*, GH35-0132
- *ADSM: Licensed Program Specification for MVS*, GH26-4038
- *ADSM: Licensed Program Specification for VM*, GH35-0115
- *ADSM: Licensed Program Specification for OS/2*, GH26-4012
- *ADSM: Licensed Program Specification for ADSM V3R6 for OS/400*, GH26-4015-02
- *ADSM: Licensed Program Specification for VSE*, GH26-4026
- *ADSM: Licensed Program Specification for SUN Solaris*, GH26-4021-02
- *ADSM: Licensed Program Specification for HP-UX*, GH26-4017-02

- *ADSM: Installing the AIX Server and Administrative Client*, SH35-0136
- *ADSM: Installing the MVS Server and Administrative Client*, SH26-4043
- *ADSM: Installing the OS/2 Server and Administrative Client*, SH26-4014
- *ADSM: Installing the OS/400 Server and Administrative Client*, SH26-4016-01
- *ADSM: Installing the VSE Server and Administrative Client*, SH26-4029
- *ADSM: Installing the SUN Solaris Server and Administrative Client*, SH26-4024

- *ADSM: Installing the HP-UX Server and Administrative Client, SH26-4020*

- *ADSM: Administrator's Guide for AIX, SH35-0134*
- *ADSM: Administrator's Guide for HP-UX, SH26-4018*
- *ADSM: Administrator's Guide for MVS, SH26-4039*
- *ADSM: Administrator's Guide for OS/2, SH26-4003*
- *ADSM: Administrator's Guide for OS/400, SH26-4008-01*
- *ADSM: Administrator's Guide for SUN Solaris, SH26-4022*
- *ADSM: Administrator's Guide for VM, SH35-0117*
- *ADSM: Administrator's Guide for VSE, SH26-4027*

- *ADSM: Administrator's Reference for AIX, SH35-0135*
- *ADSM: Administrator's Reference for HP-UX, SH26-4019*
- *ADSM: Administrator's Reference for MVS, SH26-4040*
- *ADSM: Administrator's Reference for OS/2, SH26-4004*
- *ADSM: Administrator's Reference for OS/400, SH26-4009-01*
- *ADSM: Administrator's Reference for SUN Solaris, SH26-4023*
- *ADSM: Administrator's Reference for VM, SH35-0130*
- *ADSM: Administrator's Reference for VSE, SH26-4028*
- *ADSM: Messages, SH35-0133*

- *ADSM: Device Configuration, SH35-0137*
- *ADSM: Installing the Clients, SH26-4049*
- *ADSM: Using the UNIX Hierarchical Storage Management (HSM) Clients, SH26-4030*
- *ADSM: Using the UNIX Backup/Archive Clients, SH26-4052*
- *ADSM: Using the OS/2 Backup/Archive Clients, SH26-4053*
- *ADSM: Using the DOS Backup/Archive Clients, SH26-4054*
- *ADSM: Using the Microsoft Windows Backup/Archive Clients, SH26-4056*
- *ADSM: Using the Novell NetWare Backup/Archive Clients, SH26-4055*

- *ADSM: Using the Apple Macintosh Backup/Archive Clients*, SH26-4051
- *ADSM: Using the Lotus Notes Backup Agent*, SH26-4047
- *ADSM: Client Reference Cards*, SX26-6013
- NetTAPE Publications
 - *NetTAPE General Information*, GC23-3990
 - *NetTAPE and NetTAPE Tape Library Connection User's Guide and Reference*
- CLIO/S Publications
 - *CLIO/S General Information*, GC23-3879
 - *CLIO/S User's Guide and Reference*

How To Get ITSO Redbooks

This section explains how both customers and IBM employees can find out about ITSO redbooks, CD-ROMs, workshops, and residencies. A form for ordering books and CD-ROMs is also provided.

This information was current at the time of publication, but is continually subject to change. The latest information may be found at URL <http://www.redbooks.ibm.com/redbooks>.

How IBM Employees Can Get ITSO Redbooks

Employees may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **PUBORDER** — to order hardcopies in United States
- **GOPHER link to the Internet** - type GOPHER.WTSCPOK.ITSO.IBM.COM

- **Tools disks**

To get LIST3820s of redbooks, type one of the following commands:

```
TOOLS SENDTO EHONE4 TOOLS2 REDPRINT GET SG24xxxx PACKAGE
TOOLS SENDTO CANVM2 TOOLS REDPRINT GET SG24xxxx PACKAGE (Canadian users only)
```

To get lists of redbooks:

```
TOOLS SENDTO WTSCPOK TOOLS REDBOOKS GET REDBOOKS CATALOG
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET ITSOCAT TXT
TOOLS SENDTO USDIST MKTTOOLS MKTTOOLS GET LISTSERV PACKAGE
```

To register for information on workshops, residencies, and redbooks:

```
TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ITSOREGI 1996
```

For a list of product area specialists in the ITSO:

```
TOOLS SENDTO WTSCPOK TOOLS ZDISK GET ORGCARD PACKAGE
```

- **Redbooks Home Page on the World Wide Web**
<http://w3.itso.ibm.com/redbooks/redbooks.html>
- **IBM Direct Publications Catalog on the World Wide Web**
<http://www.elink.ibm1ink.ibm.com/pb1/pb1>

IBM employees may obtain LIST3820s of redbooks from this page.

- **ITSO4USA category on INEWS**
- **Online** — send orders to: USIB6FPL at IBMMAIL or DKIBMBSH at IBMMAIL
- **Internet Listserver**

With an Internet E-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an E-mail note to announce@webster.ibm1ink.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank). A category form and detailed instructions will be sent to you.

How Customers Can Get ITSO Redbooks

Customers may request ITSO deliverables (redbooks, BookManager BOOKs, and CD-ROMs) and information about redbooks, workshops, and residencies in the following ways:

- **Online Orders** (Do not send credit card information over the Internet) — send orders to:

	IBMMAIL	Internet
In United States:	usib6fpl at ibmmail	usib6fpl@ibmmail.com
In Canada:	caibmbkz at ibmmail	lmannix@vnet.ibm.com
Outside North America:	bookshop at dkibmbsh at ibmmail	bookshop@dk.ibm.com

- **Telephone orders**

United States (toll free)	1-800-879-2755
Canada (toll free)	1-800-IBM-4YOU
Outside North America	(long distance charges apply)
(+45) 4810-1320 - Danish	(+45) 4810-1020 - German
(+45) 4810-1420 - Dutch	(+45) 4810-1620 - Italian
(+45) 4810-1540 - English	(+45) 4810-1270 - Norwegian
(+45) 4810-1670 - Finnish	(+45) 4810-1120 - Spanish
(+45) 4810-1220 - French	(+45) 4810-1170 - Swedish

- **Mail Orders** — send orders to:

IBM Publications Publications Customer Support P.O. Box 29554 Raleigh, NC 27626-0570 USA	IBM Publications 144-4th Avenue, S.W. Calgary, Alberta T2P 3N5 Canada	IBM Direct Services Sortemosevej 21 DK-3450 Allerød Denmark
--	--	--

- **Fax** — send orders to:

United States (toll free)	1-800-445-9269
Canada (toll free)	1-800-267-4455
Outside North America	(+45) 48 14 2207 (long distance charge)

- **1-800-IBM-4FAX (United States) or (+1) 415 855 43 29 (Outside USA)** — ask for:

Index # 4421 Abstracts of new redbooks
Index # 4422 IBM redbooks
Index # 4420 Redbooks for last six months

- **Direct Services** - send note to softwareshop@vnet.ibm.com

- **On the World Wide Web**

Redbooks Home Page	http://www.redbooks.ibm.com/redbooks
IBM Direct Publications Catalog	http://www.elink.ibm.com/pbl/pbl

- **Internet Listserver**

With an Internet E-mail address, anyone can subscribe to an IBM Announcement Listserver. To initiate the service, send an E-mail note to announce@webster.ibm.com with the keyword subscribe in the body of the note (leave the subject line blank).

List of Abbreviations

ACF	automatic cartridge facility	GUI	graphical user interface
ACL	automatic cartridge loader	HSM	hierarchical storage management
ACSLs	Automated Cartridge System Library Server	IBM	International Business Machines Corporation
AFS	Andrew File System	ITSO	International Technical Support Organization
APA	all points addressable	NFS	network field support
ATLDS	Automated Tape Library Dataserver	ODM	object data manager
ATM	asynchronous transfer mode	PROFS	Professional Office System
AVR	automatic volume recognition	QIC	quarter-inch cartridge
BLP	bypass label processing	RPC	remote procedure call
DDS	digital data system	SCSI	small computer system interface
DLT	digital linear tape	SMIT	System Management Interface Tool
DMA	direct memory access	SMP	symmetric multiprocessing
DRM	Disaster Recovery Manager	SSA	serial storage architecture
FDDI	fiber distributed data interface	SSI	storage server interface
FIFO	first-in first-out	STK	StorageTek Corporation

Index

Special Characters

- .netrc
 - CLIO/S configuration file 191
 - NetTAPE configuration file 111
 - sample 115

Numerics

- 3480
 - Atape device driver for SCSI attachment 24
 - mtdd device driver for ESCON and parallel-channel attachment 24
 - S370dd low-level device driver for parallel-channel attachment 24
 - S390dd low-level device driver for ESCON attachment 24
- 3490
 - Atape device driver for SCSI attachment 24
 - mtdd device driver for ESCON and parallel-channel attachment 24
 - S370dd low-level device driver for parallel-channel attachment 24
 - S390dd low-level device driver for ESCON attachment 24
- 3494
 - atlld device driver for use with SCSI-attached drives 24
 - lmcpx device driver 33
 - tape library with local intelligence and database 32
- 3495
 - lmcpx device driver 33
 - tape library with local intelligence and database 32
- 3590
 - Atape device driver for SCSI attachment 24
- 3590-B11
 - smcx device driver 33
- 4 mm
 - autochangers
 - ADSM support 209
 - tape drives
 - ADSM support 209

- 4 mm (*continued*)
 - tape drives (*continued*)
 - NetTAPE support 235
- 7331
 - Atape device driver 24
 - small 8 mm tape library 32
 - smcx device driver 33
- 7331-205
 - tape library with random access to slots 32
 - with a barcode reader 32
- 7332
 - small 4 mm tape library 32
- 8 mm
 - autochangers
 - ADSM support 209
 - tape drives
 - ADSM support 209
 - NetTAPE support 235

A

- access rights
 - tape volume control 6
- accessor
 - locating cartridges in a slot 31
 - on a tape library 31
- ADSM
 - 3494 tape library support 47
 - 3495 tape library support 47
 - 3590-B11 support 46
 - adding a new tape drive to your ADSM server 47
 - adding a new tape library to your ADSM server 47
 - automated library full situation 55
 - backing up the ADSM server 69
 - barcode support 56
 - checking in volumes to an automated library 54
 - cleaning tape use 60
 - collocation 67
 - commands
 - audit library 56
 - audit volume 58

ADSM (continued)

commands (continued)

- checkin libvolume 54
- checkout libvolume 55
- define volume 50
- delete drive 76
- dsmlabel 51
- move data 57, 58
- move drmedia 72
- query content 58
- query drmedia 71
- query volhist 49
- query volume 49
- controlling how ADSM batches data between storage media
 - movebatchsize 80
 - movesizethresh 80
- copy groups 44
- damaged storage pool volume recovery 59
- device classes 44
- device definitions 47
- device drivers
 - adsm_mt_dd 47
 - end-of-volume handling 47
 - for most SCSI-attached devices 25
 - mtx 25
 - support for label processing 47
- device sharing
 - library sharing or partitioning 77
 - sharing with other applications on the same system 76
 - sharing with other systems 77
 - using the External Library Manager 76
 - without the External Library Manager 76
- device support
 - 4 mm autochangers 209
 - 4 mm tape drives 209
 - 8 mm autochangers 209
 - 8 mm tape drives 209
 - device support modules 209
 - DLT tape drives 209
 - DLT tape libraries 209
 - ESCON-attached tape library
 - dataservers 209
 - ESCON-attached tape transports 209
 - native SCSI rewritable optical 209
 - optical drives 209
 - optical jukeboxes 209

ADSM (continued)

device support (continued)

- parallel-attached tape library
 - dataservers 209
- parallel-attached tape transports 209
- QIC tape drives 209
- SCSI-attached tape library
 - dataservers 209
- SCSI-attached tape transports 209
- StorageTek 237
- WORM optical 209
- direct client backup to tape 62
- disaster recovery 69
- documentation 256
- domains 43
- DRM enhancement 175
- DRM feature 69
- export and import 74
- external library 44
 - additional device support 155
 - ADSM drive sharing with other applications 154
 - configuring for use with the NetTAPE external manager 158
 - drive sharing with NetTAPE on two hosts 156
 - drive sharing with NetTAPE with an SP2 157
 - sample NetTAPE configurations 155
- functions 14
- grouping of users 43
- labeling volumes
 - automatic mode 51
 - do-it-yourself labeling 53
 - dsmlabel utility 51
 - manual mode 51
 - search option 51
 - using NetTAPE 166
- levels of priority for tape tasks 75
- library element 44
- management class
 - defintion 44
 - for directories 80
- managing offsite volumes 61
- managing volumes in automated libraries 53
- manual tape device support 46

ADSM (*continued*)

- mounting
 - scratch tapes 50
 - specific tapes 50
- moving cartridges to and from a remote location 72
- NetTAPE support
 - additional device support such as STK libraries 155
 - ADSM drive sharing with other applications 154
 - ADSM external library 153
 - advantages of external libraries 154
 - assigning ADSM a remote tape drive 171
 - auditing a library volume inventory 176
 - cladsmd 154
 - configuring the ADSM external library 158
 - defining the ADSM server to NetTAPE 160
 - defining the external library and external manager 160
 - drive sharing on two hosts 156
 - drive sharing with an SP2 157
 - ejecting volumes from an automated library 174
 - External Library Manager 101
 - handling deadlock conditions 169
 - inserting volumes into an automated library 172
 - labeling ADSM tapes with NetTAPE 166
 - managing an ADSM NetTAPE environment 166
 - managing library volumes for ADSM 172
 - migrating an existing ADSM server to use an external library 161
 - mount requests in an ADSM NetTAPE environment 167
 - NetTAPE external manager 153
 - restoring the ADSM database 170
 - returning volumes to scratch 176
 - running the ADSM server under its own user ID 158
 - sample configurations 155
 - sending a mount request to external manager 153
 - starting NetTAPE or ADSM first 166
 - stopping ADSM while a volume is mounted 172

ADSM (*continued*)

- nodes 43
- objectives 14
- offsite storage of tape volumes 71
- policy sets 43
- preparing volumes 51
- private tapes 50
- product level used for testing 251
- publications 256
- querying
 - database backup volumes 49
 - export volumes 49
 - storage pool volumes 49
- reclamation of offsite copy volumes 72
- redbooks 255
- retiring and replacing volumes 57
- scanning external labels 56
- scratch tapes 50
- small SCSI tape library support 46
- space management 44
- STK device support 46
- storage pool
 - definition 44
 - migration 63
- structure 41
- tape benefits
 - automated library support 15
 - disaster recovery 15
 - improved data recovery 15
 - improved tape drive utilization with disk caching 15
 - improved tape volume utilization 14
 - management of tape volume inventory 15
 - offsite rotation strategy 15
 - tape automation 15
- tape device support 46
- tape drive
 - adding to ADSM server 47
 - preemption 75
- tape library
 - adding to ADSM server 47
- tape management 14, 49
- tape operation 62
- tape performance
 - improving client performance 78
 - minimizing drive occupancy 78
 - reducing restore times 78

- ADSM (*continued*)
 - tape reclamation 64
 - for a single tape drive 65
 - single drive process 221
 - types of tape volumes
 - copy storage pool volumes 49
 - database backup volumes 49
 - export and import volumes 49
 - primary storage pool volumes 49
 - private 50
 - scratch 50
 - using cleaning tapes 60
 - using with NetTAPE
 - ADSM external library 17
 - drive sharing 17
 - NetTAPE ADSM external manager 17
 - STK Silo support 17
 - volumes
 - automated library full situation 55
 - checking into an automated library 54
 - checking out from an automated library 55
- adsm_mt_dd
 - ADSM device driver 47
- AIX
 - product level used for testing 251
- AIX device support
 - addressing tape devices 13
 - defining tape devices 13
 - lack of tape management
 - access control 13
 - cartridge protection 13
 - device allocation 13
 - operator mount interface 13
- AIX tape management
 - character special files 19
 - commands for administrators
 - bffcreate 23
 - installp 23
 - mksysb 23
 - commands for end users
 - backup 23
 - cpio 23
 - dd 23
 - mt 23
 - restore 23
 - rmt 23
 - tar 23
- AIX tape management (*continued*)
 - commands for end users (*continued*)
 - tcopy 23
 - tctl 23
 - commands for tape device handling
 - cfgmgr 22
 - chdev 22
 - lsattr 22
 - lsdev 22
 - mkdev 22
 - rmdev 22
 - tapechk 22
 - device drivers
 - ADSM 25, 47
 - adsm_mt_dd 47
 - block device interface 20
 - character device interface 20
 - control and management of specific devices 20
 - device files in /dev directory 20
 - end-of-volume handling 47
 - enhanced 24
 - for libraries 33
 - hardware 24
 - kernel extensions 20
 - ost 20
 - other SCSI tape (see ost) 20
 - SSD 24
 - standard 20
 - support for label processing 47
 - device sharing
 - two hosts accessing one tape device by sharing a SCSI bus 35
 - two hosts accessing one twin-tailed tape device 36
 - two twin-tailed devices between two hosts 36
 - device special files
 - overview 21
 - rmt*.x 19
 - functions used on tape devices
 - density settings 21
 - retention on close 21
 - rewind on close 21
 - trailer label 21
 - unload on close 21
 - library sharing and partitioning
 - dynamic switching 37

- AIX tape management (*continued*)
 - library sharing and partitioning (*continued*)
 - manual switching 37
 - static partitioning 37
 - reading from files on tape 21
 - requirements 38
 - rmt*.x 21
 - tape block size default 21
 - tape libraries
 - definition 31
 - types of 32
 - tape programming interface
 - ioctl commands 24
 - tape technology
 - device connections and protocols 27, 29, 30, 31
 - recording techniques 27
 - trends 38
 - writing to files on tape 21
- allocating tape drives 6
 - in NetTAPE 96
- API
 - for a NetTAPE end user 100
- applications
 - types of tape
 - archive 7
 - backup 7
 - batch 7
 - data acquisition 8
 - data interchange 7
 - data mining 8
 - digital libraries 8
 - hierarchical storage management 7
 - moving tapes to another location 7
 - space management 7
- archive
 - type of tape application 7
- Atape device driver
 - for SCSI-attached 3490s 24
 - for SCSI-attached 3590s 24
- atlld device driver
 - for 3494 when used with SCSI drives 24
- attaching devices
 - channel tape devices 31
 - parallel-attached tape library dataservers with ADSM 209
 - parallel-attached tape transports with ADSM 209

- attaching devices (*continued*)
 - RISC System/6000 245
 - using NetTAPE 16
- audit library ADSM command 56
- audit volume ADSM command 58
- automated library
 - labeling script 243
 - tape automation 15
 - tape usage considerations 10
 - using ADSM 15
 - volume checkin script 244
- automatic volume recognition
 - in NetTAPE 86

B

- backup
 - type of tape application 7
- backup and restore (AIX)
 - command 23
 - using with NetTAPE 241
 - using with NetTAPE and multivolume data sets 240
- barcode support
 - using ADSM 56
- batch
 - type of tape application 7
- batch pipes (MVS)
 - using with CLIO/S 178
- bffcreate AIX command 23
- block device interface
 - class of AIX device driver 20
 - for random-access devices 20
- bypass label processing
 - in NetTAPE 99

C

- capacity
 - of tape media 9
- centralization of tape resources
 - operations when using NetTAPE 16
 - system and application software requirements
 - allocation of tape drives 6
 - centralized mount interface for operators 6
 - remote access to tape drives or tape data 6

centralization of tape resources (*continued*)

- system and application software requirements (*continued*)
 - tape volume control 6
- centralized tape operations
 - using NetTAPE 16
- cfgmgr AIX command 22
- character device interface
 - class of AIX device driver 20
- character special files 19
- chdev AIX command 22
- checkin libvolume ADSM command 54
- checkout libvolume ADSM command 55
- cladsmd NetTAPE command 154
- cleaning
 - tape drives 9
 - tape media 9
- cleaning tape use with ADSM
- clftp CLIO/S command 178, 180
- client
 - NetTAPE component 89
- Client Input Output/Services (see CLIO/S) 17
- CLIO/S
 - accessing a tape drive on MVS or AIX 178
 - accessing tapes with AIX commands 203
 - adding a remote tape 198
 - AIX named pipes and MVS batch pipes 178
 - allocating a tape device 205
 - changing environment variable defaults 195
 - changing the tape pseudo device driver 200
 - coexistence with NetTAPE
 - accessing MVS tape data directly from AIX 86
 - commands
 - clftp 178, 180
 - clplink 181
 - cltalloc 205
 - cltclntd 202
 - cltterm 200, 203
 - cltunall 207
 - communication manager interface 181
 - configuration files
 - .netrc 191
 - parconf 191
 - configuring the tape mount daemon 196
 - distributed processing interface 181
 - ensuring password security 195

CLIO/S (*continued*)

- fast data transfer 180
- file transfer 180
- interface for distributed processing
 - control 181
- interface for high-speed communication 181
- monitoring tape mount daemon activity 199
- moving data between your MVS system and workstation 177
- obtaining tape driver transparency 178
- parsrv server 192
- pipng data to and from remote locations 181
- programs for starting
 - fcfconf.c 192
 - parsrv 192
- publications 259
- reconfiguring the tape mount daemon 201
- redbook 256
- removing the tape pseudo device driver 202
- restarting the tape mount daemon 201
- sample configuration 189
- selecting a port number 193
- setting environment variables 194
- sockets interface 181
- starting servers on other workstations and mainframes in your network 178
- starting the parsrv server
- starting the server 192
- starting the tape mount daemon 196, 202
- stopping the parsrv server
- stopping the tape mount daemon 200, 203
- supported hardware and software
 - mainframe hardware 186
 - mainframe software 186
 - RISC System/6000 hardware 186
 - RISC System/6000 software 187
- tape server 180
- tapesrv
 - making available 195
 - tape mount daemon 195
 - tape pseudo device driver 195
- transferring very large files 177
- transparent sharing of tape data
 - between AIX systems 17
 - between MVS and multiple AIX systems 17

CLIO/S (*continued*)

- typical environments 181
 - accessing AIX tapes from AIX or MVS with the tape server 182
 - accessing AIX tapes from AIX with the tape pseudo device driver 184
 - accessing MVS tapes from AIX with the tape pseudo device driver 184
 - accessing MVS tapes from AIX with the tape server 182
- unallocating a tape device 207
- using familiar workstation commands to transfer files 178
- clplink CLIO/S command 181
- cltalloc CLIO/S command 205
- cltalloc NetTAPE command 138
- cltclntd CLIO/S command 202
- cltclntd daemon
 - in NetTAPE 98, 109
- cltopen NetTAPE command 138, 142
- cltopr daemon
 - in NetTAPE 109
- cltopr NetTAPE command 93, 127
- cltorg daemon
 - in NetTAPE 109
- cltterm CLIO/S command 200, 203
- cltunall
 - CLIO/S command 207
 - NetTAPE command 142
- cltvols NetTAPE command 143
- CMI
 - CLIO/S interface for high-speed communication 181
- collocation
 - with ADSM 67
- command line interface
 - for a NetTAPE end user 100
- commands
 - ADSM
 - audit library 56
 - audit volume 58
 - checkin libvolume 54
 - checkout libvolume 55
 - define volume 50
 - delete drive 76
 - dsmlabel 51
 - move data 57, 58
 - move drmedia 72

commands (*continued*)

ADSM (*continued*)

- query content 58
- query drmedia 71
- query volhist 49
- query volume 49

AIX administrative for working with tapes

- bffcreate AIX command 23
- installp AIX command 23
- mksysb AIX command 23

AIX end user for working with tapes

- backup 23
- cpio 23
- dd 23
- mt 23
- restore 23
- rmt 23
- tar 23
- tcopy 23
- tctl 23

AIX tape device handling

- cfgmgr 22
- chdev 22
- lsattr 22
- lsdev 22
- mkdev 22
- rmdev 22
- tapechk 22

CLIO/S

- clftp 178, 180
- clplink 181
- cltalloc 205
- cltclntd 202
- cltterm 200, 203
- cltunall 207

NetTAPE

- cladsm 154
- cltalloc 138
- cltopen 138, 142
- cltopr 93, 127
- cltunall 142
- cltvols 143
- xclopr 128

communications manager interface (see CMI) 181

- compression
 - effect on tape capacity 9
 - hardware 9
- configuration files
 - CLIO/S
 - .netrc 191
 - parconf 191
 - NetTAPE
 - .netrc 111
 - .netrc sample 115
 - nettape.adm.config 111
 - nettape.adm.config sample 116
 - nettape.adm.config sample for ACF 3590s 119
 - nettape.adm.config sample for ACL 3480s 119
 - nettape.adm.config sample for ACL 3490s 119
 - nettape.adm.config sample for manual tape drives 117
 - nettape.adm.config sample for shared drives between two host systems 120
 - nettape.config 111
 - nettape.config sample 115
 - parconf 111
 - parconf sample 114
- configuring devices
 - using NetTAPE 17
- connectivity
 - alternate access to tape devices 11
 - of tape devices 11
 - sharing tape devices 11
- copy group
 - ADSM 44
- cpio
 - AIX command 23
 - using with NetTAPE 242
 - using with NetTAPE and multivolume data sets 240

D

- daemons
 - NetTAPE
 - cltclntd 98, 109
 - cltopr 109
 - cltorg 109
 - master 109

- daemons (*continued*)
 - NetTAPE (*continued*)
 - nettaped 109
 - operator 109
 - physical volume library 109
 - status 109
 - tape client 109
 - tape server 109
 - tapesrv 98, 109
 - data acquisition
 - type of tape application 8
 - data interchange
 - type of tape application 7
 - data mining
 - type of tape application 8
 - data recovery
 - using ADSM 15
 - Database 2 Parallel Edition (see DB2 PE) 242
 - database backup
 - ADSM volumes 49
 - DB2 PE
 - backup and restore commands
 - using with NetTAPE 242
 - dd
 - AIX command 23
 - using with NetTAPE and multivolume data sets 240
 - decr_pw 195
 - define volume ADSM command 50
 - delete drive ADSM command 76
 - density
 - of tape recording 9
 - device allocation
 - using NetTAPE 16
 - device class
 - ADSM 44
 - device domains
 - creating using NetTAPE 17
 - device drivers
 - ADSM
 - mtx 25
 - SCSI-attached devices 25
 - adsm_mt_dd 47
 - AIX
 - block device interface 20
 - character device interface 20
 - control and management of specific devcies 20

- device drivers (*continued*)
 - AIX (*continued*)
 - device files in /dev directory 20
 - kernel extensions 20
 - ost 20
 - other SCSI tape (see ost) 20
 - standard 20
 - end-of-volume handling 47
 - enhanced
 - additional ioctls 24
 - Atape 24
 - atIdd 24
 - for handling 3x9x devices 24
 - for libraries 33
 - mtdd 24
 - S370dd 24
 - S390dd 24
 - hardware 24
 - library
 - lbx 33
 - lmcpx 33
 - mtlib command interface 33
 - smcx 33
 - SSD 24
 - support for label processing 47
- device pools
 - creating using NetTAPE 17
 - esoteric 95
 - first level 95
 - in NetTAPE 84
 - second level 95
 - third level 95
- device sharing
 - ADSM
 - library sharing or partitioning 77
 - sharing with other applications on the same system 76
 - sharing with other systems 77
 - using the External Library Manager 76
 - without the External Library Manager 76
 - two hosts accessing one tape device by sharing a SCSI bus 35
 - two hosts accessing one twin-tailed tape device 36
 - two twin-tailed devices between two hosts 36
- device special files
 - rmt*.x 19
- device support
 - ADSM 209
 - StorageTek 237
 - AIX 13
 - NetTAPE 235
- device support modules
 - in ADSM 209
- devices
 - accessing using NetTAPE 17
 - accessing with NetTAPE 16, 17
 - attaching using NetTAPE 16
 - configuring using NetTAPE 17
 - managing using NetTAPE 17
- differential SCSI bus 29
- digital libraries
 - type of tape application 8
- disaster recovery
 - of ADSM
 - using the DRM feature 15
 - disaster recovery manager (see DRM) 15
- disk caching
 - using with ADSM to improve tape drive utilization 15
- distributed processing interface (see DPI) 181
- DLT
 - longitudinal recording technology 28
 - tape drives
 - ADSM support 209
 - tape libraries
 - ADSM support 209
- documentation
 - ADSM product library 256
 - CLIO/S 259
 - NetTAPE 259
 - redbooks 255
- domains
 - in ADSM 43
 - in NetTAPE 84
- DPI
 - interface for distributed processing control 181
- drive sharing
 - with ADSM and NetTAPE 17

DRM

- ADSM feature description 69
- new enhancements 175
- using for ADSM disaster recovery 15
- using for offsite tape rotation strategy 15
- using for storing client disaster recovery information 15
- using to create a disaster recovery plan 15

dsmlabel

- ADSM tape volume labeling 51
- automatic mode 51
- manual mode 51
- search option 51

E

- encr_pw 195
- end-user interface
 - in NetTAPE 85
- enhanced AIX device drivers
 - additional ioctls provided
 - display messages on drive panel 24
 - get device sense data 24
 - query and change device parameters 24
 - query and set block size 24
 - query and set tape position 24
 - reserve and release a device 24
 - set and locate block ID 24

ESCON

- attaching tape library dataservers with ADSM 209
- attaching tape transports with ADSM 209
- attachment to a RISC System/6000 245
- tape device connections 30

esoteric device pools

- in NetTAPE 95

exchanging tape volumes

- using NetTAPE 16

export

- ADSM volumes 49
- using for ADSM 74

External Library Manager

- ADSM interface to NetTAPE 101

F

- find_pw 195

H

- hardware device drivers 24
- helical scan
 - for 8 mm and 4 mm devices 27
 - for the video market 27
 - recording technology 27
 - start/stop performance 27
 - tape and head wear 27
- hierarchical storage management
 - type of tape application 7

I

- import
 - ADSM volumes 49
 - using for ADSM 74
- installp AIX command 23
- inventory
 - tape volume control 6
- ioctl AIX commands 24

L

- lbx
 - device driver for ADSM 7331 library 33
 - device driver for other SCSI libraries 33
- libraries
 - ADSM 44
 - external library 44
- library server
 - NetTAPE component 89
- library sharing and partitioning
 - dynamic switching 37
 - manual switching 37
 - static partitioning 37
- lmcpx
 - device driver for 3494 and 3495 33
- longitudinal recording
 - for the computer industry 28
 - maintenance 28
 - recording on multiple paths 28
 - reliability 28

- longitudinal recording (*continued*)
 - serpentine track interleaving 28
 - start/stop performance 28
 - streaming performance 28
- lsattr AIX command 22
- lsdev AIX command 22

M

- management class
 - ADSM 44
- managing
 - devices using NetTAPE 17
 - offsite volumes using ADSM 61
 - tape volume inventory using ADSM 15
 - volumes in automated libraries using ADSM 53
- manual tape devices
 - tape usage considerations 10
- master daemon
 - in NetTAPE 109
- media movement between storage locations
 - tracking the movement 11
 - using the ADSM DRM feature 11
- media type and reliability 9
- mkdev AIX command 22
- mksysb AIX command 23
- mount interface for operators 6
- mount queues
 - in NetTAPE 97
- move data ADSM command 57, 58
- move drmedia ADSM command 72
- moving tapes to another location
 - type of tape application 7
- mt AIX command 23
- mtdd device driver
 - for ESCON and parallel-channel-attached 3480s 24
 - for ESCON and parallel-channel-attached 3490s 24
- mtlib command
 - for tape libraries 33
- mtx
 - ADSM device drivers
 - for most SCSI-attached devices 25
- MVS batch pipes
 - using with CLIO/S 178

N

- named pipes (AIX)
 - using with CLIO/S 178
- NetTAPE
 - accessing devices 17
 - administrator operations
 - starting and stopping the NetTAPE status daemon 127
 - starting the NetTAPE GUI 128
 - ADSM support
 - additional device support such as STK libraries 155
 - ADSM drive sharing with other applications 154
 - ADSM external library 17, 153
 - ADSM tape drive support 16
 - advantages of external libraries 154
 - assigning ADSM a remote tape drive 171
 - auditing a library volume inventory 176
 - cladsmd 154
 - configuring the ADSM external library for use with the NetTAPE external manager 158
 - defining the ADSM server to NetTAPE 160
 - defining the external library and external manager 160
 - drive sharing on two hosts 156
 - drive sharing with an SP2 157
 - ejecting volumes from an automated library 174
 - enabling drive sharing between ADSM servers 17
 - External Library Manager 101
 - handling deadlock conditions 169
 - inserting volumes into an automated library 172
 - labeling ADSM tapes with NetTAPE 166
 - managing an ADSM NetTAPE environment 166
 - managing library volumes in an ADSM NetTAPE environment 172
 - migrating an existing ADSM server to use an external library 161
 - mount requests in an ADSM NetTAPE environment 167
 - NetTAPE ADSM external manager 17

NetTAPE (continued)

ADSM support (continued)

- NetTAPE external manager 153
- restoring the ADSM database 170
- returning volumes to scratch 176
- running the ADSM server under its own user ID 158
- sample configurations 155
- starting NetTAPE or ADSM first 166
- STK Silo support 17
- stopping ADSM while a volume is mounted 172

allocating a tape device to a user 96

API

- using 150

attaching devices 16

automatic volume recognition

- detecting when a volume is mounted 86
- verifying the volume labels 86

backup and restore commands 241

central operation

- answering mount and dismount messages for manual drives 93
- command line interface 84
- domains 84
- GUI 84
- managing mount queues 93
- managing tape drives from one or more operator stations 16
- monitoring NetTAPE operations 93
- moving mount requests 93
- starting the command line operator using cltopr 93
- updating NetTAPE device configuration 93

client user IDs 113

code packages

- clnettape 108
- nettape 108
- nettape_tlc 108

coexistence with CLIO/S

- accessing MVS tape data directly from AIX 86

commands

- cladsm 154
- cltalloc 138
- cltopen 138, 142
- cltopr 93, 127

NetTAPE (continued)

commands (continued)

- cltunall 142
- cltvols 143
- xcltopr 128

components

- client 89, 108
- library server 89, 108
- network 89
- operator 89, 108
- status daemon 89, 108
- tape server 89, 108

configuration

- of 3494 tape libraries 120
- of 3495 tape libraries 120
- of STK tape libraries 120

configuration files

- .netrc 111
- .netrc sample 115
- nettape.adm.config 111
- nettape.adm.config sample 116
- nettape.adm.config sample for ACF 3590s 119
- nettape.adm.config sample for ACL 3480s 119
- nettape.adm.config sample for ACL 3490s 119
- nettape.adm.config sample for manual tape drives 117
- nettape.adm.config sample for shared drives between two host systems 120
- nettape.config 111
- nettape.config sample 115
- parconf 111
- parconf sample 114

configuring devices 17

cpio command 242

daemons

- cltclntd 98, 109
- cltopr 109
- cltorg 109
- master 109
- nettaped 109
- operator 109
- physical volume library 109
- status 109
- tape client 109
- tape server 109

NetTAPE (continued)

- daemons (continued)
 - tapesrv 98, 109
- device allocation
 - enabling of tape drive sharing 16
 - selecting and allocating devices 16
- device domains 17
- device pools 17
 - esoteric 95
 - first level 95
 - generic names 84
 - hiding physical configuration of tape devices from users 84
 - second level 95
 - third level 95
- device states
 - down 135
 - reserve 135
 - up 135
- device support
 - 3494 and 3495 tape libraries 235
 - 4 mm tape drives 235
 - 8 mm tape drives 235
 - 9-track tape drives 235
 - half-inch tape drives 235
 - QIC tape drives 235
 - StorageTek tape libraries 235
- devices 17
- end-user and application use
 - ACF 3590 tape libraries 150
 - ACL 3480 and 3490 tape libraries 150
 - allocating a tape device 138
 - ctalloc command 138
 - ctalloc default parameters 139
 - ctopen command 138, 142
 - ctunall command 142
 - cltvols command 143
 - determining available device pools for allocation 137
 - handling end-of-file conditions 145
 - handling end-of-volume conditions 145
 - multivolume data sets 145
 - obtaining information about tape volumes used 143
 - opening a tape file 142
 - sending a tape mount request 138
 - setting an application block size 149
 - tape labels on 3480, 3490, 3590 147

NetTAPE (continued)

- end-user and application use (continued)
 - unallocating a tape device 142
 - using a previously allocated drive for additional I/O 142
 - volume information file 144
 - writing special applications using the NetTAPE API 150
- end-user interfaces
 - API 85, 100
 - command line 85, 100
 - using to allocate a drive 85
 - using to close a file 85
 - using to open a file 85
 - using to unallocate a drive 85
- end-user sample scenario 91
- error handling
 - resetting a tape device 136
 - stopping the NetTAPE client daemon 136
- exchange of tape volumes
 - between AIX and mainframe applications 16
 - tape label support 16
- grouping of devices 17
- hints and tips 239
- installation
 - clnettape 107
 - common CLIO/S NetTAPE code, clnettape 107
 - GUI operator code, nettape 107
 - nettape 107
 - nettape_tlc 107
 - server code, nettape 107
 - Tape Library Connection code, nettape_tlc 107
- library support 97
- managing devices 17
- mount queues 97
- multivolume data sets
 - backup and restore 240
 - cpio 240
 - dd 240
 - tar 240
- network support 102
- objectives
 - central tape operation 83
 - centralized tape operations management 16

NetTAPE (continued)

- objectives (continued)
 - exchanging tape volumes 16
 - high-end library support 16
 - logical pools of tape units 83
 - processing of labeled tapes on AIX 83
 - remote tape access 83
 - stand-alone drive support for low-end environments 16
 - support of automated tape libraries 83
 - tape drive sharing among applications 16
 - transparent application access to tape devices 16
 - transparent end-user access to tape devices 16
- occasional operator tasks
 - adding and deleting devices 134
 - changing the state of a device 135
 - initiating a temporary 3590 ACF scan 136
- operator authority
 - administrative 106
 - operator 106
 - view operator 106
- planning
 - configuring your NetTAPE server hosts 107
 - defining operator domains 106
 - defining operators and operator authorities 106
 - determining your NetTAPE clients 107
 - grouping the device pools into esoteric device pools 106
 - grouping the NetTAPE-managed devices into device pools 105
 - queuing mount requests 106
 - reviewing your tape drive and library configuration 105
 - selecting a default device pool 106
 - selecting the tape devices to be included in the NetTAPE configuration 105
- product level used for testing 251
- pseudo device driver 98
- publications 259
- record formats 85
 - fixed length 100
 - fixed length blocked 100
 - unformatted 100
 - variable 100

NetTAPE (continued)

- remote tape access 98
 - application access to drives connected to a remote system 16
- sample configuration files 113
- selecting a tape device for a user request 96
- server user IDs 108
- support for ADSM
 - External Library Manager 86
- tape device
 - allocation 84
 - support 102
- tape device allocation 96
- tape drive states
 - down 84
 - in use 84
 - pending reserve (PenRs) 84
 - reserved 84
 - up 84
- tape labels
 - ANSI 99
 - bypass label processing 99
 - EBCDIC standard labels 99
 - expiration date and retention period 99
 - formats for MVS data interchange 85
 - generation numbers 99
 - multiple files on a volume 99
 - multivolume tape files for AIX 85
 - unlabeled 99
- Tape Library Connection 97
- tape library support 104
- tape volume information
 - recording basic volume information 86
 - vfile 86
- tar with labeled tapes 239
- typical configuration 90
- typical operator tasks
 - displaying current allocation status 133
 - displaying outstanding operator messages 130
 - managing device pool mount queues 129
 - replying to an operator message 131
- unsupported tape devices 103
- updating the NetTAPE code 113
- using DB2 PE database backup and restore commands 242

NetTAPE (continued)

- vfile
 - for storing information about a tape volume 101
 - when to use 88
 - workstation hardware support 102
 - workstation software support 102
- NetTAPE Tape Library Connection
 - automated library support 83
- nettape.adm.config
 - general sample 116
 - NetTAPE configuration file 111
 - sample for ACF 3590s 119
 - sample for ACL 3480s 119
 - sample for ACL 3490s 119
 - sample for manual tape drives 117
 - sample for shared drives between two host systems 120
- nettape.config
 - NetTAPE configuration file 111
 - sample 115
- nettaped daemon
 - in NetTAPE 109
- Network Tape Access and Control System (see NetTAPE) 83
- node
 - ADSM 43

O

- offsite storage of tape volumes
 - using ADSM 71
- offsite tape rotation strategy
 - using the ADSM DRM feature 15
- operator
 - daemon in NetTAPE 109
 - NetTAPE component 89
- optical
 - drives
 - ADSM support 209
 - jukeboxes
 - ADSM support 209
 - native SCSI rewritable
 - ADSM support 209
 - WORM
 - ADSM support 209

- ost
 - standard AIX device driver 20
- other SCSI tape (see ost) 20
- ownership
 - tape volume control 6

P

- parconf
 - CLIO/S configuration file 191
 - NetTAPE configuration file 111
 - sample 114
- parsrv server
 - CLIO/S component 192
- performance
 - tape device
 - effect of data block size 10
 - effect of serpentine architecture 10
 - effect of tape drive technology 10
 - high-speed seeks to specific blocks 10
 - IBM 3590s 10
 - peak data transfer rate 10
 - streaming data to device 10
- physical volume library daemon
 - in NetTAPE 109
- picker
 - locating cartridges in a slot 31
 - on a tape library 31
- policy set
 - ADSM 43
- private
 - ADSM tape volumes 50
- publications
 - ADSM product library 256
 - CLIO/S 259
 - NetTAPE 259
 - redbooks 255

Q

- QIC
 - longitudinal recording technology 28
 - tape drives
 - ADSM support 209
 - NetTAPE support 235
- query content ADSM command 58

query drmedia ADSM command 71
query volhist ADSM command 49
query volume ADSM command 49

R

redbooks
 ADSM 255
 CLIO/S 256
reliability
 of tape drives 9
 of tape media 9
remote tape access
 centralization of tape resources 6
 using NetTAPE 16
restore AIX command 23
RISC System/6000
 attaching tape drives 245
 attaching tape libraries 245
 internal tape drives 245
rmdev AIX command 22
rmt AIX command 23
rmt*.x
 AIX device special files 19
 how devices are known to AIX 19
rotation
 of tape media using ADSM DRM 15

S

S370dd device driver
 low-level driver for parallel-channel-attached
 3480s 24
 low-level driver for parallel-channel-attached
 3490s 24
S390dd device driver
 low-level driver for ESCON-attached
 3480s 24
 low-level driver for ESCON-attached
 3490s 24
scratch
 ADSM tape volumes 50

SCSI

attachment to a RISC System/6000 245
connections
 differential SCSI bus 29
 SCSI fast 29
 SCSI wide 29

SCSI (continued)

connections (continued)
 SCSI-1 29
 SCSI-2 29
 single-ended bus 29
SCSI-attached tape library dataservers
 ADSM support 209
SCSI-attached tape transports
 ADSM support 209
serpentine track interleaving
 3x9x devices 28
 DLT devices 28
 longitudinal recording technology 28
 QIC devices 28
single drive reclamation process
 ADSM 221
single-ended SCSI bus 29
smcx
 device driver for 3590-B11 33
 device driver for 7331 33
space management
 ADSM 44
 type of tape application 7
SSD device drivers 24
standard labels
 records 219
status daemon
 in NetTAPE 109
 NetTAPE component 89
STK devices
 support with ADSM 46
 support with ADSM and NetTAPE 17
 support with NetTAPE 46
storage pool
 ADSM 44
 copy ADSM volumes 49
 migration tape to tape 63
 primary ADSM volumes 49
StorageTek
 ADSM support 237
 NetTAPE support 235

T

tape client daemon
 in NetTAPE 109

- tape data sharing
 - using CLIO/S 17
- tape device support
 - 3590-B11 support with ADSM 46
 - ADSM
 - 4 mm autochangers 209
 - 4 mm tape drives 209
 - 8 mm autochangers 209
 - 8 mm tape drives 209
 - device support modules 209
 - DLT tape drives 209
 - DLT tape libraries 209
 - ESCON-attached tape library
 - dataservers 209
 - ESCON-attached tape transports 209
 - manual device support 46
 - optical drives 209
 - parallel-attached tape library
 - dataservers 209
 - parallel-attached tape transports 209
 - QIC tape drives 209
 - SCSI-attached tape library
 - dataservers 209
 - SCSI-attached tape transports 209
 - StorageTek 237
 - NetTAPE
 - 3494 and 3495 tape libraries 235
 - 4 mm tape drives 235
 - 8 mm tape drives 235
 - 9-track tape drives 235
 - half-inch tape drives 235
 - QIC tape drives 235
 - StorageTek tape libraries 235
- tape drive sharing
 - using NetTAPE 16
- tape drive states
 - in NetTAPE
 - down 84
 - in use 84
 - pending reserve (PenRs) 84
 - reserved 84
 - up 84
- tape drives
 - 9-track support with NetTAPE 235
 - adding to an ADSM server 47
 - attaching to a RISC System/6000 245
 - ESCON attachment to a RISC
 - System/6000 245
- tape drives (*continued*)
 - half-inch support with NetTAPE 235
 - internal to a RISC System/6000 245
 - parallel channel attachment to a RISC
 - System/6000 245
 - preemption in ADSM 75
 - SCSI drive attachment to a RISC
 - System/6000 245
 - sharing using NetTAPE 16
 - utilization improvement with ADSM 15
- tape labels
 - ADSM support
 - automatic script 243
 - do-it-yourself labeling 53
 - dsmlabel in automatic mode 51
 - dsmlabel in manual mode 51
 - NetTAPE support
 - ANSI 99
 - bypass label processing 99
 - EBCDIC standard labels 99
 - expiration date and retention period 99
 - generation numbers 99
 - multiple files on a volume 99
 - overview 16, 85
 - unlabeled 99
 - with ADSM 166
 - records
 - hdr1 219
 - hdr2 219
 - vol1 219
- tape library
 - 3494 support with ADSM 47
 - 3495 support with ADSM 47
 - adding to ADSM server 47
 - ADSM support
 - 4 mm autochangers 209
 - 8 mm autochangers 209
 - device support modules 209
 - DLT tape libraries 209
 - ESCON-attached tape library
 - dataservers 209
 - parallel-attached tape library
 - dataservers 209
 - SCSI-attached tape library
 - dataservers 209
 - StorageTek 237
 - attaching to a RISC System/6000 245

- tape library (*continued*)
 - definition
 - accessor 31
 - locating cartridges in a slot 31
 - one or more tape drives 31
 - picker 31
 - several cartridge slots 31
 - volume inventory 31
 - ESCON attachment to a RISC
 - System/6000 245
 - NetTAPE support
 - 3494 and 3495 tape libraries 235
 - StorageTek tape libraries 235
 - parallel channel attachment to a RISC
 - System/6000 245
 - SCSI drive attachment to a RISC
 - System/6000 245
 - small SCSI library support with ADSM 46
 - types
 - 3494 32
 - 3495 32
 - 7331 32
 - 7332 32
 - barcode 32
 - external media identification capability 32
 - local intelligent with database 32
 - random access to slots 32
 - small 32
- tape management
 - common UNIX tape environments
 - mixed MVS and AIX tape environment 5
 - multiple systems with multiple SCSI tape devices 5
 - single SCSI tape device connected to several systems 4
 - single system with SCSI tape drive 3
 - single system with several SCSI tape devices 3
 - distributed systems environment approaches
 - centralized tape facilities 2
 - tape drives on every system 2
 - overview 1
 - requirements 38
 - trends 38
 - using ADSM 49
- tape operation
 - using ADSM 62
- tape performance
 - when using ADSM 78
- tape reclamation
 - ADSM 64
 - offsite copy volumes 72
 - single ADSM tape drive 65, 221
- tape server
 - daemon in NetTAPE 109
 - NetTAPE component 89
- tape technology
 - device connections and protocols 27
 - differential SCSI bus 29
 - ESCON 30
 - parallel channel 31
 - SCSI fast 29
 - SCSI wide 29
 - SCSI-1 29
 - SCSI-2 29
 - single-ended bus 29
 - recording techniques
 - helical scan 27
 - longitudinal 28
- tape usage considerations
 - application usage 12
 - business benefits 12
 - compression 9
 - device connectivity 11
 - drive cleaning 9
 - environmentals of device 12
 - maintenance overhead 12
 - manual and automated tape 10
 - media movement between storage locations
 - tracking the movement 11
 - using the ADSM DRM feature 11
 - media operation
 - automated tape libraries 12
 - IBM 3495 ATLDS 12
 - IBM 7331 12
 - manual tape devices 12
 - media type and reliability 9
 - performance 10
 - price of device 12
 - recording density 9
 - tape capacity 9
 - time and effort to operate devices 12

- tape volume
 - control
 - access rights 6
 - inventory 6
 - ownership 6
 - inventory management with ADSM 15
 - utilization improvement with ADSM 14
- tapechk AIX command 22
- tapes
 - sharing between AIX and mainframe applications 16
- tapesrv
 - CLIO/S tape pseudo device driver 195
 - NetTAPE daemon 98, 109
- tar
 - AIX command 23
 - using with NetTAPE 239
 - using with NetTAPE and multivolume data sets 240
- tcopy AIX command 23
- tctl AIX command 23

V

- vfile
 - NetTAPE file for storing information about a tape volume 101
 - NetTAPE volume information file 86
- volumes
 - ADSM
 - automated library full situation 55
 - checking into an automated library 54
 - checking out from an automated library 55
 - copy storage pool 49
 - database backup 49
 - export and import 49
 - primary storage pool 49
 - retiring and replacing 57

W

- WORM
 - ADSM support 209

X

- xcltopr NetTAPE command 128



Printed in U.S.A.

SG24-4705-00

