IBM Tape Device Drivers

Installation and User's Guide

GC27-2130-06

IBM Tape Device Drivers

Installation and User's Guide

GC27-2130-06

Note!

Before using this information and the product that it supports, be sure to read the general information under "Notices" on page 325.

Seventh Edition (July 2008)

This seventh edition of the *IBM Tape Device Drivers Installation and User's Guide*, GC27-2130-06, replaces and makes obsolete the following manuals:

- *IBM Tape Device Drivers Installation and User's Guide*, GC27-2130-05
- *IBM Tape Device Drivers Installation and User's Guide*, GC27-2130-04
- *IBM Tape Device Drivers Installation and User's Guide*, GC27-2130-03
- *IBM Tape Device Drivers Installation and User's Guide*, GC27-2130-02
- *IBM Tape Device Drivers Installation and User's Guide*, GC27-2130-01.
- *IBM Tape Device Drivers Installation and User's Guide*, GC27-2130-00.
- *IBM TotalStorage and System Storage Tape Device Drivers Installation and User's Guide*, GC35-0154-17.
- *IBM Ultrium Device Drivers Installation and User's Guide*, GC32-0430-13

Contents

Figures

Tables

Preface

These publications and URLs provide user information and installation assistance for IBM® tape drive, medium changer, and library device drivers.

Related Information

Reference material, including the Adobe® PDF version of this publication, is available at the following **ftp** site:

ftp://ftp.software.ibm.com/storage/devdrvr/Doc

A companion publication covering programming aspects for the device drivers is:

IBM Tape Device Drivers: Programming Reference, GA32-0566

Current Products

IBM TotalStorage Enterprise Tape System 3592 Publications

The following publications relate to the IBM TotalStorage® Enterprise Tape System 3592:

- *IBM TotalStorage Enterprise Tape System 3592 Operator Guide*, GA32-0465
- *IBM TotalStorage Enterprise Tape System 3592 Introduction and Planning Guide*, GA32-0464
- *IBM TotalStorage Silo Compatible Tape Frame 3592 Introduction, Planning, and User's Guide*, GA32-0463
- *IBM TotalStorage Enterprise Tape System 3592 Hardware Reference*, GA32-0466

IBM System Storage TS1120 Tape Drive

The following publications relate to the IBM System Storage™ TS1120 Tape Drive:

- *IBM System Storage TS1120 Tape Drive and Controller Introduction and Planning Guide*, GA32-0555
- *IBM System Storage TS1120 Tape Drive and Controller Operator Guide*, GA32-0556
- *IBM System Storage TS3500 Tape Library Operator Guide*, GA32-0560
- *IBM System Storage TS3500 Tape Library Introduction and Planning Guide*, GA32-0559
- *IBM System Storage 3953 Library Manager Model L05 Operator Guide*, GA32-0558
- *IBM System Storage 3953 Tape System Introduction and Planning Guide*, GA32-0557

IBM Ultrium Tape Drive Models T800 and T800F

The following publications relate to the IBM Ultrium Tape Drive Models T800 and T800F:

- *IBM Ultrium Tape Drive Models T800 and T800F Setup, Operator, and Service Guide*, GC26-7697

IBM LTO Ultrium 3-H Tape Drive

The following publications relate to the IBM LTO Ultrium 3-H Tape Drive:

- *IBM LTO Ultrium 3-H Tape Drive Setup, Operator, and Service Guide*, SC23-5231

IBM System Storage TS2230 Tape Drive

The following publications relate to the IBM System Storage TS2230 Tape Drive:

- *IBM System Storage TS2230 Tape Drive Setup, Operator, and Service Guide*, GC27-2099
- *IBM System Storage TS2230 Tape Drive Installation Quick Reference*, GC27-2100

IBM TotalStorage 3580 Tape Drive Models L33/L3H

The following publications relate to the IBM TotalStorage 3580 Tape Drive Models L33/L3H:

- *IBM TotalStorage 3580 Tape Drive Models L33/L3H Setup, Operator, and Service Guide*, GC26-7708
- *IBM TotalStorage 3580 Tape Drive Models L33/L3H Quick Reference*, GC26-7709

IBM System Storage TS2340 Tape Drive

The following publications relate to the IBM System Storage TS2340 Tape Drive [also known as IBM LTO4 (TS2340)]:

- *IBM System Storage TS2340 Tape Drive Models L43/S43 Setup, Operator, and Service Guide*, GC27-2103
- *IBM System Storage TS2340 Tape Drive Models L43/S43 Quick Reference Card*, GC27-2104

IBM System Storage TS3500 Tape Library (also known as IBM TotalStorage UltraScalable Tape Library 3584)

The following publications relate to the IBM System Storage TS3500 Tape Library:

- *IBM System Storage TS3500 Tape Library Planning and Operator Guide*, GA32-0408
- *IBM System Storage TS3500 Tape Library Maintenance Information*, 19P2440

IBM Virtualization Engine TS7520

The following publications relate to the IBM Virtualization Engine™ TS7520:

- *IBM Virtualization Engine TS7520 Hardware, Installation, Setup, and Problem Determination Guide*, GC26-7766-01
- *IBM Virtualization Engine TS7520 Introduction and Planning Guide*, GC27-2067-00
- *IBM Virtualization Engine TS7500 User Guide*, GC27-2068-00

IBM System Storage TS3400 Tape Library

The following publications relate to the IBM System Storage TS3400 Tape Library:

- *IBM System Storage TS3400 Tape Library Installation Quick Reference*, GA32-0573
- *IBM System Storage TS3400 Tape Library Planning and Operator Guide*, GC27-2107
- *IBM System Storage TS3400 Tape Library Maintenance Information*, GA32-0572

IBM System Storage TS3310 Tape Library

The following publications relate to the IBM System Storage TS3310 Tape Library:

- *IBM System Storage TS3310 Tape Library Setup and Operator Guide*, GA32-0477
- *IBM System Storage TS3310 Tape Library Maintenance Information*, GA32-0478
- *IBM System Storage TS3310 Tape Library SCSI Reference*, GA32-0476

IBM System Storage TS3100 Tape Library

The following publications relate to the IBM System Storage TS3100 Tape Library:

- *IBM System Storage TS3100 Tape Library and TS3200 Tape Library Setup, Operator, and Service Guide*, GA32-0454
- *IBM System Storage TS3100 Tape Library and TS3200 Tape Library Installation Quick Reference*, GA32-0456

- *IBM System Storage TS3100 Tape Library and TS3200 Tape Library SCSI Reference*, GA32-047

IBM System Storage TS3200 Tape Library

The following publications relate to the IBM System Storage TS3200 Tape Library:

- *IBM System Storage TS3100 Tape Library and TS3200 Tape Library Setup, Operator, and Service Guide*, GA32-0454
- *IBM System Storage TS3100 Tape Library and TS3200 Tape Library Installation Quick Reference*, GA32-0456
- *IBM System Storage TS3100 Tape Library and TS3200 Tape Library SCSI Reference*, GA32-047

AIX

The following URL points to information about IBM System p™ (also known as@server pSeries®) servers:

http://www-1.ibm.com/servers/eserver/pseries

HP-UX

The following URL relates to HP HP-UX systems:

http://www.hp.com

Linux

The following URLs relate to Linux® distributions:

http://www.redhat.com

http://www.suse.com

Solaris

The following URL relates to Sun Solaris systems:

http://www.sun.com

Microsoft Windows

The following URL relates to Microsoft® Windows® systems:

http://www.microsoft.com

Legacy Products

IBM TotalStorage Ultrium External Tape Drive 3580

The following publications relate to the IBM TotalStorage Ultrium External Tape Drive 3580:

- *IBM 3580 Ultrium Tape Drive Setup, Operator, and Service Guide*, GA32-0415

IBM TotalStorage Ultrium Tape Autoloader 3581

The following publication relates to the IBM TotalStorage Ultrium Tape Autoloader 3581:

- *IBM 3581 Ultrium Tape Autoloader Setup, Operator, and Service Guide*, GA32-0412

IBM TotalStorage 3581 Tape Autoloader (also known as IBM TotalStorage Ultrium Tape 2U Autoloader 3581)

The following publication relates to the IBM TotalStorage 3581 Tape Autoloader:

- *IBM 3581 Tape Autoloader Models L28/L38/L3H and F28/F38/F3H Setup, Operator, and Service Guide*, GA32-0470

IBM TotalStorage 3582 Tape Library (also known as IBM TotalStorage Ultrium Tape Library 3582)

The following publications relate to the IBM TotalStorage 3582 Tape Library:

- *IBM TotalStorage 3582 Tape Library Setup, Operator, and Service Guide*, GA32-0458

IBM TotalStorage 3583 Tape Library (also known as IBM TotalStorage Ultrium Scalable Tape Library 3583)

The following publications relate to the IBM TotalStorage Ultrium Scalable Tape Library:

- *IBM 3583 Tape Library Setup and Operator Guide*, GA32-0411
- *IBM 3583 Tape Library Service Guide*, GA32-0425
- *IBM Storage Area Network Data Gateway Module Setup, Operator, and Service Guide*, GA32-0436

IBM Virtualization Engine TS7510

The following publications relate to the IBM Virtualization Engine TS7510:

- *IBM Virtualization Engine TS7510 Hardware, Installation, Setup, and Problem Determination Guide*, GC26-7766
- *IBM Virtualization Engine TS7510 Introduction and Planning Guide*, GC26-7767
- *IBM Virtualization Engine TS7510 User Guide*, GC26-7769

IBM TotalStorage Enterprise Automated Tape Library 3494

The following publications relate to the IBM TotalStorage Enterprise Automated Tape Library (3494):

- *IBM TotalStorage Enterprise Automated Tape Library (3494) Introduction and Planning Guide*, GA32-0448
- *IBM TotalStorage Enterprise Automated Tape Library (3494) Operator Guide*, GA32-0449
- *IBM TotalStorage 3494 Tape Library Maintenance Information*, SA37-0407

IBM 3490E Tape Subsystem

The following publications relate to the IBM 3490E Tape Subsystem:

- *IBM 3490 Magnetic Tape Subsystem Enhanced Capability Models C10, C11, C1A, C22, and C2A Hardware Reference*, GA32-0219
- *IBM 3490 Magnetic Tape Subsystem Enhanced Capability Models E01 and E11 User's Guide*, GA32-0298
- *IBM 3490 Magnetic Tape Subsystem Enhanced Capability Models C10, C11, C1A, C22, and C2A Maintenance Information*, SA37-0299
- *IBM 3490E Tape Subsystem, Models F01, F1, F11, and FC0, Installation, Planning, and Operator's Guide*, GA32-0378

IBM Magstar MP 3570 Tape Subsystem

The following publications relate to the IBM Magstar® MP 3570 Tape Subsystem:

- *IBM Magstar MP 3570 Tape Subsystem Operator's Guide*, GA32-0345
- *IBM Magstar MP 3570 Tape Subsystem Introduction and Planning Guide*, GA32-0348
- *IBM Magstar MP 3570 Tape Subsystem Hardware Reference*, GA32-0365
- *IBM Magstar MP 3570 Tape Subsystem Hardware Reference C-Series Models*, GA32-0394

IBM Magstar MP 3575 Tape Subsystem

The following publications relate to the IBM Magstar MP 3575 Tape Subsystem:

- *IBM Magstar MP 3575 Tape Library Dataserver Introduction and Planning Guide*, GA32-0380
- *IBM Magstar MP 3575 Tape Library Dataserver Operator Guide*, GA32-0381
- *IBM Magstar MP 3575 Tape Library Dataserver Hardware Reference*, GA32-0382

IBM TotalStorage Enterprise Tape System 3590

The following publications relate to the IBM TotalStorage Enterprise Tape System 3590:

- *IBM TotalStorage Enterprise Tape System 3590 Introduction and Planning Guide*, GA32-0329
- *IBM TotalStorage Enterprise Tape System 3590 Operator Guide*, GA32-0330
- *IBM TotalStorage Enterprise Tape System 3590 Hardware Reference*, GA32-0331
- *IBM TotalStorage Enterprise Tape System 3590 Operator's Quick Guide*, GA32-0354

IBM 7331 8mm Tape Library

The following publications relate to the IBM 7331 8mm Tape Library:

- *IBM 7331 8mm Tape Library Installation Guide*, SA26-7110
- *IBM 7331 8mm Tape Library Operator Guide*, SA26-7111

IBM 7332 4mm DDS-2 Tape Cartridge Autoloader

The following publications relate to the IBM 7332 4mm DDS-2 Tape Cartridge Autoloader:

- *IBM 7332 4mm DDS-2 Tape Cartridge Autoloader Model 005 Installation Guide*, SA26-7138
- *IBM 7332 4mm DDS-2 Tape Cartridge Autoloader Model 005 Operator Guide*, SA26-7139
- *IBM 7332 4mm DDS-2 Tape Cartridge Autoloader Model 005 Service Guide*, SA26-7140

IBM 7332 Model 220 External 4mm Tape Autoloader

The following publications relate to the IBM 7332 Model 220 External 4mm Tape Autoloader:

- *7332 Model 220 4mm Tape Autoloader, Setup and Operator Guide*, SA26-2005
- *7332 Model 220 4mm Tape Autoloader Service Guide*, SY32-0408

IBM 7334 8mm Tape Library

The following publications relate to the IBM 7334 8mm Tape Library:

- *7334 Model 410 8mm Tape Library Setup and Operator Guide*, SA26-2009
- *7334 Model 410 8mm Tape Library Service Guide*, SY32-0412

IBM 7336 4mm Tape Library

The following publication relates to the IBM 7336 4mm Tape Library:

- *IBM 7336 4mm Tape Library Model 205 Operator and Setup Guide*, SA37-0309

IBM 3447 and 7337 Digital Linear Tape Library

The following publications relate to the IBM 3490E Tape Subsystem:

- *3447 and 7337 Digital Linear Tape Library Setup and Operator Guide*, GA32-0399
- *3447 and 7337 Digital Linear Tape Library Service Guide*, GA32-0400
- *7337 Model 360 Digital Linear Tape Library Setup and Operator Guide*, SA41-0051

Tru64

The following URL points to information about Tru64 systems:

http://www.hp.com

SGI

The following URL relates to SGI systems:

http://www.sgi.com

Other Publications

IBM Storage Area Network Data Gateway Installation and User's Guide, SC26-7304

Additional Information

The following publication contains additional information related to the IBM tape drive, medium changer, and library device drivers:

- *American National Standards Institute Small Computer System Interface*

  X3T9.2/86-109 X3.180, X3B5/91-173C, X3B5/91-305, X3.131-199X Revision 10H, and X3T9.9/91-11 Revision 1

How to Send Your Comments

Your feedback is important in helping us provide the most accurate and best quality information. If you have comments or suggestions for improving this publication, you can send us comments electronically by using these addresses:

- Internet: starpubs@us.ibm.com
- IBMLink™ from U.S.A.: STARPUBS at SJEVM5
- IBMLink from Canada: STARPUBS at TORIBM
- IBM Mail Exchange: USIB3VVD at IBMMAIL
- Fax from U.S.A., Canada, and other countries: 520-799-2906

Special Printing Instructions

This Tape Device Drivers Installation and User's Guide contains different sections for each type of operating system/platform; for example, AIX®, Tru64, HP-UX, Linux, Sun Solaris, Windows; and a separate section on these operating systems for the 3494 Enterprise Tape Library.

Note: When selecting the page range for the section you wish to print, note that the print page range is based on the page controls for Adobe Acrobat®, not the page printed on the actual document. Enter the Adobe page numbers to print.

Important printer note

If you wish to print one or more separate sections of the manual, follow these steps:

1. Navigate to the beginning of the section and note the page number.
2. Navigate to the last page in the section and note that page number.
3. Select File —> Print, then choose "Pages" and enter the page range for the section. Only the page range entered will print.
4. Repeat these steps to print additional sections.

Attention: There is only one Table of Contents and one Index for this entire book. If you wish to print those items, you must repeat the process above, entering the page range for the Table of Contents and the Index page range, respectively.

Chapter 1. Introduction

This publication describes the IBM Tape and Medium Changer Device Drivers for AIX, HP-UX, Linux, Solaris, and Windows operating systems. Not all devices are supported on all operating systems. Refer to the specific operating system/platform chapters for details on device support.

Purpose

The IBM tape and medium changer device drivers are designed specifically to take advantage of the features provided by the IBM tape drives and medium changer devices. The goal is to give applications access to the functions required for basic tape functions (such as backup and restore) and medium changer operations (such as cartridge mount and demount), as well as to the advanced functions needed by full tape management systems. Whenever possible, the driver is designed to take advantage of the device features transparent to the application. Please note that some ISV applications have certain device driver requirements. Before you install the device drivers, please refer to the ISV web site or their support to find out what device drivers should be used for the ISV.

IBM Tape Products

The IBM Tape product family provides an excellent solution for customers with small to large storage and performance requirements.

Current Products

Figure 1 on page 2 illustrates the attachment of various current products to an open systems server.

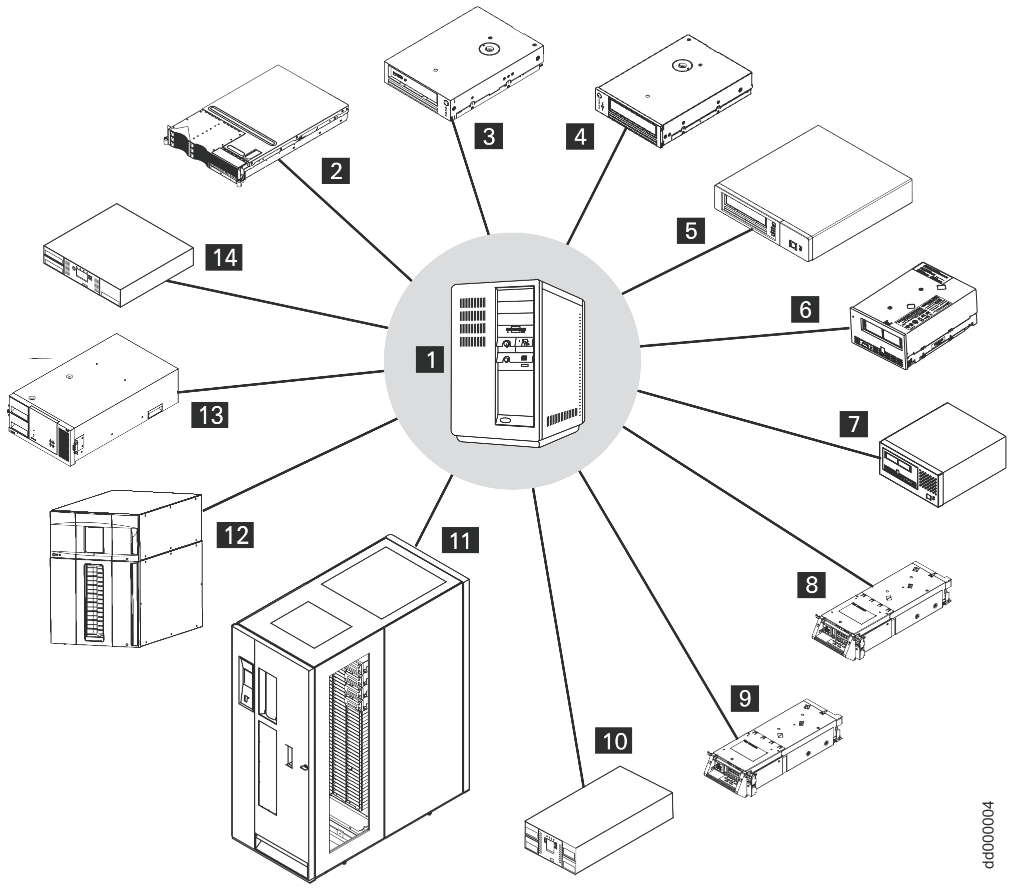


Figure 1. Current Attachment Array

1 Open Systems Server	**8** IBM TotalStorage Enterprise Tape Drive
2 IBM Virtualization Engine TS7520	**9** IBM System Storage TS1120 Tape Drive
3 IBM System Storage TS2230 Tape Drive	**10** IBM System Storage TS3200 Tape Library
4 IBM LTO Ultrium 3-H Tape Drive and IBM LTO Ultrium 4 Half-High Tape Drive	**11** IBM System Storage TS3500 Tape Library
5 IBM System Storage TS2340 Tape Drive [also known as IBM LTO4 (TS2340)]	**12** IBM System Storage TS3310 Tape Library
6 IBM TotalStorage Ultrium Tape Drive	**13** IBM System Storage TS3400 Tape Library
7 IBM TotalStorage 3580 Tape Drive	**14** IBM System Storage TS3100 Tape Library

Figure 2 on page 3 illustrates an Ultrium environment that could include an IBM 3583 Tape Library and an IBM TS3500 Tape Library.

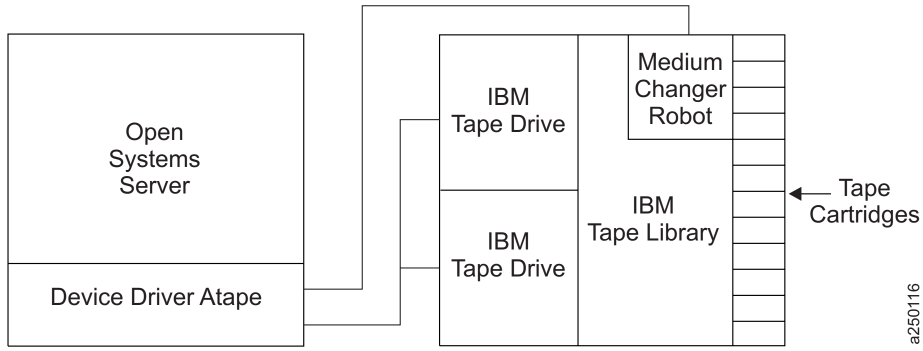


Figure 2. Example of an IBM Tape Environment

IBM Virtualization Engine TS7520

The IBM Virtualization Engine TS7520 product is a virtual-tape solution designed to help optimize tape processing. Through the implementation of an integrated, tiered storage hierarchy of disk and tape, the benefits of both technologies can be leveraged to help enhance performance and provide the capacity needed for today's tape processing requirements. Deploying this innovative offering can help reduce batch processing time and management overhead.

IBM System Storage TS3500 Tape Library (also known as IBM TotalStorage UltraScalable Tape Library 3584)

The IBM System Storage TS3500 Tape Library is a highly scalable, automated tape library combining IBM automation technology for midrange to enterprise open systems environments. The TS3500 Tape Library supports logical partitioning and can house both IBM LTO Ultrium and IBM TotalStorage Enterprise 3592 Tape Drives (in separate frames). With scalability of one to sixteen frames and up to 192 tape drives, the TS3500 tape library can have an expanded available storage capacity of over 5000 TB (with 3:1 compression) using TotalStorage Enterprise 3592 Tape Drives.

IBM System Storage TS3400 Tape Library

The IBM SYSTEM STORAGE™ TS3400 TAPE LIBRARY is designed to offer high performance drive technology and automation for the open systems environment. The TS3400 Tape Library is an addition to the family of IBM System Storage tape library offerings. The TS3400 Tape Library is an external 5U stand-alone or rack-mountable unit that incorporates up to two IBM System Storage TS1120 Tape Drives Model 3592-E05 (Machine Type 3592, Model E05), which are ordered separately. It comes with 4 Gbps dual-ported switched fabric Fibre Channel attachment. The TS1120 Tape Drive has a native data transfer rate of up to 100 MB/ sec. per drive.

IBM System Storage TS3310 Tape Library

The IBM System Storage TS3310 Tape Library is a modular, scalable tape library designed to scale vertically with expansion for Linear Tape-Open (LTO) tape cartridges, drives, and redundant power supplies. The base library module, model L5B, is the entry point for the product family. It contains all of the necessary robotics and intelligence to manage the 5U high library system, which houses up to 36 cartridges (30 storage slots and 6 Input/Output slots) and two LTO generation 3 tape drives. The TS3310 model L5B can be expanded with the addition of expansion units, the model E9U. Each model E9U contains 92 physical LTO cartridge storage cells and space for up to four LTO generation 3 tape drives.

Additionally, the E9U has space for up to two (one redundant) power supply modules. (At least one power supply module must be installed if a drive is present in the E9U.)

IBM System Storage TS3200 Tape Library

The IBM System Storage TS3200 Tape Library Express Model offers high capacity and performance technology for the midrange open systems environments. The TS3200 Tape Library is an external 4U standalone or rack-mountable unit that incorporates up to two IBM TotalStorage Ultrium 3 Tape Drives, and has a native data rate of up to 80 Mbps per drive.

IBM System Storage TS3100 Tape Library

The IBM System Storage TS3100 (Machine Type 3573) Tape Library provides a compact, high-capacity, low-cost solution for simple, unattended data backup. This unique design houses up to 22 tape cartridges in a compact 2U form factor with easy access to tape cartridges via two removable magazines and an Input/Output (I/O) Station.

IBM TotalStorage Enterprise Tape Subsystem 3592

The IBM TotalStorage Enterprise Tape Subsystem 3592 provides higher levels of performance, reliability, and cartridge capacity than the IBM TotalStorage Tape Subsystem 3590. The 3592 tape drive has a dual-ported 2-Gbps Fibre Channel interface, Fibre Channel attachment, and a new high-technology design that increases the native data rate to 40 MB/sec.

IBM System Storage TS1120 Tape Drive

The IBM System Storage TS1120 Tape Drive is the second generation of the 3592 tape drive. It provides higher levels of performance, reliability, and cartridge capacity than the IBM TotalStorage Tape System 3590 and the 3592 tape drive. The TS1120 tape drive has a dual-ported 4-Gbps Fibre Channel interface, Fibre Channel attachment, and a new high-technology design that increases the native data rate to 100 MB/sec.

IBM TotalStorage 3580 Tape Drive (Models L33 and L3H)

The IBM TotalStorage 3580 Tape Drive Model L33/L3H offers high-capacity, performance, and technology designed for the midrange open systems environment. This model incorporates the Linear Tape-Open (LTO) IBM TotalStorage Ultrium Tape Drive (Generation 3), which more than doubles maximum tape drive performance over the Generation 2 LTO Ultrium Tape Drive (Ultrium 2). This tape drive comes with a SCSI Ultra160 LVD attachment, for connection to a wide spectrum of open system servers.

IBM System Storage TS2230 Tape Drive

The IBM System Storage TS2230 Tape Drive Model H3L (Machine Type 3580, Model H3L, SAP part number 3580L3E) is a high-performance, high-capacity data-storage device that is designed to backup and restore open systems applications. This model incorporates the Linear Tape-Open (LTO) IBM Ultrium 3 Half-High Tape Drive. It is available with a Small Computer Systems Interface (SCSI) or with a Serial Attached SCSI (SAS) interface. The drive has a native storage capacity of 400 GB per cartridge (800 GB at 2:1 compression) on the Ultrium 3 data cartridge.

IBM TotalStorage Ultrium Tape Drive (Models T800 and T800F)

The IBM TotalStorage Ultrium Tape Drive is a high-performance, high-capacity data-storage device that is designed to backup and restore open systems applications. The drive can be integrated into an enclosure, such as a desktop unit, tape autoloader, or tape library. The drive is the third-generation in the Ultrium series of products. It is available as Model T800 with a Small Computer Systems Interface (SCSI) or as Model T800F with a Fibre Channel interface. The drive has a native storage capacity of 400 GB per cartridge (800 GB at 2:1 compression).

IBM LTO Ultrium 3-H Tape Drive and IBM LTO Ultrium 4 Half-High Tape Drive

The IBM LTO Ultrium 3-H Tape Drive and IBM LTO Ultrium 4 Half-High Tape Drive are high-performance, high-capacity data-storage devices in a half-height form factor that are designed to backup and restore open systems applications. Both drives can be integrated into an enclosure, such as a desktop unit, tape autoloader, or tape library. They are available with a Small Computer Systems Interface (SCSI) or with a Serial Attached SCSI (SAS) interface. The IBM LTO Ultrium 3-H Tape Drive has a native storage capacity of 400 GB per cartridge (800 GB at 2:1 compression). The IBM LTO Ultrium 4 Half-High Tape Drive has a native storage capacity of 800 GB per cartridge (1600 GB at 2:1 compression).

IBM System Storage TS2340 Tape Drive

The IBM System Storage TS2340 Tape Drive [also known as IBM LTO4 (TS2340)] is a high-performance, high-capacity data-storage device for the midrange open systems environment. It is designed to increase maximum tape drive throughput native data rate performance up to 120 MB/sec compared to the IBM LTO generation 3 Tape Drive (Ultrium 3) at 80 MB/sec native data transfer rate. In addition, with the use of the new IBM LTO Ultrium 800 Gb Data Cartridge, the Ultrium 4 Tape Drive doubles the tape cartridge capacity up to 800 GB native physical capacity (1600 GB with 2:1 compression). It is available with Small Computer System Interface (SCSI) or Serial Attached SCSI (SAS) interface. The Ultrium 4 SAS Tape Drive is encryption-capable and designed to support Application Managed Encryption.

Legacy Products

IBM TotalStorage Ultrium Scalable Tape Library 3583

The IBM TotalStorage Ultrium Scalable Tape Library 3583 is an automated tape library that incorporates IBM Ultrium tape drives in either a stand alone or optional rack mount configuration. Three different library models are available with storage capacities of 18 through 72 slots and one to six Ultrium tape drives. The IBM TotalStorage Ultrium Scalable Tape Library 3583 can be used for save, restore and mass storage archives where multiterrabyte capacities are required.

IBM TotalStorage Ultrium Tape Library 3582

The IBM TotalStorage Ultrium Tape Library 3582 is an entry tape library incorporating high-performance IBM TotalStorage Ultrium generation 2 Tape Drives for the midrange open systems environment. It can accommodate one or two Ultrium generation 2 Tape Drives and comes standard with a one-cartridge I/O station and 23 data cartridge slots giving a native library capacity of 4.8 TB uncompressed native storage.

IBM TotalStorage Ultrium Tape 2U Autoloader 3581

The Ultrium Tape 2U Autoloader 3581 is an external 2U stand-alone or rack-mountable unit that incorporates a single IBM LTO Ultrium 2 tape drive. The Ultrium Tape 2U Autoloader 3581 capacity is eight tape cartridges, providing a media capacity of up to 1.6 TB (3.2 TB with 2:1 compression) data storage per unit.

The Model L28 comes with a LVD Ultra160 SCSI attachment, while the Model F28 comes with a Native Switched Fabric Fibre Channel attachment, for connection to a wide spectrum of open systems servers.

IBM TotalStorage Ultrium Tape Autoloader 3581

The IBM TotalStorage Ultrium Tape Autoloader 3581 is an external, stand-alone, or rack-mounted tape autoloader that incorporates one IBM Ultrium Tape Drive. The autoloader has seven storage slots giving the autoloader up to 1,400 GB of uncompressed data storage. The autoloader can be used with compatible software applications to automate backup/recovery or other data storage activities.

IBM TotalStorage Ultrium External Tape Drive 3580

The IBM TotalStorage Ultrium External Tape Drive 3580 is a stand alone, large capacity, high performance tape drive that adheres to the Linear Tape-Open (LTO) specifications and supports the IBM Ultrium format tape. The drive supports native tape capacities of up to 200 GB with uncompressed data transfer rates of up to 35 MB per second. With both Low Voltage Differential (LVD) Ultra-2 SCSI and High Voltage Differential (HVD) Ultra SCSI attachments, this device is suitable for a variety of save, restore, and archiving requirements for PC and Open Systems platforms.

StorageSmart by IBM Ultrium Products

The StorageSmart™ by IBM Ultrium family of products is compatible with the IBM branded versions of the Ultrium family products. In this manual, where there is no specific mention of the StorageSmart by IBM Ultrium branded set of products, the documentation pertaining to the IBM branded family of products should be used. Table 1 cross references the StorageSmart by IBM Ultrium product set to the IBM Ultrium product set.

Table 1. Ultrium Product Comparison

StorageSmart by IBM Ultrium	IBM Version of Ultrium
StorageSmart by IBM Ultrium External Tape Drive TX200	IBM TotalStorage Ultrium External Tape Drive 3580
StorageSmart by Ultrium Tape Autoloader SL7	IBM TotalStorage Ultrium Tape Autoloader 3581
StorageSmart by Ultrium Scalable Tape Library SL72	IBM TotalStorage Ultrium Scalable Tape Library 3583
No equivalent in the IBM Ultrium product set	IBM TotalStorage UltraScalable Tape Library 3584

IBM TotalStorage Enterprise Automated Tape Library 3494

The IBM TotalStorage Enterprise Automated Tape Library 3494 is an automated tape library, providing an automated tape solution to a variety of system environments. It provides a solution for automating tape operations such as save/restore, migration of data between direct access storage devices (DASD) and tape, and other mass data applications.

IBM TotalStorage Enterprise Tape System 3590

The IBM TotalStorage Enterprise Tape System 3590 consists of a tape drive combined with an automatic cartridge facility (ACF). In addition to the functions that the ACL or CSL provide for the IBM 3490E Magnetic Tape Subsystem, the ACF on the Enterprise Tape System 3590 can operate as an integrated medium changer device to provide random access of cartridges in the magazine cells under program control.

Note: The 3590 drive has two Fibre Channel addressing modes. When attaching an open systems server to a 3590 Fibre Channel drive, you must use the hard addressing mode. When attaching a 3590 drive directly to a host (without a switch), use lower addresses for hosts and higher addresses for drives.

IBM TotalStorage Virtual Tape Server (Models B10, B18, and B20)

The IBM TotalStorage Virtual Tape Server (VTS) product delivers an increased level of storage capability to the traditional storage product hierarchy. To the host software, a VTS subsystem looks like a 3490E Enhanced Capability Tape Subsystem with associated Cartridge System Tape or Enhanced Capacity Cartridge System Tape.

IBM Magstar MP 3570 Tape Subsystem

The IBM Magstar MP 3570 Tape Subsystem consists of one or two tape drives (depending on the model) integrated with a full capability medium changer robotic facility, which provides random access of cartridges in the storage cells under program control.

IBM Magstar MP 3575 Tape Library Dataserver

The IBM Magstar MP 3575 Tape Library Dataserver is a scalable family of high-performance tape libraries that use the IBM Magstar MP tape drives. The IBM device drivers control both the Magstar MP tape drives and the robotic cartridge handling mechanism in these tape libraries that instructs the robot to mount and demount the cartridges.

IBM Virtualization Engine TS7510

The IBM Virtualization Engine TS7510 product delivers and increased level of storage capability to the traditional storage product hierarchy. To the host software, the IBM Virtualization Engine TS7510 looks like a 3584 library with associated tape drives.

IBM 3490E Magnetic Tape Subsystem

The IBM 3490E Magnetic Tape Subsystem consists of one or two tape drives (depending on the model) combined with an automatic cartridge loader (ACL) or cartridge stack loader (CSL). The ACL or CSL can load the next tape from the cartridge stack into the tape drive either automatically when a tape is unloaded, or when the tape drive is under program control.

IBM 7332 4mm Tape Cartridge Autoloader

The IBM Magstar MP 3575 Tape Library Dataserver is a scalable family of high-performance tape libraries that use the IBM Magstar MP tape drives. The IBM device drivers control both the Magstar MP tape drives and the robotic cartridge handling mechanism in these tape libraries that instructs the robot to mount and demount the cartridges.

IBM 7331, 7334, 7336, or 7337 Tape Medium Changer

The IBM 7331 and 7334 products are 8mm Tape Medium Changer Library devices. The IBM 7336 product is a 4mm Tape Medium Changer Library device. The IBM 7337 product is a DLT Tape Medium Changer Library device. The AIX Enhanced Device Driver controls the robotic cartridge handling mechanism in these tape libraries that instructs the robot to mount and demount the cartridges. Access to the tape drives within these libraries is through the native operating system AIX Tape Device Driver.

Chapter 2. Common Extended Features

Purpose

This chapter provides general information about the IBM Device Drivers. Refer to the specific operating system/platform chapters for details on device support.

Supported Hardware

Table 2 lists the IBM device drivers that are supported.

Table 2. Supported Hardware by Platform

DEVICE	AIX	HP-UX	Linux			Solaris	Tru64	Windows			
			Intel® and AMD Opteron Processors	System p Models	System z™ Models			2000	Server 2003	NT	Server 2008
IBM LTO Ultrium Tape Drive	X	X (not LTO4 on HP-UX 11.0)	X	X	X	X	X (not LTO 4)	X (only LTO 1)	X	X (LTO 1 and LTO 2 only)	
IBM System Storage TS1120 Tape Drive Model E05	X	X	X	X	X	X		X	X		X
IBM System Storage TS2230 Tape Drive	X	X (only SCSI LVD)	X	X		X			X		X
IBM System Storage TS2240 Tape Drive	X		X	X		X			X		X
IBM System Storage TS2340 Tape Drive	X	X (not HP-UX 11.0)	X	X		X			X		X
IBM System Storage TS3100 Tape Library	X	X	X	X	X	X		X	X		X
IBM System Storage TS3200 Tape Library	X	X	X	X	X	X		X	X		X

Table 2. Supported Hardware by Platform (continued)

DEVICE	AIX	HP-UX	Linux			Solaris	Tru64	Windows			
			Intel® and AMD Opteron Processors	System p Models	System z™ Models			2000	Server 2003	NT	Server 2008
IBM System Storage TS3310 Tape Library	X	X	X	X	X	X		X	X		X
IBM System Storage TS3400 Tape Library	X	X				X			X		X
IBM System Storage TS3500 Tape Library	X	X	X	X	X	X		X	X		X
IBM System Storage Virtualization Engine TS7520	X	X (not HP-UX 11.0, 11i v3)	X	X	X	X			X		
IBM 3490E Model C10, C11, C1A, C22, C2A with FC5040	X					X		X	X		
IBM 3490E Model E01, E11, F00, F01, F11, F1A	X	X				X					
IBM TotalStorage Enterprise Tape Library 3494	X	X	X	X	X	X	X	X	X	X	X
IBM TotalStorage Ultrium External Tape Drive 3580	X	X (not LTO 4 on HP-UX 11.0)	X	X		X	X (only LTO 1)	X (not LTO 4)	X	X (LTO 1 and LTO 2 only)	X
IBM TotalStorage Ultrium Tape Autoloader 3581	X	X	X	X		X	X	X	X	X	
IBM TotalStorage Ultrium Tape 2U Autoloader 3581	X	X	X	X	X	X		X	X		
IBM TotalStorage Ultrium Tape Library 3582	X	X	X	X	X	X		X	X	X	

Table 2. Supported Hardware by Platform (continued)

DEVICE	AIX	HP-UX	Linux			Solaris	Tru64	Windows			
			Intel® and AMD Opteron Processors	System p Models	System z™ Models			2000	Server 2003	NT	Server 2008
IBM TotalStorage Ultrium Scalable Tape Library 3583	X	X	X	X	X	X	X	X	X	X	
IBM TotalStorage UltraScalable Tape Library 3584	X	X	X	X	X	X	X	X	X	X	
IBM TotalStorage Enterprise Tape System 3590, Models B11, B1A	X	X	X			X		X	X	X	
IBM TotalStorage Enterprise Tape System 3590, Models E11, E1A, H11, H1A	X	X	X	X	X	X	X	X	X	X	
IBM TotalStorage Enterprise Tape System 3592, Model J1A	X	X	X	X	X	X		X	X	X	
IBM Virtualization Engine TS7510	X	X	X	X	X	X		X	X		
IBM TotalStorage Virtual Tape Server, Models B10, B18, B20	X	X				X		X	X	X	
IBM Magstar MP 3570 Tape Subsystem Model B00, B01, B02, B11, B12, B1A	X	X				X		X		X	
IBM Magstar MP 3570 Tape Subsystem Model C00, C01, C02, C11, C12, C1A	X	X				X		X		X	

Table 2. Supported Hardware by Platform (continued)

DEVICE	AIX	HP-UX	Linux: Intel® and AMD Opteron Processors	Linux: System p Models	Linux: System z™ Models	Solaris	Tru64	Windows: 2000	Windows: Server 2003	Windows: NT	Windows: Server 2008
IBM Magstar MP 3570 Tape Subsystem Model L06, L12, L18, L24, L32	X	X				X		X		X	
IBM 7331 8mm Tape Library Model 205 or 305 (support for autochanger only)	X										
IBM 7332 4mm Tape Cartridge Autoloader Model 005, 110, 220	X										
IBM 7334 8mm Tape Library Model 410	X										
IBM 7336 4mm Tape Library Model 205 or 305 (support for autochanger only)	X					X					
IBM 7337 DLT Tape Library (support for autochanger only)	X										

For detailed requirements for each operating system/platform, refer to the appropriate chapter. To obtain the most current service and documentation for this software, refer to Appendix A, “Accessing Documentation and Software Online,” on page 305.

Path Failover and Load Balancing

Device driver path failover support will configure multiple physical paths to the same device within the device driver and provides two basic functions:

1. Automatic failover to an alternate physical path when a permanent error occurs on one path.
2. Dynamic load balancing for tape devices using multiple Host Bus Adapters (HBA).

Path Failover is supported on certain tape products with the latest IBM device drivers available on the ftp download site: ftp://ftp.software.ibm.com/storage/devdrvr. Some devices require a path failover feature code to be installed prior to enabling the path failover support in the device driver. Refer to "Supported Devices and Feature Codes" on page 14 for a list of supported devices and what path failover feature code may be required for your machine type.

Automatic Failover

The automatic failover support provides error recovery on an alternate path when a permanent error occurs on one path. This is transparent to the running application.

For example, consider a simple multipath connection that consists of two Host Bus Adapters (HBA) connected through a switch to the tape drive. The first HBA is connected to port 0 of the drive, and the second HBA is connected to port 1 of the drive. This connection provides two physical paths to same tape drive for redundancy if one path from an HBA to the drive fails.

At startup or configuration, the system detects two logical devices of the tape drive. Each logical device is a physical path to the same tape drive. A backup and restore application can open and use only one logical device at a time because they represent the same physical device.

Without path failover support, if a permanent path error occurs (because of an HBA or cable failure, for example), the application fails. It is possible to initiate manual failover by restarting the application on the alternate logical device, but the application has to be restarted from the beginning. A long backup or restore operation may have been in progress when the path error occurred. Sometimes manual failover may require operator intervention to reset the drive because a SCSI Reservation could still exist on the failing HBA path.

When the path failover support is enabled on both logical devices, the device driver configures them internally as a single device with multiple paths. The application can still open and use only one logical device at a time. If an application opens the primary device and a permanent path error occurs, the device driver initiates failover error recovery automatically on the alternate path. If successful, the current operation continues on the alternate path without interrupting the application. The failover error recovery first restores the previous device state, SCSI Reservation, and tape position, then retries the failing operation.

Dynamic Load Balancing

The dynamic load balancing support optimizes resources for tape devices that have physical connections to multiple Host Bus Adapters (HBA) in the same machine. When an application opens a device that has multiple HBA paths configured, the device driver determines which path has the HBA with the lowest usage, and assigns that path to the application. When another application opens a different device with multiple HBA paths, the device driver again determines the path with the lowest HBA usage and assigns that path to the second application. The device driver will update the usage on the HBA assigned to the application when the device is closed. Dynamic load balancing will use all Host Bus Adapters whenever possible and balance the load between them to optimize the resources in the machine.

For example, consider a machine with two Host Bus Adapters, HBA1 and HBA2, with multiple tape drives attached. Each tape drive is connected to both HBA1 and

HBA2. Initially, there are no tape drives currently in use. When the first application opens a tape drive for use, the device driver will assign the application to use HBA1. When a second application opens a tape drive for use, the device driver will assign the second application to use HBA2. A third application would be assigned to HBA1 and a fourth application would be assigned to HBA2. There would be two applications using HBA1 and two applications using HBA2.

If the first application finishes and closes the device, there would now be one application using HBA1 and two applications using HBA2. When the next application opens a tape drive, it would be assigned to HBA1, so again there would be two applications using HBA1 and two applications using HBA2. Likewise, if the second application finishes and closes the device, HBA2 would have one application using it and the next application that opens a tape drive would be assigned to HBA2.

The dynamic load balancing support is independent from the automatic failover support. Regardless of the path assigned initially for load balancing, if that path fails the automatic failover support will attempt recovery on the next available path.

Supported Devices and Feature Codes

Path failover is supported only for the devices listed in Table 3 on page 15. Path failover includes Control Path Failover (CPF) for tape libraries and Data Path Failover (DPF) for tape drives. In order to use path failover support, some devices require feature codes as listed in Table 3 on page 15.

Table 3. Supported Devices and Feature Codes

Supported Tape Library	Feature Code (FC), if required
TS1120	Standard, no FC required (DPF only)
3592	Standard, no FC required (DPF only)
TS3500	FC 1682 (CPF and DPF)
TS3400	Standard, no FC required (CPF and DPF)
TS3310	FC 1682 (CPF and DPF)
TS3200	FC 1682 (CPF and DPF)
TS7520	FC 1682 (CPF and DPF)
3583	FC 1680 (CPF), FC 1681 (DPF)
3582	FC 1680 (CPF), FC 1681 (DPF)

Notes:

1. Path failover is not supported on parallel SCSI (except for some parallel SCSI drives on AIX operating systems). Path Failover is only supported on SAS devices attached to Windows and Linux for Intel/AMD-processor based servers.
2. Path failover is not supported on Virtual Tape Libraries (except for those listed in Table 3) that emulate IBM tape devices.

Data Encryption

Tape and Library Requirements

Encryption support is available in the IBM encryption-capable TS1120 and Ultrium LTO4 (Fibre Channel or SAS) tape drives that support encryption. If system-managed or library-managed encryption will be used, the Transparent LTO Encryption feature code is required for the library with Ultrium LTO4 tape drives. This feature code is not required for TS1120 tape drives. Application-managed encryption does not require this feature code for any of encryption-capable tape drives. Refer to “Feature Codes (Encryption)” on page 20 for the feature code numbers.

In the open system environment, there are three methods of encryption management to choose from. These methods differ in where you choose to locate your Encryption Key Manager (EKM) application. Your operating environment determines which is the best for you, with the result that key management and the encryption policy engine may be located in any one of the three environmental layers: application layer, system layer, and library layer.

Application-Managed Tape Encryption

This method is best where operating environments run an application already capable of generating and managing encryption policies and keys, such as Tivoli® Storage Manager (TSM). Policies specifying when encryption is to be used are defined through the application interface. The policies and keys pass through the data path between the application layer and the encryption-capable tape drives. Encryption is the result of interaction between the application and the encryption-enabled tape drive, and is transparent to the system and library layers.

Application-managed encryption is supported on AIX, Windows 2000, Windows Server 2003, Windows Server 2008, Linux, Solaris, and HP-UX. It is required to use the latest device drivers available on the ftp down load site: **ftp://ftp.software.ibm.com/storage/devdrvr/** . Refer to your software vendor for support details.

Please refer to "Planning for Application-Managed Tape Encryption" for details on the hardware and software requirements for application-managed encryption. For details on setting up application-managed tape encryption refer to the Tivoli Storage Manager documentation or visit **http://publib.boulder.ibm.com/infocenter/tivihelp/v1r1/index.jsp** for more information.

System-Managed Tape Encryption

In this method, key generation and management is performed by the EKM, a Java™ application running on the host. Policy controls and keys pass through the data path between the system layer (device drivers) and the encryption-capable tape drives. Encryption is transparent to the applications.

It is required to use the latest device drivers available on the ftp down load site: **ftp://ftp.software.ibm.com/storage/devdrvr/**.

Please refer to "Planning for System-Managed Tape Encryption" on page 17 for details on the hardware and software requirements for system-managed encryption. For details on setting up system-managed encryption in different operating system environment, please refer to the chapter for each operating system/platform.

Library-Managed Tape Encryption

This method is best for encryption-capable tape drives in an open attached IBM tape libraries. Scratch encryption policies specifying when to use encryption are set up through the IBM System Storage Tape Library Specialist Web interface. Policies are based on cartridge volume serial numbers. Key generation and management is performed by the EKM, a Java application running on a host. Policy control and keys pass through the library-to-drive interface, therefore encryption is transparent to the applications.

Library-managed encryption is supported on AIX, Windows Server 2003, Windows Server 2008, Linux, Solaris, and HP-UX. Please refer to "Planning for Library-Managed Tape Encryption" on page 19 for details on the hardware and software requirements for library-managed encryption. For details on setting up library-managed encryption on encryption-capable tape drives, please refer to the IBM System Storage Tape Library Operator's Guide for your library.

Planning for Application-Managed Tape Encryption

Note: Please contact your IBM Representative for additional information about encryption on the IBM encryption-capable tape drive.

In order to perform encryption on the encryption-capable tape drive, the following is required:

- Encryption-capable tape drive(s)
- Encryption configuration features:
  - Library code updates and Transparent LTO Encryption feature code for libraries with Ultrium LTO4 drives
  - Tape drive code updates

Application-Managed Tape Encryption Setup Tasks

Any task not identified as an IBM service task is the responsibility of the customer.

1. Install, cable, and configure the encryption-capable tape drive (refer your IBM System Storage Tape Drive or Library Operator's Guide)
2. Install appropriate IBM tape device driver level (Atape, for example).
3. Set up encryption policies. Refer to the appropriate TSM documentation.
4. Perform write/read operation to test encryption.
5. Verify encryption of the test volume by Autonomic Management Engine (AME): issue QUERY VOLUME FORMAT=DETAILED

Verify that Drive Encryption Key Manager is set to Tivoli Storage Manager.

Planning for System-Managed Tape Encryption

Note: Please contact your IBM Representative for additional information about encryption on the IBM encryption-capable tape drive.

In order to perform encryption on the encryption-capable tape drive., the following is required:

- Encryption-capable tape drive(s)
- Keystore (refer to the *IBM System Storage Encryption Key Manager Introduction, Planning, and User Guide*)
- Encryption configuration features:
  - Encryption Key Manager (EKM)
  - Tape drive code updates and Transparent LTO Encryption feature code for libraries with Ultrium LTO4 drives
  - Tape system library code updates

Setup Tasks for System-Managed Tape Encryption on Open Systems Platforms

Any task not identified as an IBM service task is the responsibility of the customer.

1. Install, verify, and configure:
   a. Keystore
   b. EKM (Refer to the *IBM System Storage Encryption Key Manager Introduction, Planning, and User Guide* for more information on both.)
2. Install, cable, and configure encryption-capable tape drive (refer your IBM System Storage Tape Drive or Library Operator's Guide).
3. Install and enable appropriate Device Driver Level (refer to the appropriate operating system/platform chapter in this document for more details)
4. Edit `/etc/ibmekm.conf` file.
5. Use tapeutil EKM test functions to verify.

System-Managed Encryption Configuration

Note: The tape drive must be set to system-managed encryption from the drive panel or library user interface before using device driver system-managed encryption.

After installing the device driver, then Encryption Key Manager needs to be configured. The servers are configured in a text file "`ibmekm.conf`" that is installed in the /etc directory by the device driver if a current configuration file does not already exist.

Figure 3 is an example of the sample configuration file installed.

```
#  IBM Encryption Key Manager Configuration File
#
#  (C) COPYRIGHT International Business Machines Corp. 2006
#  All Rights Reserved
#  Licensed Materials - Property of IBM
#
#  US Government Users Restricted Rights - Use, duplication or
#  disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
#  This file contains the TCP/IP address(s) and port(s) for the Encryption Key
#  Server with a configuration entry in the following formats. The IPv4 address
#  entered as x.x.x.x:port. The IPv6 address entered as x:x:x:x:x:x:x:x port.
#  The server is for information only and is not used. The timeout value is
#  specified in seconds.
#
#  The format for IPv4 address:
#  server timeout address:port
#  for example,
#  ekmtest  10  9.12.123.1234:8050
#
#  The format for IPv6 address:
#  server timeout address port
#  for example,
#  ekmtest  10  fe80::207:30ee:edcb:d05d  8050
#
#  The Encryption Key Server address and port can be a local loop back
#  address 127.0.0.1:port in IPv4 format or ::1 port in IPv6 format if the server
#  is on the same host or a network address and port if external to the host.
#  Up to 16 server address and port entries are supported if there are multiple
#  TCP/IP connections to the same server and/or multiple servers.
#
# Interoperability between IPv4 and IPv6 versions running on dual-stack hosts:
#    IPv4 Client <--> IPv4/IPv6 Server    using IPv4 address for EKM server
#    IPv6 Client <--> IPv4 Server             using IPv4 address for EKM server
#    IPv6 Client <--> IPv6 Server             using IPv6 address for EKM server
#
#  Sample entry for a local server with a 10 second timeout using port 8050
#  in IPv4 format
#  ekmtest   10   127.0.0.1:8050
#
#  in IPv6 format
#  ekmtest   10   ::1   8050
```

Figure 3. Sample Encryption Configuration File

The following shows the different entry formats for IPv4 and IPv6 addresses in the ibmekm.conf configuration file:

- IPv4 format: "EKMserver timeout IPv4_address:Port_number"
- IPv6 format: "EKMserver timeout IPv6_address Port_number"

To set up an IP address for an EKM server, use Table 4 on page 19 to choose an appropriate IP address type and then add the IP address in the entry of the encryption configuration file.

Table 4. Interoperability between IPv4 and IPv6 Clients and Servers

	IPv4 EKM server IPv4-only host	**IPv6 EKM server IPv6-only host**	**IPv4 EKM server dual-stack host**	**IPv6 EKM server Dual-stack host**
IPv4 Client/IPv4-only host	IPv4	(no)	IPv4	IPv4
IPv6 Client/IPv6-only host	(no)	IPv6	(no)	IPv6
IPv4 Client dual-stack host	IPv4	(no)	IPv4	IPv4
IPv6 Client dual-stack host	IPv4	IPv6	(no)	IPv6

The timeout value in seconds is used when a request is sent to the server and the time for the server to respond in case no response is received.

A maximum of 16 server connections can be configured for failover. When a connection can not be made or is lost on the current server being used, the operation will be retried on the next configured server.

After configuring servers in the `ibmekm.conf` file then specific tape drives need to be configured to the device driver for system-managed encryption. The device driver default does not enable encryption. Refer to the following sections for specific information depending on the O/S.

Planning for Library-Managed Tape Encryption

Note: Please contact your IBM Representative for additional information about encryption on the IBM encryption-capable tape drive.

In order to perform encryption on the encryption-capable tape drive, the following is required:

- Encryption-capable tape drive(s)
- Keystore (refer to the *IBM System Storage Encryption Key Manager Introduction, Planning, and User Guide*)
- Encryption configuration features:
  - Encryption Key Manager (EKM)
  - Tape system library code updates and Transparent LTO Encryption feature code for libraries with Ultrium LTO4 drives.
  - Tape drive code updates

Library-Managed Tape Encryption Tasks

Any task not identified as an IBM service task is the responsibility of the customer.

1. Install, verify, and configure
   a. Keystore
   b. EKM (Refer to the *IBM System Storage Encryption Key Manager Introduction, Planning, and User Guide* for more information on both.
2. Install and cable the encryption-capable tape drive (IBM service task for TS1120 Tape Drive).

3. Use IBM System Storage Tape Library Specialist to enable the tape drive for library-managed tape encryption (refer to your IBM System Storage Tape Drive or Library Operator's Guide).
4. Use library diagnostic functions to verify.

Bulk Rekey

For customers using Library-Managed Encryption with 3592 Enterprise Tape Drives and IBM tape and changer drivers running on open systems platforms (AIX, HP-UX, Linux, Solaris, Windows), sample code for performing bulk rekey operations is available. The sample code packages are provided ″as-is″ with limited testing, and are provided to give customers guidance on bulk rekey operations.

For Unix platforms, a sample script (`rekey_unix.sh`) is provided and must be used in conjunction with the tapeutil version bundled in the same package. For Windows platforms, a sample c program (`rekey_win.c`) is provided. Both of these sample programs must be used in conjunction with both the IBM tape and changer drivers. In addition, data cartridges must be located in storage cells, not in I/O station cells or tape drives.

For more information and to download the sample code packages, see ftp://ftp.software.ibm.com/storage/devdrvr/sample_code.

Feature Codes (Encryption)

In order to use system-managed and library-managed encryption, the Transparent LTO Encryption feature codes listed in Table 5 are required for the associated IBM tape libraries with Ultrium LTO4 tape drives. If the drives in use are TS1120 Tape Drives, this feature code is not required for system-managed or library-managed encryption. If you are using application-managed encryption, no feature code is required on any encryption-capable tape drives.

Table 5. Feature Codes (Encryption)

Tape Library	Feature Code
TS3500	FC 1604
TS3400	Standard Feature
TS3310	FC 5900
TS3100/3200	FC 5900

Chapter 3. AIX Tape and Medium Changer Device Driver

This chapter describes the IBM AIX Enhanced Tape and Medium Changer Device Driver (Atape) for IBM tape devices.

Purpose

The IBM AIX Enhanced Tape and Medium Changer Device Driver is designed specifically to take advantage of the features provided by the IBM tape drives and medium changer devices. The goal is to give applications access to the functions required for basic tape operations (such as backup and restore) and medium changer operations (such as mount and demount the cartridges), as well as to the advanced functions needed by full tape management systems. Whenever possible, the driver is designed to take advantage of the device features transparent to the application. Please note that some independent software vendor (ISV) applications have certain device driver requirements. Before you install the device drivers, please refer to the ISV web site or their support to find out what device drivers should be used for the ISV.

Data Flow

The software described in this chapter covers the AIX Enhanced Device Driver (Atape device driver) and the interface between the application and the tape device. For data flow, refer to Figure 4.

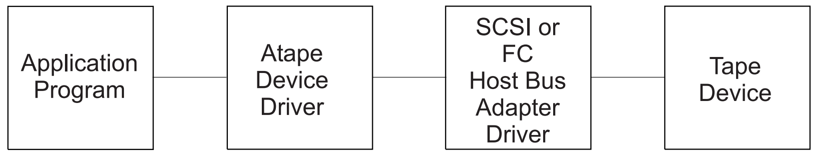


Figure 4. Data Flow for AIX Device Driver (Atape)

Product Requirements

The following hardware and software components are supported by IBM.

Hardware Requirements

The following hardware is supported by the Atape device driver:

- One or more of the following IBM RS/6000® or System p (also known as pSeries) SCSI host bus adapters for libraries containing Ultrium tape drives:
  - PCI-X DDR Dual -x4 SAS Adapter (FC 5900)
  - PCI Dual Channel Ultra-3 SCSI Adapter (LVD) (FC 6203)
  - PCI Differential Ultra SCSI Adapter (HVD) (FC 6207)
  - PCI Universal Differential Ultra SCSI Adapter (HVD) (FC 6204)
  - Integrated Ultra2 SCSI Adapter for LVD attachment (with a VHDCI connector):
    - System p (also known as pSeries) 620 (7025 - 6F0, 6F1)
    - 7025 - F80
    - System p (also known as pSeries) 660 (7026 - 6H0, 6H1, 6M1)
    - System p (also known as pSeries) 640 (7026 - B80)
    - 7026 - H80, M80
    - 7044 - 170, 270
  - Integrated Ultra3 SCSI Adapter for LVD attachment (with a VHDCI connector):
    - System p (also known as pSeries) 610 (7028 - 6C1, 6E1)
    - System p (also known as pSeries) 630 (7028 - 6C4, 6E4
    - System p (also known as pSeries) 650 (7038 - 6M2)
    - System p (also known as pSeries) 655 (7039 - 651)
    - 9112 - 265
  - PCI Dual Channel Ultra-2 SCSI Adapter (LVD) (FC 6205)
  - PCI-X Dual Ultra320 SCSI Blind Swap Adapter (FC 5710)
  - PCI-X Dual Channel Ultra 320 SCSI Adapter (FC 5712)
  - PCI-X Dual Channel Ultra 320 SCSI Adapter (FC 5736)
- One or more of the following IBM RS/6000 or System p (also known as pSeries) FC-AL host bus adapters for TS1120 tape drives and all IBM libraries:
  - Gigabit Fibre Channel Adapter (PCI) (FC 6227)
  - Gigabit Fibre Channel Adapter for 64–bit PCI bus (FC 6228)

  – Two Gigabit Fibre Channel Adapter (PCI—X) (FC 6239)
  – Two Gigabit Fibre Channel Adapter (PCI-X) (FC 5716)
  – Four Gigabit Fibre Channel Adapter (PCI-X) (FC 5758 and FC 5759)
  – Four Gigabit Fibre Channel Adapter (PCI-E) (FC 5773 and FC 5774)
- For the 3490E and Magstar devices, one of the following RS/6000 or System p (also known as pSeries) host adapters:
  – Micro Channel® SCSI Adapters
    - RS/6000 or System p (also known as pSeries) FC 2412 Enhanced SCSI-2 Differential Fast/Wide Adapter/A
    - RS/6000 or System p (also known as pSeries) FC 2416 SCSI-2 Differential Fast/Wide Adapter/A
    - RS/6000 or System p (also known as pSeries) FC 2420 SCSI-2 Differential High Performance External I/O Controller (limited to seven SCSI IDs)
  – PCI SCSI Adapters
    - RS/6000 or System p (also known as pSeries) FC 2409 PCI SCSI-2 Differential Fast/Wide Adapter
    - RS/6000 or System p (also known as pSeries) FC 6209 PCI SCSI-2 Differential Fast/Wide Adapter
    - RS/6000 or System p (also known as pSeries) FC 6207 PCI Differential Ultra SCSI Adapter
    - RS/6000 or System p (also known as pSeries) FC 6204 PCI Universal Differential Ultra SCSI Adapter
- For 733x devices, refer to the specific product document for hardware attachment requirements.
- The IBM TotalStorage Enterprise Tape System 3590 Model E and Model H Tape Drives with Fibre Channel Attachment, is supported with the System p (also known as pSeries) FC 6227 or FC 6228 Gigabit Fibre Channel Adapters and FC6239 2 Gigabit Fibre Channel PCI-X Adapters.
- The following SCSI-attached devices are supported through the IBM 2108-G07 (SAN Data Gateway) attached to the FC 6227 or FC 6228 Gigabit Fibre Channel Adapters and FC6239 2 Gigabit Fibre Channel PCI-X Adapters:

  IBM TotalStorage Enterprise Tape System 3590, Models B11, B1A, E11, E1A, H11, and H1A

  IBM TotalStorage Virtual Tape Server (Models B10, B18, and B20)

  IBM Magstar MP 3570 Tape Subsystem Model C

  IBM Magstar MP 3575 Tape Library Dataserver
- The IBM TotalStorage Enterprise Tape System 3592 Model J is supported with FC6239 2 Gigabit Fibre Channel PCI-X Adapters.

  Four Gigabit Fibre Channel Adapter (PCI-E) (FC 5773 and FC 5774)

Note: Using a single Fibre Channel host bus adapter (HBA) for concurrent tape and disk operations is generally not recommended. Tape and disk devices require incompatible HBA settings for reliable operation and optimal performance characteristics. Under stress conditions (high I/O rates for either tape, disk, or both) where disk and tape subsystems share a common HBA, stability problems have been observed. These issues are resolved by separating disk and tape I/O streams onto separate HBAs and using SAN zoning to minimize contention. IBM is focused on assuring server/storage configuration inter-operability. We strongly recommend that your overall implementation plan includes provisions for separating disk and tape workloads.

Software Requirements

The AIX Enhanced Device Driver (Atape device driver) supports AIX 5L™ Version 5.2 and later releases on IBM POWER-based AIX servers.

To obtain the most current service and documentation for this software, refer to Appendix A, "Accessing Documentation and Software Online," on page 305.

Installation and Configuration Instructions

The recommended procedure for installing a new version of the device driver is to uninstall the previous version.

Instructions for uninstalling the device driver are outlined below in "Uninstalling" on page 26.

Attention: At the end of the installation procedure, the *installp* facility will automatically run the AIX *bosboot* command to update the boot record with the newly installed Atape files. When the *bosboot* command completes, the following messages will be displayed:

0503-292 This update will not fully take effect until after a system reboot.
installp: bosboot process completed.

This message is referring to the updates to the boot record only. If the installation summary shows that the Atape driver was installed successfully, it is **not** necessary to reboot the machine at this time.

If the installation summary shows that the install failed, you should reboot the machine and attempt to install the Atape driver a second time.

Installation Procedure

For information on obtaining the latest version of device drivers and the latest documentation, refer to Appendix A, "Accessing Documentation and Software Online," on page 305.

Preinstallation Considerations

Before proceeding with the installation, verify the following items:

1. The tape device is properly functioning, properly attached to the server, and is powered up. If you are attaching an IBM TotalStorage Enterprise Tape System 3590 Model E or Model H Tape Drive with the Fibre Channel Attachment, make sure it is defined with the hard addressing option.
2. You have logged onto the server on an account which has *root* authority.
3. You have a command shell window open on the server to perform the installation procedure.
4. Make sure the current path is defined in the command shell PATH variable. This can be accomplished in the korn shell using the following command:
   ```
   EXPORT PATH=.:$PATH
   ```

5. If the tape device was configured previously by another device driver (not Atape), remove any existing device definitions for it. For example, ’rmdev -l ost1 -d’.

Installation Procedure

Enter the following command to list the currently installed *Atape.driver* version:

```
lslpp -l Atape.driver
```

Enter the following command to install the Atape driver in the current directory for example

```
installp -acXd Atape.x.x.x.x Atape.driver
```

This will install and commit the Atape driver on the system.

Configuring Tape and Medium Changer Devices

After the driver software is installed and a tape device is connected to the adapter, the device can be configured and made available for use. Access to the device is not provided until the device is configured.

Note: If the tape device was configured previously by another SCSI device driver, such as OST (Other SCSI Tape), issue the following command to remove the device definition before performing the following steps:

```
rmdev -l [device]
```

Configure a tape device using **one** of the following procedures:

- Enter the following command with no parameters:

  ```
  cfgmgr
  ```

  The command configures all devices automatically (including any new tape or medium changer devices).

  or
- Power Off your subsystem and reboot the system to configure it automatically and make available any new tape or medium changer devices on the system.

Deconfiguring Tape Devices

Note: In the following examples, replace the letter *n* with the appropriate number for the chosen device.

Deconfigure the tape device using **one** of the following procedures:

1. The first method leaves the tape device defined in the configuration database. It is similar to bringing the device offline (not in use).

   Enter the following command to bring the */dev/rmtn* tape device offline, but leave it defined in the device database:

   ```
   rmdev -l rmtn
   ```
2. The second method brings the tape device offline and removes its definition from the device database.

   Enter the following command:

   ```
   rmdev -l rmtn -d
   ```

   The device driver is not unloaded from the kernel until the last device is deconfigured.

Deconfiguring Medium Changer Devices

Note: In the following examples, replace the letter *n* with the appropriate number for the chosen device.

Deconfigure the medium changer device using **one** of the following procedures:

1. The first method leaves the device defined in the configuration database. It is similar to bringing the device offline.

   Enter the following command to bring the */dev/smcn* medium changer device offline, but leave it defined in the device database:

   ```
   rmdev -l smcn
   ```
2. The second method brings the medium changer device offline and removes its definition from the device database.

   Enter the following command:

   ```
   rmdev -l smcn -d
   ```

   The device driver is not unloaded from the kernel until the last device is deconfigured.

Uninstalling

Attention: All devices that use the Atape driver must be closed and cannot be in use when Atape is uninstalled or the uninstall will fail.

You can uninstall the Atape device driver using the *smit* command menu to uninstall software and selecting Atape.driver or using the following *installp* command:

```
installp -u Atape.driver
```

Tape Drive, Media, and Device Driver Parameters

This chapter describes the parameters that control the operating modes of the AIX Enhanced Tape and Medium Changer Device Driver.

Configuration Parameters

The operating parameters for the tape drive and device driver are set and changed by configuration parameters. The installation defaults are provided for all parameters initially. The AIX *smit* command is used to set these parameters when configuring a device or to change these parameters. The AIX *chdev* command is used to change the configuration parameters.

The configuration parameters are used to set the operating mode of the tape drive and device driver when a device is opened. These parameters are queried by an application. Some parameters can be temporarily changed during the *open* subroutine by an application, but they are always restored to the configuration values when a device is closed. The configuration parameters are:

- Alternate Pathing
- Autoloading
- Emulate autoloader (359x devices only)
- Block size
- Buffered mode (359x devices only)
- Compression

- Fail degraded media (359x devices only)
- Logging
- Maximum size of the log file
- New logical name
- Read error recovery time (359x devices only)
- Record space mode
- Rewind immediate
- Trailer labels

Alternate Pathing

This parameter enables or disables the path failover support when a device is configured. "Data Path Failover and Load Balancing Support for Tape Drives" on page 37 for a description of the path failover and failover support.

The installation default is Off (path failover is not enabled).

Autoloading

This parameter enables the autoloading feature of the device driver. It is used with the autoloading capability of the autoloader, ACF, ACL, or CSL installed on the tape device.

Note: The autoloading feature is not supported on the IBM 3584 UltraScalable Tape Library and the IBM 3583 Ultrium Scalable Tape Library with more than one IBM 3580 Ultrium Tape Drive installed.

Note: The autoloading feature is supported only on the following device types and configurations:
- IBM 3490E Models C11, C22, E01, E11, F01, and F11
- IBM TotalStorage Enterprise Tape System 3590, Models B11, E11, and H11
- IBM Magstar MP 3570 Models B01, C01, B11, and C11
- IBM Magstar MP 3570 Models B02, B12, C02, and C12 (configured in split mode only)
- IBM 7332 (all models)

Do not enable autoloading if one of the following conditions is true:
- The device is used by an application that provides library medium changer support for the IBM 3581 or IBM 3583.
- The device is installed in a 3494 Enterprise Tape Library.
- The device is used by an application with stack loader support.
- The application is MKSYSB.
- The tapes being read were not written using the autoloading feature.

Tapes created with AUTOLOAD=YES may not be readable in configurations without Atape autoload enabled, or on other UNIX® platforms, or on device types/models that are different from the backup device type/model.

If the parameter is set to on, then the tape stacker acts as one large virtual tape. During a read, write, or forward space file operation, no end of tape is detected by the application. When the end of tape is reached, the device driver automatically rewinds and unloads the tape, loads the next tape, then continues reading or writing the next tape. The following conditions are required to use this feature:
- The autoloading parameter must be set to On.

- The cartridge stacker must be loaded with one or more tapes.
- The ACF, ACL, or CSL must be set to Automatic, System, or Random mode.

This feature allows multivolume backups (with commands such as *tar*) without prompting for a volume change.

The installation default is Off (no autoloading).

Emulate Autoloader

This parameter controls how the device driver operates when the ACF on the IBM TotalStorage Enterprise Tape System 3590, the IBM Magstar MP tape device, or the IBM 3490E Model Fxx is set to *Random* mode. If this parameter is set to On and the ACF is in Random mode, the device driver emulates an autoloading tape drive. When an unload command is sent to the device driver to unload a tape, the tape is unloaded, returned to the magazine, and the next tape in the magazine is loaded automatically into the tape drive. If this parameter is set to Off, the normal unload operation occurs, and the tape remains in the drive.

The emulate autoloader parameter can be used for legacy applications written for the IBM 3490E Automated Cartridge Loader (ACL) when the IBM TotalStorage Enterprise Tape System 3590, the IBM Magstar MP 3570, or the IBM 3490 Model F autoloader is set to *Random* mode. This eliminates the need to reconfigure the device's autoloader for *Random* or *Automatic* operation.

The installation default is Off (do not emulate autoloader).

Note: On IBM Magstar MP 3570 Models B02, C02, and C12, this feature is supported only when the two drives are configured in *Split* mode, or in *Base* mode with only one drive configured and available to AIX. This feature does not work in Base mode if both drives are in the available state to AIX.

Block Size

This parameter specifies the block size used for read and write operations. A value of zero is the variable block size. Any other value is a fixed block size.

The installation default is zero (use variable length) **except** for the IBM 7332 4mm Tape Cartridge Autoloader, for which the default is a fixed block size of 1024 bytes.

Buffered Mode

When a write command is processed, the data is either stored directly on the physical tape or buffered in the tape device. Buffering can increase the device performance.

The installation default is On (use Buffered mode).

Compression

Hardware compression is implemented in the device hardware. This parameter turns the compression feature On and Off. If compression is enabled, then the effective performance can increase based on the compressibility of the data.

The installation default is On (use compression).

Fail Degraded Media

This parameter controls whether the device driver fails a tape operation when degraded media is detected by the IBM TotalStorage Enterprise Tape System 3590. If a tape is loaded and the IBM 3590 cannot read the positioning information from the tape, the device driver is notified when the first command is sent to the tape drive. If this parameter is set to On, the device fails the command and returns a media error to the application. If this parameter is set to Off, the device driver does not fail the command.

Degraded media is a correctable condition that prevents the IBM TotalStorage Enterprise Tape System 3590 from performing high speed *Locate* operations. A *Locate* command can take over 20 minutes, depending on the desired position and the amount of data on the tape. This parameter is intended for use by real time applications that cannot tolerate long *Locate* commands.

The installation default is Off (do not fail the tape operation if degraded media is detected).

Logging

This parameter turns the volume information logging on and off. If logging is set to On, the statistical information about the device and media is saved in a log file when a tape is unloaded. If logging is set to Off, the information is not saved. This parameter has no effect on error logging because error logging is always enabled. For more information, refer to “Device and Volume Information Logging” on page 52.

The installation default is Off (no logging).

Maximum Size of the Log File

This parameter specifies the number of entries made before the log file starts to wrap. Each entry is approximately 2 KB (2048 bytes). After the log file starts to wrap, the number of entries stays constant. Each time a new entry is made, the oldest entry is overlaid. For more information, refer to “Device and Volume Information Logging” on page 52.

The installation default is 500.

New Logical Name

Setting this parameter changes the logical name of the device to a new name as specified. After the logical name is changed, the new logical name parameter is cleared.

There is no installation default value for this parameter.

Read Error Recovery Time

This parameter controls the read error recovery time for the IBM TotalStorage Enterprise Tape System 3590. If this parameter is set to On, the recovery time for read errors is limited to a maximum of 5 seconds. If this parameter is set to Off, full recovery is used by the device and can take up to 10 minutes. This parameter is intended for use by real-time applications that cannot tolerate long delays when reading data from the tape.

The installation default is Off (do not limit the read error recovery time).

Record Space Mode

This parameter specifies how the device driver operates when a forward or backward space record operation encounters a filemark. The two modes of operation are SCSI and AIX.

The SCSI mode is the default mode of operation. When a forward or backward space record operation is issued to the driver and a filemark is encountered, the device driver returns -1 and the *errno* variable is set to input/output error (EIO). The tape is left positioned after the filemark (the end-of-tape side of the filemark on the forward space and the beginning-of-tape side of the filemark on the backward space).

The AIX mode returns the same EIO *errno* value as the SCSI mode when a filemark is encountered except that the tape is left positioned before the filemark (the beginning-of-tape side of the filemark on the forward space and the end-of-tape side of the filemark on the backward space).

The installation default is SCSI mode.

Rewind Immediate

This parameter turns the immediate bit On and Off in rewind commands. If it is set to On, then the rewind tape operation executes faster, but the next command takes a long time to finish unless the rewind operation is physically complete. Setting this parameter reduces the amount of time that it takes to close a device for a Rewind on Close special file.

The installation default is Off (no rewind immediate) **except** for the IBM 7332 4mm Tape Cartridge Autoloader, for which the default is On (rewind immediate).

Trailer Labels

If this parameter is set to On, then writing a record past the early warning mark on the tape is allowed. The first write operation to detect EOM fails, and the *errno* variable is set to ENOSPC. No data is written during the operation. All subsequent write operations are allowed to continue until the physical end of the volume is reached and EIO is returned.

This parameter can also be selected using one of three device special files that allow trailer-label processing. The special files are *rmtx.40*, *rmtx.41*, and *rmtx.60*, where *x* is the name of the device (for example, *rmt0.40*).

The installation default is Off (no trailer labels).

Media Parameters

The media parameters can be queried and set by the *tapeutil* application using the **Query/Set Parameters** option in the window. These parameters cannot be set or changed by the configuration procedures. The media parameters are:

- Capacity scaling
- Logical write protect
- Volume ID for logging

Capacity Scaling

This parameter sets the capacity or logical length of the current tape on IBM TotalStorage Enterprise Tape System 3590, IBM TotalStorage Enterprise Tape System 3592, or Magstar MP tape subsystems. By reducing the capacity of the tape, the tape drive can access data faster at the expense of data capacity. Capacity scaling can be set at 100% for the entire tape (which is the default) or set at 75%, 50%, or 25% of the tape or any device specific hex value. For example on IBM 3592, to change capacity scaling from a 300 GB format tape (100%) to a 60 GB format tape, select the capacity scaling option, then select the option to enter a hex value and enter 35. Capacity scaling remains with the tape across mounts until it is changed.

Notes:

1. The tape position must be at the start of the tape to change this parameter from its current value.
2. Changing this parameter destroys any existing data on the tape.
3. Attempting to set capacity scaling that is not supported by a device or the current media loaded will always return 100% and cannot be changed. For example, 60 GB media for the IBM 3592 cannot be capacity scaled and will always be 100%.

Logical Write Protect

This parameter sets or resets the logical write protect of the current tape on IBM TotalStorage Enterprise Tape System 3590, IBM TotalStorage Enterprise Tape System 3592, or Magstar MP tape subsystems. The three types of logical write protect are: associated protect, persistent protect, and write-once read-many (WORM) protect.

Associated protect remains only while the current tape is mounted or associated with the tape drive. It is reset when the tape is unloaded or the tape drive is reset.

Persistent protect remains or persists with the tape across mounts until it is reset.

WORM protect also remains with the tape across mounts, but (unlike persistent protect) it cannot be reset on the tape. After a tape is WORM protected, it can never be written on again.

Notes:

1. The tape position must be at the start of the tape to change this parameter from its current value.
2. Attempting to set logical write protect that is not supported by a device or the current media loaded will always return ″No″ and cannot be changed.

Volume ID for Logging

This parameter is the volume ID of the current loaded tape. It is used in the log file entry (if volume logging is active) to identify the entry with a particular volume. The device driver sets the volume ID to UNKNOWN initially and when the tape is unloaded.

Automatic Cartridge Facility Mode

If the IBM TotalStorage Enterprise Tape System 3590 or Magstar MP tape subsystem has an ACF installed, then the ACF mode can be queried by the *tapeutil* application using the **Query/Set Parameters** option in the window.

Special Files

Once the driver is installed and a tape device is configured and available for use, access is provided through the special files. These special files, which consist of the standard AIX special files for tape devices (with other files unique to the Atape driver), are in the */dev* directory.

Special Files for Tape Devices

Each tape device has a set of special files that provides access to the same physical drive but to different types of functions. As shown in Table 6, in addition to the tape special files, a special file is provided for tape devices that allows access to the medium changer as a separate device.

Note: The asterisk (*) represents a number assigned to a particular device (such as *rmt0*).

For tape drives with attached SCSI medium changer devices, the *rmt*.smc* special file provides a separate path for issuing commands to the medium changer. When this special file is opened, the application can view the medium changer as a separate SCSI device.

Both this special file and the *rmt** special file can be opened at the same time. The file descriptor that results from opening the *rmt*.smc* special file does not support the following operations:

- Read
- Write
- Open in Diagnostic mode
- Commands designed for a tape drive

If a tape drive has a SCSI medium changer device attached, then all operations (including the medium changer operations) are supported through the interface to the *rmt** special file. For detailed information, refer to Table 6.

Table 6. Special Files for Tape Devices

Special File Name	Rewind on Close[1]	Retension on Open[2]	Bytes per Inch[3]	Trailer Label	Unload on Close
/dev/rmt*	Yes	No	N/A	No	No
/dev/rmt*.1	No	No	N/A	No	No
/dev/rmt*.2	Yes	Yes	N/A	No	No
/dev/rmt*.3	No	Yes	N/A	No	No
/dev/rmt*.4	Yes	No	N/A	No	No
/dev/rmt*.5	No	No	N/A	No	No
/dev/rmt*.6	Yes	Yes	N/A	No	No
/dev/rmt*.7	No	Yes	N/A	No	No
/dev/rmt*.10[4]	No	No	N/A	No	No
/dev/rmt*.20	Yes	No	N/A	No	Yes
/dev/rmt*.40	Yes	No	N/A	Yes	No
/dev/rmt*.41	No	No	N/A	Yes	No
/dev/rmt*.60	Yes	No	N/A	Yes	Yes
/dev/rmt*.null[5]	Yes	No	N/A	No	No
/dev/rmt*.smc[6]	N/A	N/A	N/A	N/A	N/A

Table 6. Special Files for Tape Devices (continued)

Special File Name	Rewind on Close[1]	Retension on Open[2]	Bytes per Inch[3]	Trailer Label	Unload on Close

Notes:
1. The Rewind on Close special files write filemarks under certain conditions before rewinding.
2. The Retension on Open special files rewind the tape on open only. Retensioning is not performed because these tape products perform the retension operation automatically when needed.
3. The Bytes per Inch options are ignored for the tape devices supported by this driver. The density selection is automatic.
4. The *rmt*.10* file bypasses normal close processing, and the tape is left at the current position.
5. The *rmt*.null* file is a pseudo device similar to the */dev/null* AIX special file. The input/output control (*ioctl*) calls can be issued to this file without a real device attached to it, and the device driver returns a successful completion. Read and write system calls return the requested number of bytes. This file can be used for application development or debugging problems.
6. The *rmt*.smc* file can be opened independently of the other tape special files.

Special Files for Medium Changer Devices

After the driver is installed and a medium changer device is configured and made available for use, access to the robotic device is provided through the *smc** special file in the */dev* directory.

Table 7 shows the attributes of the special file. The asterisk (*) represents a number assigned to a particular device (such as *smc0*). The term *smc* is used for a SCSI medium changer device. The *smc** special file provides a path for issuing commands to control the medium changer robotic device. For more information, refer to Table 7.

Table 7. Special Files for Medium Changer Devices

Special File Name	Description
/dev/smc*	Access to the medium changer robotic device
/dev/smc*.null	Pseudo medium changer device

Note: The *smc*.null* file is a pseudo device similar to the */dev/null* AIX special file. The commands can be issued to this file without a real device attached to it, and the device driver returns a successful completion. This file can be used for application development or debugging problems.

The file descriptor that results from opening the *smc* special file does not support the following operations:
- Read
- Write
- Commands designed for a tape device

Persistent Naming Support

Persistent naming support is used to ensure that attached devices will always be configured with the same logical name based on the SCSI ID, LUN ID, and host bus adapter (HBA), even when the system is rebooted.

When the AIX operating system is booted, the HBA performs a device discovery and assigns a default logical name to each device found in a sequential order. For example, if there are three tape drives attached to a parallel SCSI adapter, each with a LUN ID of 0 and a target address of 0, 1, and 2 respectively, the HBA would initially configure them as Available with the following logical names:

rmt0	target 0, lun 0	Available
rmt1	target 1, lun 0	Available
rmt2	target 2, lun 0	Available

On the next reboot, for example, if the existing rmt1 target 1 device is powered off or not connected, the HBA would initially configure two devices as Available with the following logical names:

rmt0	target 0, lun 0	Available
rmt1	target 2, lun 0	Available

If the previous rmt1 target 1 device is powered on after reboot and the cfgmgr command is run, the HBA will configure the device as rmt2 instead of rmt1:

rmt2	target 1, lun 0	Available

This is one example, but there are other cases where the logical names of devices could change when the system is rebooted. For applications that need a consistent naming convention for all attached devices, this is accomplished with persistent naming support by defining a unique logical name (other than the AIX default names) that are associated with the specific SCSI ID, LUN ID, and HBA that the device is connected to.

Changing the logical name after initial boot

The logical name of a device can be changed after an initial boot and configured. This can be done using the SMIT menu or the chdev command from a script or command line.

For example, a default rmt0 logical name for a tape drive could be changed to rmt-0, tape0, or any descriptive name desired. In this example, if the three tape drives are changed to rmt-0, rmt-1, and rmt-2 respectively, and the system is then rebooted with rmt-1 powered off, the HBA would detect that there are unique names predefined for the attached devices and the HBA will use those names. In this case, the devices would configure as follows:

rmt-0	target 0, lun 0	Available
rmt-1	target 1, lun 0	Defined
rmt-2	target 2, lun 0	Available

Since rmt-1 is not detected by the HBA but has been predefined at the SCSI ID and LUN ID, it remains in the defined state and is not configured for use, but the next rmt-2 tape drive configures as the same name at the same location after reboot.

Using SMIT to change the logical name

To change the logical name using SMIT, perform the following steps:

1. Run SMIT from a command line and select Devices.
2. Select **Tape Drive**.
3. Select **Change/Show Characteristics of a Tape Drive**.
4. Select the logical device to be changed from the list displayed.
5. In the New Logical Name field, enter a non-AIX default logical name.

6. Press Enter to process the change.

Using chdev command to change the logical name

To change the logical name using the chdev command, run `chdev –l rmt0 –a new_name=rmt-0` (this example performs a logical name change from rmt0 to rmt-0). The output should display rmt0 changed.

Note: When path failover is enabled, changing the logical name for either a primary or alternate device only changes that individual device name.

Control Path Failover Support for Tape Libraries

Note: The library control path failover feature code must be installed prior to enabling the path failover support in the Atape device driver. Refer to “Supported Devices and Feature Codes” on page 14 for what feature codes may be required for your machine type.

The Atape device driver path failover support will configure multiple physical control paths to the same logical library within the device driver and provide automatic failover to an alternate control path when a permanent error occurs on one path. This is transparent to the running application.

Configuring and Unconfiguring Path Failover Support

Path failover support is not enabled automatically when the device driver is installed. It must be configured initially on each logical device after installation. When path failover support is enabled for a logical device, it remains set until the device is deleted or the support is unconfigured. The alternate pathing setting is retained even if the system is rebooted.

To enable or disable the support on a single logical device, use the smit menu to Change/Show Characteristics of a Tape Drive, select the logical device to change such as smc0, smc1, and so on, then select Yes or No for Enable Path Failover Support. The support can also be enabled or disabled using the chdev command, for example:

```
chdev -l smc0 -aalt_pathing=yes
chdev -l smc1 -aalt_pathing=yes
chdev -l smc0 -aalt_pathing=no
chdev -l smc1 -aalt_pathing=no
```

Primary and Alternate Paths

When the device driver configures a logical device with path failover support enabled, the first device configured always becomes the primary path. On SCSI attached devices, -P is appended to the location field and on Fibre attached devices -PRI is appended to the location field of the device.

When a second logical device is configured with path failover support enabled for the same physical device, it configures as an alternate path. On SCSI attached devices, -A is appended to the location field and on Fibre attached devices -ALT is appended to the location field of the device. A third logical device is also configured as an alternate path with either -A or -ALT appended, and so on. The device driver supports up to 16 physical paths for a single device.

The labeling of a logical device as either a primary or alternate path is for information only, in order to:

1. Be able to identify the actual number of physical devices configured on the system and a specific logical device associated with them. There will be only one logical device labeled as the primary path for each physical device. However, there can be many (multiple) logical devices labeled as an alternate path for the same devices.
2. Provide information about which logical devices configured on the system have path failover support enabled.

Querying Primary and Alternate Path Configurations

You can display the primary and alternate path configuration for all devices with the lsdev command. There can be two or more logical devices configured for a single physical device, but the first device configured is labeled the primary device. All other logical devices configured after the first device are labeled as alternate devices. To see this, run the lsdev -Cc tape command and look at the location field in the data. By running

```
lsdev -Cc tape | grep P
```

, for example, you can easily determine how many physical devices are configured with path failover support.

You can display the primary and alternate path configuration for any device by running:

```
tapeutil -f/dev/smcx path (where smcx is the logical name of any device)
```

This command shows specific information for the primary path and all alternate paths, such as the logical name of the device, SCSI IDs, the current enabled status, and how many paths are configured for the device.

Configuring and Unconfiguring Primary and Alternate Devices

Logical devices configured as alternate paths can be unconfigured and reconfigured at any time after the initial configuration is run. Unconfiguring an alternate path device removes that device from the primary device path list, removes the -A or -ALT appended to the location field, and changes the device to the Defined state. The primary and any other alternate devices are still available.

Likewise, configuring a new alternate path device or reconfiguring an existing one in the Defined state adds that device to the primary device path list, appends -A or -ALT to the location field, and makes the device available.

Logical devices that are configured as primary paths can also be unconfigured and reconfigured at any time after initial configuration is run. However, the operation is different for a primary device. When a primary device is unconfigured, the following events occur:

1. All alternate devices are unconfigured as described previously.
2. The primary device is unconfigured.
3. The -P or -PRI appended to the location field is removed.
4. The device is changed to the Defined state.
5. All alternate devices that were unconfigured are reconfigured. The first device that is reconfigured becomes the new primary device. All remaining alternate devices are reconfigured as alternate paths.

These methods provide the ability to unconfigure and reconfigure physical devices on the system when device connections or addressing changes are made.

Data Path Failover and Load Balancing Support for Tape Drives

Notes:

1. Some devices require a path failover feature code installed prior to enabling the path failover support in the Atape device driver. Refer to “Supported Devices and Feature Codes” on page 14 for what feature code may be required for your machine type.
2. DPF keys do not need to be added if you are running the latest drive code on Ultrium-3 and Ultrium-4 drives.
3. This function is not supported for devices that are attached through an IBM San Data Gateway or on the IBM Virtualization Engine TS7510.
4. The AIX operating system only supports a static configuration of devices, which also applies to the Path Failover and Failover Support. When devices are initially configured at a specific SCSI ID and physical connection (drive port, host bus adapter, and switch number/port, if applicable) and in the Available state, changing the physical device address/connection without either rebooting or unconfiguring and reconfiguring the devices has unpredictable results and is not supported.

Installing Data Path Failover License Key

Use the following command line script to query, add, or delete license keys for this feature before enabling the path failover feature as described below. The key is a 16-digit hexadecimal value, for example, 1234567890abcdef.

All key values “A-F” should be entered in lowercase letters as “a-f.”

- Query installed keys: dpf_keys
- Install a license key: dpf_keys -a key
- Delete a license key: dpf_keys -d key

Configuring and Unconfiguring Path Failover Support

Path failover support is not enabled automatically when the device driver is installed. It must be configured initially on each logical device after installation. When path failover support is enabled for a logical device, it remains set until the device is deleted or the support is unconfigured. The path failover setting is retained even if the system is rebooted.

Path failover support can be enabled on all configured devices at one time, or it can be enabled or disabled selectively by logical device. It may be desirable at times to configure some, but not all, logical paths to a device with the support enabled.

To enable the support globally on all currently configured devices, run the command:

```
/usr/lpp/Atape/instAtape -a
```

This will unconfigure all devices that have path failover set to No, and will reconfigure all devices, setting path failover to Yes.

To enable or disable the support on a single logical device, use the *smit* menu to **Change/Show Characteristics of a Tape Drive**, then select Yes or No for **Enable Path Failover Support**. The support can also be enabled or disabled using the *chdev* command, for example:

```
chdev -l rmt0 -aalt_pathing=yes
```

```
chdev -l rmt0 -aalt_pathing=no
```

Primary and Alternate Paths

When the device driver configures a logical device with path failover support enabled, the first device configured always becomes the primary path and PRI is appended to the location field of the device. When a second logical device is configured with path failover support enabled for the same physical device, it configures as an alternate path and ALT is appended to the location field. A third logical device is configured as the next alternate path with ALT appended, and so on. The device driver supports up to 16 physical paths for a single device.

For example, if *rmt0* is configured first, then *rmt1*, the *lsdev -Cc* tape command output will be similar to the following:

```
rmt0 Available 20-60-01-PRI IBM 3590 Tape Drive and Medium Changer (FCP)
rmt1 Available 30-68-01-ALT IBM 3590 Tape Drive and Medium Changer (FCP)
```

If *rmt1* is configured first, then *rmt0*, the command output will be similar to the following:

```
rmt0 Available 20-60-01-ALT IBM 3590 Tape Drive and Medium Changer (FCP)
rmt1 Available 30-68-01-PRI IBM 3590 Tape Drive and Medium Changer (FCP)
```

The labeling of a logical device as either a primary or alternate path is for information only, in order to:

1. Be able to identify the actual number of physical devices configured on the system and a specific logical device associated with them. There will be only one logical device labeled the primary path for each physical device. However, there may be many (multiple) logical devices labeled as an alternate path for the same devices.
2. Provide information about which logical devices configured on the system have path failover support enabled.

Querying Primary and Alternate Path Configuration

You can display the primary and alternate path configuration for all devices with the *lsdev* command. There may be two or more logical devices configured for a single physical device, but the first device configured is labeled the primary device. All other logical devices configured after the first device are labeled as alternate devices. To see this, run the *lsdev -Cc tape* command and look at the location field in the data. By running *lsdev -Cc tape | grep PRI*, for example, you can easily determine how many physical devices on the RS/6000 or System p (also known as pSeries) server are configured with path failover support. You can display the primary and alternate path configuration for a single device by running the *tapeutil -f/dev/rmtx path* command (where *rmtx* is the logical name of any device).

This command shows specific information for the primary path and all alternate paths, such as the logical name of the device, SCSI IDs, the current enabled status, and how many paths are configured for the device.

Configuring and Unconfiguring Primary and Alternate Devices

Logical devices configured as alternate paths can be unconfigured and reconfigured at any time after the initial configuration is run. Unconfiguring an alternate path device removes that device from the primary device path list, removes the *ALT* appended to the location field, and changes the device to the

Defined state. The primary and any other alternate devices are still available. Likewise, configuring a new alternate path device or reconfiguring an existing one in the Defined state adds that device to the primary device path list, appends *ALT* to the location field, and makes the device available.

Logical devices that are configured as primary paths can also be unconfigured and reconfigured at any time after initial configuration is run. However, the operation is different for a primary device. When a primary device is unconfigured, the following events occur:

1. All alternate devices are unconfigured as described previously.
2. The primary device is unconfigured.
3. The PRI appended to the location field is removed.
4. The device is changed to the Defined state.
5. All alternate devices that were unconfigured are reconfigured. The first device that is reconfigured becomes the new primary device. All remaining alternate devices are reconfigured as alternate paths.

These methods provide the ability to unconfigure and reconfigure physical devices on the system when device connections or addressing changes are made.

System-Managed Encryption

Device Driver Configuration

System-Managed Encryption can be set on a specific tape drive using the standard SMIT panels to Change/Show Characteristics of a tape device or the command line chdev command. There are 2 new attributes added for encryption:

```
sys_encryption "yes/no"              Use System Encryption FCP Proxy Manager
wrt_encryption "off/on/custom"       System Encryption for Write Commands at BOP
```

The sys_encryption attribute enables device driver system-managed encryption for a tape drive by setting the value to yes.

The wrt_encryption attribute controls if the device driver will set the tape drive to encryption enabled for write commands. When set to no, the tape drive will use encryption for read operations and write operations will not use encryption. When set to yes, the tape drive will use encryption for both read/write operations. When set to custom, the device driver will not modify current tape drive setting. The custom setting is intended for applications using system-managed encryption to control write encryption without device driver intervention.

Note: If wrt_encryption is set to on, an application can not open a tape drive using the append mode.

Querying Tape Drive Configuration

There are is a new tapeutil command added for encryption to query tape drive encryption settings.

The current tape drive encryption settings can be queried using the tapeutil menu option 38 “Get Drive Encryption Settings” or the command line “tapeutil –f/dev/name encryption”.

Following is an example of a correct tape drive configuration with wrt_encryption set to “on”. If the wrt_encryption attribute is set to “no”, the tape drive “Encryption State” will be Off.

```
>tapeutil -f/dev/rmt1 encryption

Getting drive encryption settings...
Encryption settings:

  Drive Encryption Capable.... Yes
  Encryption Method........... System
  Encryption State............ On
```

Testing Data Encryption Configuration and Connectivity

There is a new tapeutil command added to validate the `ibmekm.conf` file server entries and test tape drive to server connectivity operations. This test can be run using the tapeutil menu option 40 “Data Encryption Test” or the command line “tapeutil –f/dev/name ekmtest”.

The first test checks the server configuration defined in the `ibmekm.conf` file and then communication to the configured servers. This test reports back the number of servers available. The second test runs a basic diagnostic that checks the tape drive to server communication and reports success or fail. The third test runs an enhanced diagnostic that checks a key operation between the tape drive and server then reports success or fail.

Following is an example of a successful test:

```
> tapeutil –f /dev/rmt2 ekmtest

Testing server configuration and connections...
Test complete, servers available 2
Running basic drive to server encryption test...
Test complete, completion code 0
Running full drive to server encryption test...
Test complete, completion code 0
>
```

If the first server test fails with one of the following errors, then the remaining tests will not be run. Perform the recommended problem determination in the order below to resolve the problem.

1. “Invalid argument or Operation not supported on device” - System encryption is not configured for the tape drive in the device driver. Use SMIT to set system encryption to “yes” for the tape drive. If system encryption can not be set then the tape drive is not encryption capable.
2. “Can’t assign requested address” - Either the `ibmekm.conf` file is missing or is invalid. Check and correct.
3. “Network is down” – All configured servers in the `ibmekm.conf` file are not available. Check all servers configured in the `ibmekm.conf` file are currently running on the server IP address and configured correctly.

Error Logging

Encryption errors are logged along with other tape operation errors using the standard TAPE_ERR1 Template “Tape Operation Error” with associated sense data in the detail data.

An encryption failure is indicated when the asc/ascq in the sense data is EFxx or EExx. Refer to the tape drive hardware reference for specific information on the asc/ascq being reported. The asc/ascq can be found in the first column of the second row in detail sense data. For example:

```
Detail Data
SENSE DATA
0A00 0000 5A08 25FF 0000 00FF FF00 0000 0000 0000 F000 0600 0000 1458 0000 0000
EF11 FF02 D105 0000 0009 0191 0002 0000 0000 0A00 0000 0000 0000 0000 0000 0000
0000 0000 0000 FFFF FF00 0000 FFF0 B7E3 0001 2127 0000 0000 0000 0000 3930 3220
2020 2000 0041 4A00 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
```

Field Support Information

When encryption failures require field support and/or development analysis the following data should be provided for a problem on a specific tape drive from the machine (rmt1 for example) for the device driver. Tape drive dumps and EKM server logs may be needed in addition to this information.

```
errpt –a > errpt.rmt1
lsattr –El rmt1 > lsattr.rmt1
tapeutil –f /dev/rmt1 encryption > encryption.rmt1
All Atape files in /var/adm/ras/Atape*
```

Tape Utility Program (tapeutil)

Installed with the device driver is a tape utility program, *tapeutil*, that exercises or tests the functions of the tape device and the device driver. It also performs basic tape and medium changer operations. The tape utility program provides two versions (the interactive interface and the AIX command-line interface) with syntax similar to the *tctl* and *mt* commands.

The C source code for the *tapeutil.c* program can be found in the */usr/lpp/Atape/samples* directory. The program contains a sample of the interface to the device driver and the *ioctl* commands supported.

Note: This utility program does not support the data path for the IBM 7331, 7336, and 7337 library devices. It can be used only for the medium changer functions on the IBM 7331, 7336, and 7337. The data path is supported by the AIX base device driver and utilities, such as *tctl* and *mt*.

Interactive Interface

The interactive interface of the tape utility program can be called from the AIX command-line using the *tapeutil* command. A list of general subcommands, medium changer subcommands, and tape subcommands is displayed. You must open a device before using these commands and operations (except for the tape drive service aids).

To open a device:

1. Select **Open a Device** from General Commands.
2. Enter the name of the device special file. Use any special file that exists for the device, for example, */dev/rmt0*, */dev/rmt0.1*, */dev/rmt1.smc*, or */dev/smc0*.
3. Enter the *Read/Write*, *Read Only*, *Write Only*, or *Append* mode to open a device. These modes apply only to the tape devices.

After you open a device, select a command using the appropriate number for the command from the interface. Some commands require additional information after they are selected. If an error occurs, then the error number, the error text, and the device sense data (if applicable) are displayed.

Command-Line Interface

The command-line interface of the *tapeutil* has a syntax similar to the AIX *tctl* and *mt* commands and provides the same basic tape commands. The program also supports tape device, device driver, SCSI, and medium changer subcommands that use the additional functions of the tape device and device driver.

Call the *tapeutil* command from the AIX command-line or from within a shell script. If you enter the *tapeutil* command without any arguments, the interactive interface is called.

The syntax for the command-line interface of the tape utility program is:

```
tapeutil -f Device Subcommand [Subcommand ...]
```

Notes:

1. *Device* is the name of the device special file (for example, */dev/rmt0*).
2. *Subcommand* is any valid command for the device.

Multiple subcommands can be combined in a single command to perform more than one operation. The subcommands are processed one at a time in the order specified on the command-line. For a list of subcommands and syntax, enter the tapeutil ? command on the AIX command line.

General Subcommands

The following general subcommands are available for the tape and medium changer devices.

checkpath: This subcommand issues a SCSI Inquiry command to the device on all configured paths to determine the status of each path and sets the path to enabled if successful or disabled if unsuccessful. The status of the primary and alternate path is displayed when the subcommand completes. The output is the same as the *qrypath* and *resetpath* subcommands. If more than two paths are configured for the device, run the *path*subcommand also.

Examples:

```
# Check the status of paths and display results
tapeutil -f/dev/rmt0 checkpath

# Check the status of more than 2 paths and display results
tapeutil -f/dev/rmt0 checkpath path
```

devinfo: This subcommand displays the device information returned from the IOCINFO *ioctl* command.

disable path Primary|Alternate: This subcommand is used to disable the device driver from using either the primary SCSI path or the alternate SCSI path to a device. This subcommand is valid only if the device has both a primary and alternate path enabled.

fuser: This subcommand is similar to the AIX *fuser*command. If the device special file is currently open, the process ID is displayed. Otherwise, it indicates the device special file is not currently open.

inquiry [Page]: This subcommand issues the SCSI Inquiry command to the device for either standard inquiry data, with no *page* parameter, or for specified page data, and displays the inquiry data. The *page* parameter must be specified as a hex value.

Examples:

```
# Get standard inquiry data
tapeutil -f/dev/rmt0 inquiry

# get inquiry page x'83'
tapeutil -f/dev/rmt0 inquiry 83
```

kill: This subcommand kills a currently active process on a device. This subcommand should be used when the standard Ctrl-C or AIX *kill* command methods cannot terminate the process. For example:

```
# Kill any active process on rmt0
tapeutil -f/dev/rmt0 kill
```

logpage Page: This subcommand issues the SCSI Log Sense command to the device for the specified *page* and displays the *log sense* data. The *page* parameter must be specified as a hex value. For example:

```
# Get log page x'31'
tapeutil -f/dev/rmt0 logpage 31
```

loop [Count]: This subcommand loops all subsequent subcommands continuously or a number of times if the *Count* parameter is specified. Also refer to the *sleep* subcommand.

Examples:

```
# Continuously loop inquiry commands with a 2-second delay
tapeutil -f/dev/rmt0 loop inquiry sleep 2

# Issue 3 Test Unit Ready commands
tapeutil -f/dev/rmt0 loop 3 tur
```

modepage Page: This subcommand issues the SCSI Mode Sense command to the device for the specified *page* and displays the *mode sense* data. The *page* parameter must be specified as a hex value. For example:

```
# Get mode page x'1D'
tapeutil -f/dev/rmt0 modepage 1d
```

Note: Issuing this subcommand for *page x′3F′ all pages* is not supported.

passthru: This subcommand opens the device special file using the SC_PASSTHRU mode. This mode bypasses normal open/close processing and no SCSI commands are issued to the device during open or close.

path: This subcommand displays information about the device and SCSI paths, such as logical parent, SCSI IDs, and the status of the SCSI paths for the primary path and all alternate paths that are configured. The output from this subcommand is similar to the *qrypath* subcommand, except more than one alternate path is supported by this subcommand. For example:

```
# Display path information
tapeutil -f/dev/rmt0 path
```

print ″Text″: This subcommand prints the associated text to standard output. It can be used at any time to display the progress of the subcommands. For example:

```
# Set volume id, erase current tape and backup myfile.tar
tapeutil -f/dev/rmt0 volid "My Volume"           \
                     rewind                      \
                     erase                       \
                     print "Writing myfile.tar"  \
                     write -s myfile.tar
```

qrypath: This subcommand displays information about the device and SCSI paths, such as logical parent, SCSI IDs, and status of the SCSI paths.

release: This subcommand explicitly releases a device and makes it available for other hosts. Refer to "Reserve and Release Commands" on page 51 for more information.

reqsense: This subcommand issues the SCSI Request Sense command to the device and displays the sense data.

reserve: This subcommand explicitly reserves a device. Refer to "Reserve and Release Commands" on page 51 for more information.

reset: This subcommand opens the device special file using SC_FORCED_OPEN mode and causes a bus device reset to be sent to the device.

resetpath: This subcommand is used to enable the device driver to use both the primary SCSI path or the alternate SCSI path after one path has been disabled with the *disablepath* subcommand.

sleep Seconds: This subcommand causes the *tapeutil* program to sleep the specified seconds before executing the next subcommand. Also refer to the *loop* subcommand.

Examples:

```
# Issue 2 inquiry commands with a 1-second delay between commands
tapeutil -f/dev/rmt0 inquiry sleep 1 inquiry

# Loop inquiry commands with a 2-second delay between commands
tapeutil -f/dev/rmt0 loop inquiry sleep 2
```

tur: This subcommand issues the SCSI Test Unit Ready command to the device.

vpd: This subcommand obtains the vital product data (VPD) from a SCSI tape device. It opens the device special file in SC_DIAGNOSTIC mode and uses the SCSI pass-through *ioctl* command to obtain the *inquiry* data from the device.

Medium Changer Subcommands

The following medium changer subcommands are available for the integrated and independent medium changer devices.

allow: This subcommand allows medium removal by an operator. It is normally used after the *prevent* subcommand to restore the device to the default state.

audit [Address[Count]]: This subcommand with no parameters issues the SCSI Initialize Element Status command to the device.

Using the optional parameters *Address* and *Count* issues the SCSI Initialize Element Status With Range command to the device. The *Address* parameter specifies the starting element address and the *Count* parameter, if used, specifies the number of elements to initialize. If *Count* is omitted, it defaults to 1. For example:

```
# Initialize all elements
tapeutil -f/dev/smc0 audit

# Initialize element 32
```

```
tapeutil -f/dev/smc0 audit 32

# Initialize elements 36 to 40
tapeutil -f/dev/smc0 audit 36 5
```

Note: The SCSI Initialize Element Status With Range command is supported on the IBM 7337 and the IBM 3575.

devids: This subcommand issues the SCSI Read Element Status command to the device with the *read device id* option for all drive elements and displays the element status information, which includes the *device id* field.

elementinfo: This subcommand displays the information returned from the SMCIOC_ELEMENT_INFO *ioctl* command that contains the number and address of each element type.

exchange "Source" "Dest1" "Dest2": This subcommand issues the SCSI Exchange Medium command to the device using the *Source*, *Dest1*, and *Dest2* addresses specified. This command performs the equivalent function of two *Move Medium* commands. The first moves the cartridge from the element address specified by the *Dest1* parameter to the element address specified by the *Dest2* parameter. The second moves the cartridge from the element address specified by the *source* parameter to the element address specified by the *Dest1* parameter. For example:

```
# Exchange cartridge in slot 34 with cartridge in drive address 16
# and return that cartridge to slot 40 (drive must be unloaded first)
tapeutil -f/dev/smc0 exchange 34 16 40

# Use move medium commands to perform exchange
tapeutil -f/dev/smc0 move 16 40 move 34 16
```

inventory [-i | -v Volid]: This subcommand with no parameters issues the SCSI Read Element Status command for each element type and displays the element status information. If the optional -i parameter is used, then only the import/export element status information will be returned. If the optional -v parameter is used, then only the element status information for the specified Volid if found will be returned.

mount [Slot]: This subcommand mounts a tape from the specified *slot* into the drive or from the first full *slot* into the drive if the slot is omitted. For example:

```
# Mount cartridge from slot 3
tapeutil -f/dev/smc0 mount 3

# Mount cartridge from first full slot
tapeutil -f/dev/smc0 mount
```

move "Source" "Destination": This subcommand issues the SCSI Move Medium command using the *source* and *destination* addresses specified. The element addresses can be obtained using the *elementinfo* subcommand. For example:

```
# Get slot and drive addresses
tapeutil -f/dev/smc0 elementinfo

# Move cartridge in slot 20 to drive at address 16
tapeutil -f/dev/smc0 move 20 16
```

position "Destination": This subcommand issues the SCSI Position to Element command using the destination specified. For example:

```
# Position to slot at address 20
tapeutil -f/dev/smc0 position 20
```

prevent: This subcommand prevents medium removal by an operator until the *allow* subcommand is issued or the device is reset.

unmount [Slot]: This subcommand moves a tape from the drive to the specified *slot* or the first empty one if the slot is omitted. The tape is rewound and unloaded automatically from the drive first when this command is issued to the tape device special file. For example:

```
# Move tape from drive to slot 4 (tape is already unloaded)
tapeutil -f/dev/rmt0.smc unmount 4

# Unload tape and move to the first empty slot
tapeutil -f/dev/rmt0 unmount
```

Tape Subcommands

The following tape subcommands are available for the tape devices.

allow: This subcommand issues the SCSI Prevent Allow Medium Removal command to the device to allow medium removal by an operator. It is normally used after the *prevent* subcommand to restore the device to the default state.

append: This subcommand opens the device in *append* mode and allows the appending of data to the end of the current tape. The subcommand can be used with a No Rewind on Close special file to set the tape position after the last file that was written.

```
# Append myfile.tar to the end of tape using dd subcommand
tapeutil -f/dev/rmt0.1 append
dd if=myfile.tar of=dev/rmt0
```

autoload: This subcommand turns on the autoload feature only for subsequent subcommands. For example:

```
# Backup large.tar (requires multiple tapes) using autoload feature
tapeutil -f/dev/rmt0  autoload  write -s large.tar
```

bsf [Count]: This subcommand backward spaces the filemarks. An optional *count* can be specified. The default is 1.

bsr [Count]: This subcommand backward spaces the records. An optional *count* can be specified. The default is 1.

chgpart Number [Blockid]: This subcommand changes the current active tape partition to a new partition specified by *Number*. Optionally, a *Blockid* can also be specified. If *Blockid* is omitted, the tape is positioned at the start of the new partition, otherwise, the tape is positioned at the *Blockid* specified. For example:

```
# Change to beginning of partition 16
tapeutil -f/dev/rmt0 chgpart 16

# Change to partition 0, block number 25
tapeutil -f/dev/rmt0 chgpart 0 25
```

compress and nocompress: These subcommands turn On and Off compression only for subsequent subcommands.

density: This subcommand issues the SCSI Report Density command for all supported media and for the current media loaded in the drive, and displays the results. If the drive is not loaded, the *current media density* is not reported and a Drive Not Ready error is returned.

display ″Message″: This subcommand displays a message on the display panel of the tape device. Up to 16 characters can be used for the message. If the message is longer than eight characters, the display alternates between the first eight characters and the remainder of the message. For example:

```
# Display "Test 1" and "Running" messages
tapeutil -f/dev/rmt0  display "Test 1  Running"
```

eof [Count] and weof [Count]: These subcommands write the filemarks. An optional *count* can be specified. The default is 1.

erase: This subcommand erases the tape.

erg: This subcommand performs an erase record gap operation.

fsf [Count]: This subcommand forward spaces the filemarks. An optional *count* can be specified. The default is 1.

fsr [Count]: This subcommand forward spaces the records. An optional *count* can be specified. The default is 1.

idp: This subcommand creates an IDP partition on tape. For example:

```
# Create IDP partition
tapeutil -f/dev/rmt0 idp
```

list: This subcommand displays the content of a tape. The output lists filemarks and the size of each record found on the tape until the end of data is reached. The output generated from this subcommand can be large depending on the amount of data on the tape and should usually be directed to a file. For example:

```
# List tape contents to file
tapeutil -f/dev/rmt0 list > tape.list
```

load: This subcommand issues the SCSI Load command to load the next tape.

logsense: This subcommand issues the STIOC_LOG_SENSE *ioctl* command and displays the data. If volume logging is active, then the *log sense* data is also saved in the log file.

mtdevice: This subcommand issues the MTDEVICE *ioctl* command and displays the library sequence number. Refer to the MTLIB command *-D* flag in Figure 41 on page 244.

noautoload: This subcommand turns off the autoload feature only for subsequent subcommands. For example:

```
# Make sure autoload feature is off before writing file to tape
tapeutil -f/dev/rmt0 noautoload write -s myfile.tar
```

nosili: This subcommand turns off the Suppress Incorrect Length Indication (SILI) bit in variable length SCSI Read commands for all subsequent subcommands, such as *rtest*, *rwtest*, and *read*.

offline, rewoffl, and unload: These subcommands rewind and unload the tape.

parms and status: These subcommands issue the STIOCQRYP *ioctl* command and display the current *tape drive*, *media*, and *device driver* parameters.

prevent: This subcommand issues the SCSI Prevent Allow Medium Removal command to the device to prevent medium removal by an operator until the *allow* subcommand is issued or the device is reset.

qrypart: This subcommand queries and displays tape partitioning information.

qrypos: This subcommand issues the STIOCQRYPOS *ioctl* command for the logical and physical tape positions and displays the data. In addition, the current tape position is saved and can be restored using a subsequent *setpos* subcommand. For example:

```
# Append myfile.tar to the end of tape and then read back
tapeutil -f/dev/rmt0.1 append                \
                       qrypos                \
                       write -s myfile.tar   \
                       setpos                \
                       read -d temp.tar

# Verify myfile.tar was written correctly
diff myfile.tar temp.tar
```

read -d Destination [-c Count]: This subcommand reads a file, or a specified number of records, from the tape to the destination file name specified with the *-d* flag. If the optional *count* parameter is used, only the number of records specified with the *-c* flag are read unless a filemark is encountered before the number of specified records. If the *count* parameter is not used, all records up to the next filemark on tape are read. For example:

```
# Restore myfile.tar from tape
tapeutil -f/dev/rmt0 read -d myfile.tar

# Read 3 records from the tape into myfile
tapeutil -f/dev/rmt0 read -d myfile -c3
```

rewind and retension: These subcommands rewind the tape.

rtest [-b Blocksize] [-c Count] [-r Repetition]: This subcommand performs a read test by reading a random data pattern from the tape and verifying that it matches the written data. The *rtest* subcommand can be used after the *wtest* subcommand to verify the data.

An optional *block size*, *count*, and *repetition* can be specified with the *-b*, *-c*, and *-r* flags, respectively. If the *block size* is fixed, then the *count* specifies the number of blocks to read on each *repetition*. If the *block size* is zero (variable), then the *count* specifies the number of bytes to read on each *repetition*. The default is a *block size* of 10240, a *count* of 20 blocks, and a *repetition* of 1. For example:

```
# R/W test using 256 KB blocks, 5 megabytes per write, 100 times
tapeutil -f/dev/rmt0 rewind                    \
                     wtest -b 262144 -c 20 -r 100 \
                     rewind                    \
                     rtest -b 262144 -c 20 -r 100
```

rwtest [-b Blocksize] [-c Count] [-r Repetition]: This subcommand performs a read and write test by writing a random data pattern on the tape, reading it, and verifying that it matches the written data.

An optional *block size*, *count*, and *repetition* can be specified with the *-b*, *-c*, and *-r* flags, respectively. If the block size is fixed, then the *count* specifies the number of blocks to write on each *repetition*. If the block size is zero (variable), then the *count* specifies the number of bytes to write on each *repetition*. The default is a *block size* of 10240, a *count* of 20 blocks, and a *repetition* of 1. For example:

```
# R/W test using 256 KB blocks, 5 megabytes per write, 10 times
tapeutil -f/dev/rmt0 rwtest -b 262144 -c 20 -r 10
```

sdp Number: This subcommand creates the specified *Number* of SDP partitions on tape. For example:

```
# Create 32 SDP partitions and then query partitioning
tapeutil -f/dev/rmt0 sdp 32 qrypart
```

seod: This subcommand spaces to the end of data on the tape.

setpos [Blockid]: This subcommand issues the SCSI *Locate* command to the device to set the tape position. If the optional *Blockid* parameter is specified, the tape position is set to the *Blockid*. Otherwise, if the *Blockid* parameter is omitted, the tape position is set to the last position saved using the *qrypos* subcommand. The *Blockid* can be specified in decimal or in hex, with a leading x. For example:

```
# Append myfile.tar to the end of tape and then read back
tapeutil -f/dev/rmt0.1 append                \
                       qrypos                \
                       write -s myfile.tar   \
                       setpos                \
                       read -d temp.tar

# Verify myfile.tar was written correctly
diff myfile.tar temp.tar

# Set tape position to block 32 and leave positioned on close
tapeutil -f/dev/rmt0.1 setpos 32

# Set tape position to block 32 and leave positioned on close
tapeutil -f/dev/rmt0.1 setpos x20
```

sili: This subcommand turns on the SILI bit in variable length SCSI Read commands for all subsequent subcommands, such as *rtest*, *rwtest*, and *read*.

sync: This subcommand synchronizes or flushes the tape buffers to tape.

volid "Name": This subcommand sets the volume ID for logging. Refer to "Volume ID for Logging" on page 31 and "Volume ID for Logging" on page 51.

write -s Source: This subcommand writes the source file specified with the *-s* flag on the tape. For example:

```
# Backup myfile.tar to tape
tapeutil -f/dev/rmt0 write -s myfile.tar
```

wtest [-b Blocksize] [-c Count] [-r Repetition]: This subcommand performs a write test by writing a random data pattern on the tape. The *rtest* subcommand can be used after the *wtest* subcommand to verify the data that was written.

An optional *block size*, *count*, and *repetition* can be specified with the *-b*, *-c*, and *-r* flags, respectively. If the block size is fixed, then the *count* specifies the number of blocks to write on each *repetition*. If the block size is zero (variable), then the *count* specifies the number of bytes to write on each *repetition*. The default is a *block size* of 10240, a *count* of 20 blocks, and a *repetition* of 1. For example:

```
# R/W test using 256 KB blocks, 5 megabytes per write, 100 times
tapeutil -f/dev/rmt0 rewind                      \
                     wtest -b 262144 -c 20 -r 100 \
                     rewind                      \
                     rtest -b 262144 -c 20 -r 100
```

Service Aid Subcommands

The following service aid subcommands are available for the tape devices.

dump [Filename]: This subcommand forces a dump on the tape device and stores the dump in the specified *Filename* or, if *Filename* is omitted, in the system */var/adm/ras* directory. The device driver stores up to three dump files in this directory. The first dump file is named *Atape.rmtx.dump1*, where *x* is the device number (for example, *rmt0*). The second and third dump files are named *dump2* and *dump3*, respectively. After a third dump file is created, the next dump starts at *dump1* again and overlays the previous dump file.

fmrtape: This subcommand issues a Send Diagnostic SCSI command to the device to create a field microcode replacement (FMR) cartridge tape. The tape is created using the loaded functional microcode in the device when the command is issued.

forcedump: This subcommand issues a Send Diagnostic SCSI command to force a dump on the tape drive. The dump can be read using the *readdump* subcommand.

readdump [*Filename***]:** This subcommand reads and stores a tape drive dump in the specified filename or, if filename is omitted, in the system */var/adm/ras directory*. It performs the same function as the *dump* subcommand above without forcing a dump first.

resetdrive: This subcommand issues a Send Diagnostic SCSI command to reset the device.

ucode ″**Name**″**:** This subcommand downloads microcode to the device. The *Name* parameter can specify a diskette drive, such as */dev/rfd0*, or a microcode file on the RISC machine. For example:

```
# download microcode from diskette
tapeutil -f/dev/rmt0 ucode /dev/rfd0

# download microcode from RISC file
tapeutil -f/dev/rmt0 ucode /etc/microcode/device.ucode
```

Automatic Cartridge Facility Mode

If the device is the IBM TotalStorage Enterprise Tape System 3590 with an Automatic Cartridge Facility (ACF) or the IBM Magstar MP 3570 library, the ACF mode can be queried using either the interactive interface of the tape utility program and selecting **Query/Set Parameters** under Tape Commands or using the command-line interface and issuing the *parms* or *status* subcommand.

Block Size and SCSI Transfer Size

The minimum and maximum block sizes for the tape device and the maximum SCSI transfer size can be queried using either the interactive interface of the tape utility program and selecting **Query/Set Parameters** under Tape Commands or using the command-line interface and issuing the *parms* or *status* subcommand.

Configuration Parameters

The configuration parameters can be queried using either the interactive interface of the tape utility program and selecting **Query/Set Parameters** under Tape Commands or using the command-line interface and issuing the *parms* or *status* subcommand.

The configuration parameters can be changed temporarily using the interactive interface of the tape utility program and selecting **Query/Set Parameters** under Tape Commands.

Note: The changes are effective only while the current device is open. All configuration parameters are reset to their current default values when the device is closed.

Capacity Scaling

The capacity scaling of the current tape can be queried or set using the interactive interface of the tape utility program and selecting **Query/Set Parameters** under Tape Commands.

Logical Write Protect

The logical write protect of the current tape can be queried or set using the interactive interface of the tape utility program and selecting **Query/Set Parameters** under Tape Commands.

Reserve and Release Commands

The device driver reserves the device automatically on the *open* call and releases the device on the *close* call. This prevents other applications and hosts from accessing the device. However, there may be situations when the reserve should be maintained after the *close* call. For example, some backup programs such as *tar* can open and close the device multiple times. In other situations, a device is shared by multiple initiators or hosts. In these cases, the reservation must be retained between the *close* call and the next *open* call. This is done by explicitly reserving the device. After all work is completed on the device, the user will explicitly release the device.

A device can be reserved and released explicitly using either the interactive interface of the tape utility program and selecting **Reserve** or **Release** under General Commands or using the command-line interface and issuing the reserve and release subcommands. For example:

```
# Reserve device, run tar, and then release device
tapeutil -f/dev/rmt0 reserve
tar    ... /dev/rmt0 ...
tapeutil -f/dev/rmt0 release
```

After the reserve command is used, the device driver retains the reservation until a release command is issued, even if the device is deconfigured and reconfigured.

Volume ID for Logging

The volume ID used in the log entry when volume logging is active can be set using either the interactive interface of the tape utility program and selecting **Query/Set Parameters** under Tape Commands or using the command-line interface and issuing the *volid* subcommand. For example:

```
# Unload tape, load next tape, and set volume id
tapeutil -f /dev/rmt0  unload  load  volid "VOLSER001"
```

Problem Determination

A set of tools is provided with the device driver to determine if the device driver and the tape device are functioning correctly. The standard AIX interface is provided for problem determination.

Using the Dump Support

Dump support is provided through the dump entry point in the driver. Refer to appropriate AIX manuals for a description of how to use the dump devices and how to read the dump data.

Dump Device Commands

To list the current dump devices, enter the following command:

```
sysdumpdev -l
```

To establish the *rmt1* tape device as a secondary dump device, enter the following command:

```
sysdumpdev -s /dev/rmt1
```

To perform a dump operation, use the *sysdumpstart* command. To send the dump data to the secondary dump device, enter the following command:

```
sysdumpstart -s
```

Note: This command stops the system. Use the *sync* command to ensure that the cache is flushed before issuing the *sysdumpstart -s* command.

To list the last dump data, enter the following command:

```
sysdumpdev -z
```

After the dump data is placed on the tape, copy it to a file on the disk before using the *crash* command to process it. For example:

```
dd if=/dev/rmt1 of=tapedump1 ibs=4096 obs=512
crash tapedump1
```

Note: The *ibs* value is the input block size.

If the block size of the tape device is larger than the block size sent during the dump process, the dump operation fails. Set the block size to zero on the tape device and experiment with the *ibs* value for the *dd* command.

Device and Volume Information Logging

An optional logging utility is provided to log the information about the device and the media. The information is extensive for some devices and limited for other devices. If set to On, the logging facility gathers all available information through the SCSI Log Sense command.

This process is a separate facility from error logging. Error logging is routed to the system error log. Device information logging is sent to a separate file.

The following parameters control this utility:

- Logging
- Maximum size of the log file
- Volume ID for logging

Refer to “Tape Drive, Media, and Device Driver Parameters” on page 26 for a description of these parameters.

Each time the rewind and unload sequence occurs or the STIOC_LOG_SENSE *ioctl* command is issued, an entry is added to the log. Each time a new cartridge is loaded, the values in the device log buffers are reset with the Log Sense command. The log data is gathered on a per-volume basis.

Log File

The data is logged in the */usr/adm/ras* directory. The file name is dependent on each device; therefore, each device has a separate log. An example of the *rmt1* device file is:

```
/usr/adm/ras/Atape.rmt1.log
```

The files are in binary format. Each entry has a header followed by the raw Log Sense pages as defined for a particular device.

The first log page is always page 0x00. This page, as defined in the SCSI-2 ANSI specification, contains all pages the device supports. Page 0x00 is followed by all pages specified in page 0x00. The format of each following page is defined in the SCSI specification and the device manual.

Tape Log Utility

A tape log utility is installed with the *tapelog* device driver that displays the contents of the log file in ASCII text. The log pages are displayed as hexadecimal values in dump format.

The C source code (*tapelog.c*) for the program is in the */usr/lpp/Atape/samples* directory. The program contains a sample of the interface to the log file and the structure used to read the file.

The syntax for the tape log utility is:

```
tapelog -l Name [-d] or tapelog -f File [-d]
```

Notes:

1. *Name* is the logical name of the device (such as *rmt0*).
2. *File* is the name of a log file (such as *Atape.rmt0.log*).
3. The *-d* parameter, if used, deletes the log file for the specified device.

The contents of the log file are displayed as standard output. To save the log in a file, use the AIX redirection function. For example:

```
tapelog -l rmt0 > rmt0.log
```

Error Logging

The device driver provides logging to the AIX system error log for various errors. The error log can be viewed for specific devices using the Error Log Analysis utility provided with the tape drive service aids. Refer to “Error Log Analysis” on page 59. The error log can also be viewed using the *smit* or the *errpt* command.

Error Log Templates

The error log templates the device driver uses follow the same format as the default AIX tape error log entries. Each error log entry is identified by an error label and contains detail data associated with the type of error. The following items describe the error labels and detail data for the templates used for logging tape device, media, and SCSI adapter related errors in the AIX system error log.

Error Labels

Errors are logged with an associated error label and error ID. The error label indicates the basic type of error:

- TAPE_ERR1

  Tape media error
- TAPE_ERR2

  Tape hardware error
- TAPE_ERR4

  SCSI Adapter detected error
- TAPE_ERR5

  Unknown error
- RECOVERED_ERROR

  Temporary tape hardware or media error
- SIM_MIM_RECORD_3590

  3590 Service/Media Information Message (Log Page X'31')
- TAPE_SIM_MIM_RECORD

  Tape drive Service/Media Information Message (Log Page X'31')
- DEV_DUMP RETRIEVED

  Device dump retrieved
- TAPE_DRIVE_CLEANING

  Tape drive needs cleaning

Detail Data

Detail data is logged with the associated error that identifies the cause of the error. Detail data for the SIM_MIM_RECORD_3590 or TAPE_SIM_MIM_RECORD entries contain the raw data from Log Sense Page X'31'. Refer to the hardware reference manual for the format of this entry. All other error log entries use the following format for detail data:

```
Detail Data
SENSE DATA
aabb xxxx ccdd eeee eeee eeee eeee eeee ffgg hhxx ssss ssss ssss ssss ssss
ssss ssss ssss ssss ssss ....
```

where:

aa Length of the command descriptor block (CDB)

bb SCSI target address

xx Unused or reserved

cc Start of CDB, cc is the operation code (byte 0)

dd Logical unit (byte 1) in the CDB

ee Bytes 2 through 12 in the CDB

ff Status validity field. If this field is 01, then a SCSI error was reported, and byte *gg* indicates the type of error. If this field is 02, then an adapter error was reported, and byte *hh* indicates the type of error.

gg This byte indicates the type of SCSI error that occurred:

- 02 CHECK CONDITION - Device reported a check condition.
- 08 BUSY STATUS - Target is busy.
- 18 RESERVATION CONFLICT - Target is reserved to another initiator.

- 22 COMMAND TERMINATED - Device terminated the command.
- 28 QUEUE FULL - Device command queue is full.

hh This byte indicates the type of adapter error that occurred. For parallel SCSI adapters, this is the general_card status code as defined in */usr/include/sys/scsi.h*:

- 01 HOST IO BUS ERROR - Host I/O bus error during data transfer.
- 02 SCSI BUS FAULT - SCSI bus protocol or hardware error.
- 04 COMMAND TIMEOUT - Command timed out before completion.
- 08 NO DEVICE RESPONSE - Target did not respond to selection phase.
- 10 ADAPTER HARDWARE FAILURE - Adapter indicated a hardware failure.
- 20 ADAPTER SOFTWARE FAILURE - Adapter indicated a microcode failure.
- 40 FUSE OR TERMINAL PWR - Blown terminator fuse or bad termination.
- 80 SCSI BUS RESET - Adapter indicated SCSI bus has been reset.

For FCP or SAS adapters, this is the adapter_status code as defined in */usr/include/sys/scsi_buf.h*:

01 HOST IO BUS ERROR - Host I/O bus error during data transfer.

02 TRANSPORT FAULT - Failure in the transport layer.

03 COMMAND TIMEOUT - Command timed out before completion.

04 NO DEVICE RESPONSE - Target did not respond to attempts to select it.

05 ADAPTER HARDWARE FAILURE - Adapter indicated a hardware failure.

06 ADAPTER SOFTWARE FAILURE - Adapter indicated a microcode failure.

07 WW NAME CHANGE - Adapter detected a new worldwide name for the device.

08 FUSE OR TERMINAL PWR - Blown terminator fuse or bad termination.

09 TRANSPORT RESET - Adapter detected an external SCSI bus reset.

0A TRANSPORT BUSY - The transport layer is busy.

0B TRANSPORT DEAD - The transport layer is currently inoperative.

ss If byte *gg* indicates a check condition, the *ss* byte is the sense data from the device. Refer to the appropriate device reference manual for the specific format and content of these bytes.

Automatic Dump Facility

The device driver provides an automatic dump facility for devices that support reading a dump and indicating when a dump is available in device sense data. Whenever a check condition occurs and the sense data indicates that a dump is available, the device driver reads the dump from the device and stores it in the */var/adm/ras* directory. A maximum of three dumps for each device are stored in this directory as:

```
Atape.rmtx.dump1
Atape.rmtx.dump2
Atape.rmtx.dump3
```

Note: *X* is the device number, for example, *rmt0*.

When the device is first configured, the dump name is set to *dump1*. If more than three dumps occur, the driver starts over at *dump1*; therefore the last three dumps are always kept.

Trace Facility

The AIX trace facility is supported for the device driver. The trace event is identified with a *hookword*. The hookword used by the device driver is 326. The trace can be initiated at any time before an operation on a tape device.

Enter the following AIX command to start the trace:

```
trace -a -j 326
```

This command starts the trace in the background and collects only the trace events with the 326 hookword (*Atape* device driver).

Enter the following AIX command to stop the trace:

```
trcstop
```

This command stops the trace after the tape operations are performed.

Enter the following AIX command to view the trace:

```
trcrpt > trace.out
```

This command formats the trace output into a readable form and places it into a file for viewing.

Atape System Trace (ATRC) Utility

The *atrc* trace utility is also installed with the device driver to start, stop, and format a device driver trace. To start the trace, enter the *atrc* command. To stop and format the trace, enter the *atrc* command again. The trace is formatted to an *atrc.out* AIX file in the current directory.

Component Tracing

Later releases of AIX 5.3 and above support component tracing. Unlike system tracing that needs to be started and stopped, component tracing by default is on all the time and runs continually.

To determine if component tracing is available run the command **ctctrl -q** to display a list of supported components with their default settings. You must have root authority to run this command. Refer to the AIX ctctrl man page for a complete description of the **ctctrl** command and parameters.

To dump and format the current component trace to a file (for example, *actrc.out*) into the current directory, run the following commands:

```
ctctrl -D -c Atape
trcrpt -l Atape -o actrc.out
```

The Atape component trace can also be retrieved from a system dump. This eliminates the need to start the Atape system trace prior to a system dump or to recreate an AIX system dump when a system trace is not running. The AIX system dump is normally stored in the */var/adm/ras* directory as a *vmcore.x.BZ* file, where *x* is a dump number 1, 2, and so on.

To retrieve and format the Atape component trace from a dump file (for example, *vmcore.1.BZ*) to a file (for example, *actrc.dump*) into the current directory, run the following commands:

```
dmpuncompress /var/adm/ras/vmcore.1.BZ
trcdead -c /var/adm/ras/vmcore.1
trcrpt -l Atape -o actrc.dump
```

Atape Component Trace (ACTRC) Utility

The actrc component trace utility is also installed with the device driver to dump and format the current Atape component trace. To dump and format the component trace, run the command *actrc*. The trace is formatted to an *actrc.out* file in the current directory.

Tape Drive Service Aids

The service aids provided with the device driver can be called using the interactive interface of the tape utility program and selecting **Tape Drive Service Aids** under General Commands or using the Service Aid Subcommands in the command-line interface of the tape utility program. Refer to "Service Aid Subcommands" on page 50.

Note: The AIX diagnostic subsystem must be installed in order to use **Tape Drive Service Aids** from the *tapeutil* window.

The service aids described here are accessible through the AIX diagnostic subsystem using the AIX *diag* command, or the interactive and command-line interfaces of the *tapeutil* program installed with the device driver. Refer to "Service Aid Subcommands" on page 50 and "Details of Tape Service Aids."

To access the service aids using the *diag* command:

1. Enter the *diag* command.
2. Select **Service Aids** from the Diagnostic Function Selection window.
3. Select **IBM Tape Drive Service Aids** from the Service Aid Selection window.

Note: You must have *root* authority to use the *diag* command.

To access the service aids using *tapeutil*:

1. Enter the *tapeutil* command.
2. Select **Tape Drive Service Aids** under General Commands from the window.

Note: The AIX diagnostic subsystem must be installed in order to use **Tape Drive Service Aids** from *tapeutil*.

Details of Tape Service Aids

The following service aid utilities are installed with the device driver:

- Force Microcode Dump
- Read Dump
- Microcode Load
- Error Log Analysis
- Reset Drive
- Create an FMR Tape

Force Microcode Dump

This utility forces a dump operation on the tape drive. After the dump operation is performed, the dump data can be transferred from the tape drive using the Read Dump utility.

To access this utility:

1. Open the Service Aids window.
2. Select **Force Microcode Dump** from the IBM Tape Drive Service Aids window, then press Enter.
3. Select the device from the IBM Tape Drive Selection window, then press Enter.

   The Force Microcode Dump operation starts, and a window opens when the operation is completed.

Read Dump

This utility transfers the dump data from the device to a file, a diskette, or a tape cartridge.

To access this utility:

1. Open the Service Aids window.
2. Select **Read Dump** from the IBM Tape Drive Service Aids window, then press Enter.
3. Select the device from the IBM Tape Drive Selection window, then press Enter.
4. Enter the destination file name or device in the Prompting for Destination window. The default destination is the */dev/rfd0* diskette drive. To transfer the dump data to a tape cartridge, enter the device name of the tape drive (for example, */dev/rmt0*). To transfer the dump data to a file, enter the file name. Press F7 to commit.

   Note: On certain terminal types, it may be necessary to press the Esc key and the number 7 key instead of F7.

   The Read Dump operation starts, and a window opens when the operation is completed.

Microcode Load

This utility downloads microcode to the device from a file or a diskette (AIX format only).

Note: To download the microcode from a DOS diskette, you must first use the AIX *dosread* command to transfer the file from the DOS diskette to the AIX file. Then you can use the Microcode Load utility to download the AIX file to the tape drive.

To access this utility:

1. Open the Service Aids window.
2. Select **Microcode Load** from the IBM Tape Drive Service Aids window, then press Enter.
3. Select the device from the IBM IBMTape Drive Selection window, then press Enter.
4. Enter the source file name or device on the Prompting for Source File window. The default source is the */dev/rfd0* diskette drive. To load from a file, enter the file name. Press F7 to commit.

Note: On certain terminal types, it may be necessary to press the Esc key and the number 7 key instead of F7.

5. If the current microcode on a tape drive is Federal Information Processing Standard (FIPS) code, then a window will display the following:

   ```
   Warning: The drive is currently using FIPS code. Press Enter to
   continue with downloading new drive code.
   ```

   If you do not want to download the new code, press either F3 to cancel or F10 to exit without downloading new code. Otherwise, press Enter to continue with the download code procedure.

The Microcode Load operation starts, and a window opens when the operation is completed.

Error Log Analysis

This utility displays and analyzes the system error log entries for a specific tape drive and can be used for problem determination. The type of error, the SCSI command, and the sense data (if applicable) are displayed for each entry in the error log (one screen at a time).

To access this utility:

1. Open the Service Aids window.
2. Select **Error Log Analysis** from the IBM Tape Drive Service Aids window, then press Enter.
3. Select the device from the IBM Tape Drive Selection window, then press Enter.
4. If entries are listed in the error log for the selected device, then the first entry is displayed. Press Enter to display the next entry.
5. After all entries are displayed, a window opens, and the operation is completed.

Reset Drive

This utility resets the tape drive.

To access this utility:

1. Open the Service Aids window.
2. Select **Reset Drive** from the IBM Tape Drive Service Aids window, then press Enter.
3. Select the device from the IBM IBMTape Drive Selection window, then press Enter.

   The Reset Drive operation starts, and a window opens when the operation is completed.

Create an FMR Tape

This utility creates a field microcode replacement (FMR) cartridge tape using the loaded functional microcode in the tape drive.

To access this utility:

1. Open the Service Aids window.
2. Select **Create an FMR Tape** from the IBM Tape Drive Service Aids window, then press Enter.
3. Select the device from the IBMTape Drive Selection window, then press Enter.

The Create an FMR Tape operation starts, and a window opens when the operation is completed.

Performance Considerations

This chapter describes the parameters and issues that may affect the perceived performance of the tape drive. In general, AIX applications that operate at a file level to move data between disk storage devices and tape do not exploit the full capabilities of a high end tape device. The goal of this discussion is to give an overview of the data path components involved in moving data between disk storage devices and tape. The following chapter describes basic techniques and common utilities in a specific environment that can be used to understand how a device is performing. Performance issues encountered by advanced application developers are beyond the scope of this document.

- Refer to the hardware reference for the specific device for performance specifications.
- Refer to the application documentation for information on device-specific application configuration.
- Refer to the operating system documentation for information on disk storage device striping and other techniques for improving file system performance.

Data Path

The simplified model in Figure 5 shows the components involved in the data path for moving data at a file level between disk storage devices and tape.

Performance analysis must be approached by determining which component of the data path impacts performance. Typically, a performance problem can be isolated by looking at one leg of the data path at a time. The goal of this analysis is to confirm that the tape data path is not impacting the performance adversely.

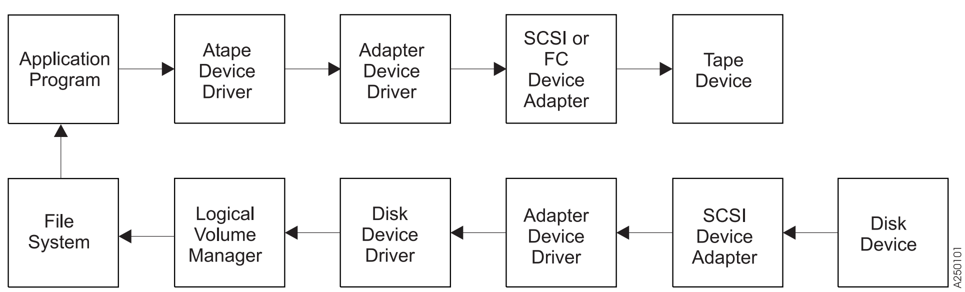


Figure 5. Data Path for AIX Device Driver (Atape)

Common AIX Utilities

The most commonly reported cause for poor tape performance is the use of small block sizes or the modification of the installation defaults for the tape device.

Note: The device parameters should not be changed from the defaults for most applications.

The following guidelines typically result in good tape path performance for use with AIX utilities:

1. Hardware compression should be enabled for maximum performance if the data sent to the device is uncompressed.

2. The *block_size* parameter should be set to variable (block_size=0) and command or application parameters specified to a block size appropriate for the device.
3. Block sizes of 128 KB or greater should be used to improve performance.

AIX iostat Utility for Tape Performance

In releases of AIX 5.3 and above, the AIX **iostat** utility supports tape performance statistics in addition to other supported devices (such as disk). To determine if the **iostat** utility supports the configured tape drives, run the command **iostat -p**. If the configured tape drives are supported, a list of configured tape drives will be displayed with the statistics listed for each drive. Refer to the AIX iostat man page for a complete description of the **iostat** command and parameters. When using the Data Path Failover feature, only the primary path for the tape drive is listed. The statistics apply to both the primary and alternate paths that are used.

Before Calling Support

System performance tuning is not a support responsibility. If tests indicate raw tape performance is below specifications, record the exact failing command and collect the output from the commands in Table 8 before contacting support.

Table 8. Error Description

Information	Command
Configuration	lscfg -v
Device parameters	lsattr -E -l rmtN
Error log. Call hardware support if errors are found for TAPE_ERR* or SCSI* error labels.	errpt -a
Driver version	lslpp -l Atape.driver
Device microcode level (may not apply to all devices)	tapeutil -f /dev/rmtN reqsense
Trace of failing command	Refer to “Trace Facility” on page 56

Chapter 4. HP-UX Tape and Medium Changer Device Driver

This chapter provides an overview of the IBM Tape and Medium Changer Device Driver for HP-UX that provides support for IBM tape drives and medium changer devices.

Purpose

This device driver product provides SCSI-3 attachment for the IBM tape devices to selected Hewlett-Packard Precision Bus and PCI-based platforms running HP-UX
| 10.20, 11.0, 11i v1, 11i v2, and 11i v3.

The HP-UX installed name of this device driver is Advanced Tape Device Driver (ATDD). ATDD is used throughout this section to identify this driver.

ATDD is designed specifically to take advantage of the features provided by the IBM tape devices (refer to "Supported Hardware" on page 9. The goal is to give applications access to the functions required for basic operations (such as backup and restore) and the advanced functions needed by full tape management systems. Whenever possible, the device driver is designed to take advantage of the IBM tape system features transparent to the application.

Data Flow

Both data and commands flow between the application program and the tape subsystem by way of the IBM Tape and Medium Changer Device Driver for HP-UX (ATDD). Figure 6 on page 64 shows the relationships between the IBM Tape and Medium Changer Device Driver for HP-UX, the application program, the

SCSI or Fibre Channel adapter device driver, and the IBM tape subsystem.

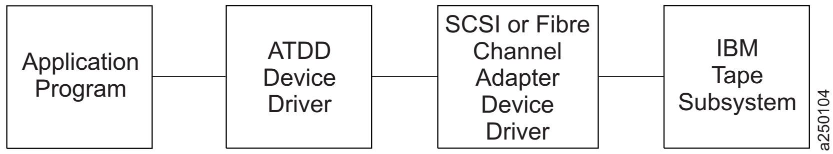


Figure 6. Data Flow for HP-UX Device Driver (ATDD)

Product Requirements

The following hardware and software components are required and supported by the IBM Tape and Medium Changer Device Driver for HP-UX (ATDD).

ATDD Implementation

ATDD is actually a set of device driver implementations for operation in the following HP-UX platform environments:

- HP Precision Bus - HP-UX 10.20 (32 bit) - versions of this driver are identified by levels ATDD 2.9.x.x.
- HP Precision Bus - HP-UX 11.00 (32 bit) - versions of this driver are identified by levels ATDD 4.9.x.x.
- HP PCI Bus - HP-UX 11.00 (64 bit) - versions of this driver are identified by levels ATDD 1.x.x.x.
- HP PCI Bus - HP-UX 11i v1 (64 bit) - versions of this driver are identified by levels ATDD 3.x.x.x.
- HP PCI Bus - HP-UX 11i v2 (64-bit for Itanium® 2) - versions of this driver are identified by levels ATDD 5.5.x.x.
- HP PCI Bus - HP-UX 11i v2 (64-bit for PA-RISC) - versions of this driver are identified by levels ATDD 5.6.x.x.
- HP PCI Bus - HP-UX 11i v3 (64-bit for Itanium 2) - versions of this driver are identified by levels ATDD IA.6.x.x.x.
- HP PCI Bus - HP-UX 11i v3 (64-bit for PA-RISC) - versions of this driver are identified by levels ATDD PA.6.x.x.x.

Hardware Requirements

ATDD supports the following HP SCSI host bus adapters for Ultrium tape drives:

- HP A4800A PCI Ultra SCSI Host Bus Adapter (HVD)
- HP A5159A PCI Dual Port FWD SCSI Host Bus Adapter (HVD)
- HP A5149A PCI to Ultra 2 SCSI Host Bus Adapter (LVD Single Port)
- HP A5150A Dual Channel PCI to Ultra 2 SCSI Host Bus Adapter (LVD Dual Port)
- HP A6828A Ultra 160 SCSI Adapter (LVD Single Port)
- HP A6829A Ultra 160 SCSI Adapter (LVD Dual Port)

ATDD supports the following HP SCSI host adapters:

- HP-PB FWD SCSI-2 Host Adapter (HP 28696A) with firmware revision 3543 or later (ATDD Precision Bus drivers)
- HP PCI Ultra SCSI Host Bus Adapter (HP A4800A, HVD) (ATDD PCI Bus Drivers

- HP PCI Dual Port FWD SCSI Host Bus Adapter (HP A5159A, HVD) (ATDD PCI Bus Drivers)
- HP PCI-X Dual Port Ultra320 SCSI Host Bus Adapter (HP A7173A, LVD) (ATDD PCI Bus Driver)

ATDD supports the following HP Fibre Channel adapter:

- HP A5158A Fibre Channel Mass Storage adapter (ATDD PCI Bus Drivers)
- HP A6795A Fibre Channel PCI Tachyon XL2 host bus adapter (ATDD PCI Bus Drivers)
- HP AB378A/B and AB379A/B PCI-X Fibre Channel Adapter (ATDD PCI Bus Drivers)
- HP A6826A PCI-X Fibre Channel Adapter (ATDD PCI Bus Drivers)

Notes:

1. ATDD does not support tape devices that are attached to HSC/GSC bus architectures.
2. For attachment of Enterprise Tape System 3590 and IBM Magstar MP devices to HP rp8400 (formerly V-Class) platforms, a feedthrough SCSI terminator, which attaches to the A4800A Host Bus Adapter, is required. This feedthrough SCSI terminator is available as FC 9798 for Enterprise Tape System 3590 and FC 5098 for the 3575 Tape Library Dataserver or MP 3570 C model tape subsystem. Attach the male end of the feedthrough SCSI terminator to the host adapter and connect the host end of the standard FC 51xx SCSI cable to the feedthrough terminator. Terminate the SCSI bus at the last device on the bus as usual.
3. The low level FC-Tape protocol standard is not supported in the HP A5158A HBA firmware and users should restrict their SAN configurations. The jobs will be aborted with HP-UX A5158A HBA configurations when errors are detected in the SAN.

   The HP A5158A host adapter is not compliant with the Fibre Channel Protocol for SCSI, second version (FCP-2), dated 17 November 2000, a draft proposed by the American National Standard of Accredited Standards Committee (NCITS).
4. Using a single Fibre Channel host bus adapter (HBA) for concurrent tape and disk operations is not recommended. Tape and disk devices require incompatible HBA settings for reliable operation and optimal performance characteristics. Under stress conditions (high I/O rates for tape, disk, or both) where disk and tape subsystems share a common HBA, stability problems have been observed. These issues are resolved by separating disk and tape I/O streams onto separate HBAs and using SAN zoning to minimize contention. IBM is focused on assuring server and storage configuration interoperability. It strongly recommends that your implementation plan includes provisions for separating disk and tape workloads.
5. The HP A6826A HBAs need the Qlogic firmware level B.11.11.02 or later. If the adapter is directly connected to a 3592 without a switch, then the topology on the drive port needs to be set to L-port.

The most recent supported HBA information can be found at:

http://www-01.ibm.com/servers/storage/support/config/ess/index.jsp

Software Requirements

The following software is required and supported by ATDD:

- HP-UX operating system version 10.20, 11.0, 11i v1, 11i v2, and 11i v3.

- SCSI and FC Host Bus Adapter (HBA) is supplied by HP. Refer to HP's documentation for the SCSI and FC adapter to determine which patch is required.

The latest driver information can be found at ftp://ftp.software.ibm.com/storage/devdrvr/HPUX.

Attention: To install ATDD in HP-UX 11i v3, some HP-UX patches must be applied first on the system. For the detailed information, refer to the atdd.Readme file at ftp://ftp.software.ibm.com/storage/devdrvr/HPUX/11i_PCI/.

Software Interface to the Device Driver

The IBM Tape and Medium Changer Device Driver for HP-UX provides the following standard HP-UX entry points for IBM tape systems:

Open
This entry point is driven by the *open* system function call.

Write
This entry point is driven by the *write* system function call.

Read
This entry point is driven by the *read* system function call.

Close
This entry point is driven explicitly by the *close* system function call and implicitly by the operating system at program termination.

IOCTL
This entry point is driven by the input/output control (*ioctl*) system function call. It provides a set of tape device, medium changer device, and SCSI specific operations. It allows HP-UX applications to access and control the features and attributes of IBM SCSI tape subsystems programmatically.

For programming information, refer to the *IBM Tape Device Drivers Programming Reference*, GA32-0566.

Installation and Configuration Instructions

The following sections describe the installation and configuration for three types of HP-UX devices.

Installation and Configuration for PCI Drivers

This chapter describes how to install, configure, and uninstall the IBM Tape and Medium Changer Device Driver for HP-UX (ATDD) for PCI-based systems. By default, ATDD automatically configures all supported IBM tape drives that are attached and powered On when the system starts. ACDD, the changer part of ATDD, does not configure IBM Medium Changer devices by default because many applications use either their own changer drivers or the native *schgr* or *sctl* drivers. In HP-UX 11.0, 11i v1 and v2, the ATDD package includes with atdd tape driver and acdd changer driver dynamic loadable kernel modules (DLKM). In HP-UX 11i v3, the ATDD package consists of five modules in the kernel:

- atdd - static tape driver kernel module
- acdd - static changer driver kernel module
- atdd_leg - static legacy tape driver kernel module
- acdd_leg - static legacy changer driver kernel module
- atdd_core - DLKM core kernel module

Overview of the Installation Process

During and after the ATDD is installed, ACDD, the changer part of ATDD, can be set to configure (CLAIM) all attached IBM Medium Changer devices. Additionally, selected IBM Medium Changer devices can be configured by running the *swinstall* command with the **-x ask=true** command option, or by running */opt/atdd/bin/atdd_claim –c* c in 11.0, 11i v1 and v2. The installation process depends on whether all IBM tape drives are to be configured by ATDD or only selected ones, and whether configured tape drives exhibit default behavior or require specific configuration settings. Additionally, for IBM Medium Changers, installation selection or post-installation configuration can determine whether all IBM Medium Changers should be configured, selectively configured, or **NOT** configured at all.

Preinstallation Considerations

Make sure your system is ready to install ATDD before you begin. Follow these steps to perform the install:

1. Copy the software from the distribution medium to the depot.
2. Check the README file and verify that your system is configured appropriately for installing ATDD.
3. Ensure that the tape drives to be managed by ATDD are powered on.
4. Install and configure the software.

The following describes facts about the command sequences:

- In some of the examples, filenames given on the command-line must be referenced with an absolute path. Using *'pwd'/filename* to reference a file instead of *filename* ensures this.
- All the software distributor commands (for example, *swinstall* or *swcopy*) can be run first with the *-p* flag to preview the command. After observing the preview output, you can reissue the command without the *-p* flag to perform the actual operation.
- The software distributor commands are moderately complex scripts that frequently require several steps. The steps are typically *Selection*, *Analysis*, and *Execution*. Each step may produce useful information and error messages, so it is a good idea to carefully observe the results of the installation process as it occurs.

If you encounter unexpected results during the installation, check the associated log file.

While using the software distributor commands, you may encounter the following error about mounted file systems:

```
ERROR: "hostname/": One or more filesystems that appear in the
filesystem table are not mounted and cannot be mounted.
```

Many commands (for example, *swinstall* and *swremove*) attempt to mount all file systems in the */etc/fstab* file at the start of the analysis phase. This ensures that all listed file systems are mounted before proceeding. This also helps ensure that files are not loaded into a directory that may be below a future mount point, but it often causes an error with NFS mounted file systems.

This error can be overridden using *'-x'* to set the *mount_all_filesystems* option to FALSE. When this option is used, the command finishes with a warning that indicates no attempt is made to mount all file systems. For example:
 # swinstall -x mount_all_filesystems=false -x ask=true -x autoreboot=true atdd

```
WARNING: "hostname:/": There will be no attempt to mount filesystems
that appear in the filesystem table.
```

Create the Drive Configuration File, (Optional)

If you are not using the standard device driver defaults, you must create a configuration file that directs the device driver on how to customize driver behavior for particular IBM TotalStorage and System Storage devices. If all configured devices are to use the device driver configuration defaults, it is not necessary to create a configuration file before installing the driver for ATDD.

The Drive Configuration File is required if:

- You are running a level of ATDD before level 1.7.1.0 on HP-UX 11.0.
- You do not use the device driver configuration defaults.

If you do not need to create a Drive Configuration File, skip to the instructions in “Power Off the Tape Drives” on page 69.

If you do need to create a Drive Configuration File, consult the following instructions.

The configuration file is named *sbin/init.d/atdd* and has the following syntax:

```
ATDD_global_parameter=value
ATDD_device_parameter[index]=value
```

Note:

- Blank lines and lines starting with # are ignored.
- No spaces may appear within each entry.
- No trailing comments may appear on a variable definition line.

Determine the Device Hardware Path for IBM Tape Drives: To determine the device hardware path for IBM tape drives with a SCSI attachment, perform the following steps:

1. Run *ioscan* to determine the hardware path (i.e., the SCSI adapter) the tape drive is connected to.

   # ioscan -f -C ext_bus
2. Record the hardware path entry of the adapter:

   For example: *0/10/0/0*
3. Determine the SCSI target address of the tape drive.

   HWPATH = ***adapterpath.drivetargetaddress***.0 (tape drive)

   For a tape drive at SCSI target address 3, the device hardware path is:

   0/10/0/0.3.0

To determine the device hardware path for IBM tape drives with the Fibre Channel attachment, perform the following steps:

1. Run *ioscan* to determine the hardware path (i.e., the Fibre Channel) the tape drive is connected to.

   # ioscan -f -C tape
2. Record the hardware path entry of the Fibre Channel:

   For example: HWPATH = *0/5/0/0.8.0.255.7.9.0*

Create the Hardware Path Entry: You must tell the driver which hardware path you want to attach to if you are not using the installation defaults for driver behavior. Create an entry in the */etc/rc.config.d/atdd* configuration file for each device you want the driver to attach. For example:

ATDD_HWPATH[0]=*0/10/0/0.1.0*
ATDD_HWPATH[1]=*0/10/0/0.6.0*
ATDD_HWPATH[2]=*0/5/0/0.8.0.255.6.14.0*

Note: This example shows that three devices will be CLAIMED by the ATDD. If you have a single 3590 device operating in one LUN mode, then you would have only a single ATDD_HWPATH entry. If you have four tape devices attached, four ATDD_HWPATH entries are required in the configuration file.

Create the Device Specific Configuration Entries (Optional): The ATDD is shipped with default settings for all configuration parameters. To change these settings, an entry is made in the configuration file assigning an appropriate value to the desired configuration variable. Each device may have a different value for each configuration variable. The index number associated with the configuration variable associates the parameter setting with the device at the hardware path with the same index.

`Example 1:`

ATDD_IMMEDIATE[0]=1

This parameter allows application control to return from the device before the rewind completes for the device at hardware path *0/10/0/0.1.0* (based on the hardware path entry in “Create the Hardware Path Entry” on page 68).

`Example 2:`

ATDD_TRAILER[1]=1

This parameter allows write operations after an early end-of-tape warning for the device at hardware path *0/10/0/0.6.0*. Refer to Table 9 on page 84 and Table 10 on page 85 for a description of all configuration parameters.

Note: If you experience difficulties with your tape device, check the */etc/rc.log* for errors and correct the problem.

Power Off the Tape Drives

When the ATDD software is initially installed, it attaches itself to all tape drives that are in the CLAIMED state as shown by entering the command:

 # ioscan -fk -C tape

Before you continue, make sure all devices CLAIMED with the preceding command are devices you want this device driver to manage.

To list device file names in the */dev* directory and its subdirectories, enter:

 # ioscan -funC tape

| In HP-UX 11i v3, you can also enter:

| **# ioscan -funNC tape**

To remove a tape drive from the CLAIMED state so it is not configured by this driver, power Off the tape drive and run *ioscan* without the ′*-k*′ argument as follows:

 # ioscan -f -C tape

Install the Driver Using the ATDD Installation Script

An installation script (*InstallAtdd)* is provided to automate driver installation and perform some checking functions. It copies the latest version of the driver to the software depot and installs the latest driver version.

To install ATDD with the script, download the install script from the ftp site, then run the install script.

When the installation script runs, the customer is prompted with the question, ″Would you like to activate the Medium Changer support by ATDD (Y/N)?″. Enter ″Y″ to allow the ATDD to configure all attached IBM Medium Changer devices, or answer ″N″ to allow the HP native changer driver to configure these Medium Changer devices.

Notes:

1. If a previous version of ATDD is installed on your system, uninstall it before attempting to install the latest version. Refer to “Uninstalling the software” on page 74.
2. The system reboot is required to install ATDD in HP-UX 11i v3.

To install ATDD manually, follow the steps in “Install Drivers Manually.”

Install Drivers Manually

Installing the drivers manually requires the steps detailed in the following sections.

1. “Copy the Software to the Software Depot” on page 70.
2. “Install the ATDD Software” on page 70.
3. If a previous version of ATDD is installed on your system, uninstall it before attempting to install the latest version. Refer to “Uninstalling the software” on page 74.

Copy the Software to the Software Depot:

Attention: If you do not copy the ATDD software into a depot, you will not be able to uninstall the product readily.

Copy the appropriate driver to the Software Depot. For example:

 # swcopy -p -s /driver_location/atdd.x.x.x.x atdd (preview option)
 # swcopy -s /driver_location/atdd.x.x.x.x atdd

Use *swlist* to verify that the ATDD software is in the depot:

 # swlist -d atdd

Install the ATDD Software: When the software is in the depot and only the appropriate drives are powered on, the ATDD software can be installed to the root file system using the HP-UX *swinstall* command. If the target root file system is the primary, the ATDD software is configured automatically upon installation.

If the target is an alternate root file system, then the ATDD software is not configured automatically.

Note: If an earlier version of the product is already installed on the target root file system, the existing version is replaced. This is true even if the version already installed is more recent than the version being installed.

The following commands install ATDD from the depot to the default root file system by issuing the *swinstall* command with the **ask** options set as follows:

 # swinstall -p -x ask=true atdd (preview option)
| **# swinstall -x ask=true atdd** in HP-UX 11.0, 11i v1, and v2
| **# swinstall -x autoreboot=true -x ask=true atdd** in HP-UX 11i v3

Running the *swinstall* command with the **ask** option set to TRUE will prompt the customer, ″Would you like to activate the Medium Changer support by ATDD (Y/N)?″. Enter ″Y″ to allow the ATDD to configure all attached IBM Medium Changer devices, or answer ″N″ to allow the HP native changer driver to configure these Medium Changer devices.

You can use *swlist* to list the software installed on the default root file system as follows:

 # swlist atdd

You can verify correct installation to the default root file system with the *swverify* command as follows:

 # swverify atdd

After the installation is completed, run the *ioscan* command to list the tape drives found:

```
# ioscan -fknC tape
Class I  H/W Path                Driver     S/W State       H/W Type       Description
===========================================================================================
tape  0  0/3/0/0.5.0             atdd       CLAIMED         DEVICE         IBM 03590E11
   /dev/rmt/0m   /dev/rmt/0mn   /dev/rmt/c4t5d0BEST   /dev/rmt/c4t5d0BESTn
   /dev/rmt/0mb  /dev/rmt/0mnb  /dev/rmt/c4t5d0BESTb  /dev/rmt/c4t5d0BESTnb
tape  1  0/3/0/0.6.0             atdd       CLAIMED         DEVICE         IBM 03590B11
   /dev/rmt/1m   /dev/rmt/1mn   /dev/rmt/c4t6d0BEST   /dev/rmt/c4t6d0BESTn
   /dev/rmt/1mb  /dev/rmt/1mnb  /dev/rmt/c4t6d0BESTb  /dev/rmt/c4t6d0BESTnb
tape  4  0/5/0/0.1.23.232.0.0.2  atdd       CLAIMED         DEVICE         IBM ULT3580-TD1
   /dev/rmt/4m   /dev/rmt/4mn   /dev/rmt/c6t0d2BEST   /dev/rmt/c6t0d2BESTn
   /dev/rmt/4mb  /dev/rmt/4mnb  /dev/rmt/c6t0d2BESTb  /dev/rmt/c6t0d2BES
```

If you have enabled ACDD, run the following *ioscan* command to view the medium changers found:

```
# ioscan -fnkC autoch
Class   I  H/W Path                Driver S/W State  H/W Type   Description
=====================================================================
autoch  0  0/3/0/0.5.1             acdd   CLAIMED    DEVICE     IBM   03590E11
     /dev/rmt/0chng
autoch  2  0/6/0/0.1.17.255.5.10.1 acdd   CLAIMED    DEVICE     IBM   03590E11
   /dev/rmt/2chng
autoch  4  0/5/0/0.1.23.232.0.0.1  acdd   CLAIMED    DEVICE     IBM   ULT3583-TL
    /dev/rmt/4chng
```

In HP-UX 11i v3, run `# ioscan -funNC tape` or `autoch` to display the tape devices configured with ATDD and ACDD drivers. Run `#ioscan -funC tape` or `autoch` to show the tape devices bound with the legacy atdd_leg and acdd_leg drivers.

Configure all IBM Tape Drives/Media Changers with ATDD/ACDD: By default, ATDD automatically configures all supported IBM tape drives that are attached and powered on when the ATDD package is installed. The ACDD driver does not configure IBM Medium Changer devices by default. To configure all IBM Medium Changer devices on the system during the install of ATDD package, run the *swinstall* command with the *-x ask=true* command option and answer "Yes" to allow the ACDD to claim these devices.

To configure all IBM Medium Changer devices on the system after the install of the ATDD package:
- In HP-UX 11.0, 11i v1 and v2, run the command:
  ```
  # /opt/atdd/bin/atdd_claim -c
  ```
- In HP-UX 11i v3, do the following:
  1. Bind each changer device with the ACDD driver:
     ```
     # ioscan -b -M acdd -H hw_path
     ```
  2. Reboot the system to configure the changer devices:
     ```
     # shutdown -ry now
     ```

To remove the claim with ACDD after the next reboot:
- In HP-UX 11.0, 11i v1 and v2, run the command:
  ```
  # /opt/atdd/bin/atdd_claim -u
  ```
- In HP-UX 11i v3, do the following:
  1. Bind each changer device with a changer driver:
     ```
     # ioscan -b -M acdd -H hw_path
     ```
  2. Reboot the system to unconfigure the changer devices:
     ```
     # shutdown -ry now
     ```

Configure Selected IBM Tape Devices with ATDD/ACDD in HP-UX 11.0, 11i v1 and v2:

Attaching Selected IBM Tape Devices to ATDD/ACDD During Auto Booting: To claim a specific tape device with atdd/acdd during reboot, edit the "Claiming Device" section in the file of /sbin/init.d/atdd.

For example, to attach IBM 3590E11 medium changer with acdd driver:
1. To determine the HW path:
   ```
   # ioscan -fnk|grep IBM
        autoch      0  0/3/0/0.5.1   schgr    CLAIMED   DEVICE     IBM   03590E11
   ```
2. Comment out the command line(s) in the init script of /sbin/init.d/atdd:
   - For tape drives:
     ```
     /opt/atdd/bin/atdd_claim -t
     ```
   - For medium changers:
     ```
     /opt/atdd/bin/atdd_claim -c
     ```
3. Add a command line in the "Configuring Device" section in /sbin/init.d/atdd:
   ```
   /opt/atdd/bin/atdd_claim -H 0/3/0/0.5.1
   ```

Attaching Selected IBM Tape Devices to ATDD/ACDD Dynamically: To dynamically attach selected IBM tape device after the ATDD installation, run the following commands.

1. To determine the HW path of the device:
```
# ioscan -fnk|grep IBM
 autoch      0  0/3/0/0.5.1   schgr    CLAIMED   DEVICE     IBM   03590E11
```
2. To claim the device with ATDD/ACDD:
```
# /opt/atdd/bin/atdd_claim -H 0/3/0/0.5.1
```

Note: The dynamic attachment may be lost after the system reboots.

Attaching Selected IBM Tape Device with a Particular Driver: Use the system *ioscan* and *rmsf* commands to force a non-atdd/acdd driver to bind an IBM tape device. For example, force bind the IBM 3590E media changer with the HP native changer driver schgr with these steps:

1. Determine the HW path of the device:
```
# ioscan -fnk |grep IBM
tape       0  0/3/0/0.5.0   atdd      CLAIMED    DEVICE       IBM    03590E11
autoch     0  0/3/0/0.5.1   acdd      CLAIMED    DEVICE       IBM    03590E11
```
2. Remove the I/O configuration from the system:
```
# rmsf -H 0/3/0/0.5.1
```
3. Force binding:
```
# ioscan -M schgr -H 0/3/0/0.5.1
```
4. Create the device special files:
```
# insf -H 0/3/0/0.5.1
```
5. Verify the force binding:
```
# ioscan -fnk |grep IBM
tape       0  0/3/0/0.5.0   atdd      CLAIMED    DEVICE       IBM    03590E11
autoch     0  0/3/0/0.5.1   schgr     CLAIMED    DEVICE       IBM    03590E11
```

Note: The force binding will be lost after the system reboots. To keep the binding to a selected device with a particular driver after a reboot, add these command-lines in steps 2–4 in ″Claiming Device″ section in /sbin/init.d/atdd file. For example,
```
# Configuring Device Section:
# add specific device claim here, for example
# /opt/atdd/bin/atdd_claim -H 0/3/0/0.5.1
# /opt/atdd/bin/atdd_claim -H 0/6/0/0.1.30.255.0.0.1
rmsf -H 0/3/0/0.5.1
ioscan -M schgr -H 0/3/0/0.5.1
insf -H 0/3/0/0.5.1
```

Configure Selected IBM Tape Devices with ATDD/ACDD in HP-UX 11i v3: By default, atdd driver automatically configures all supported IBM tape drives that are attached and powered on when the ATDD package is installed. The acdd driver does not configure IBM medium changer devices by default. All of IBM medium changers are claimed if you use acdd and acdd_leg as the changer drivers during ATDD are installed. To attach a selected IBM tape device with a particular driver, run the system ioscan command to force a non-ATDD/ACDD driver to bind an IBM tape device, and then reboot the system. For example, to force bind the IBM 3592E05 tape drive with the HP native tape driver estape, perform these steps:

1. Determine the HW path of the device:
```
# ioscan -fNk | grep IBM
tape   1  64000/0xfa00/0x7  atdd  CLAIMED  DEVICE  IBM 03592E05
```

2. Force the binding:

```
# ioscan -b -M estape -H 64000/0xfa00/0x7
```

3. Reboot the system:

```
# shutdown -ry now
```

4. Create the device special files if it does not exist after the system bootup:

```
# insf -H 64000/0xfa00/0x7
```

5. Verify the force binding:

```
# ioscan -fNk | grep IBM
tape   1  64000/0xfa00/0x7  estape  CLAIMED  DEVICE  IBM 03592E05
```

Uninstalling the software:

Attention: Do not try to uninstall the ATDD software by simply deleting the files that make up the ATDD fileset. This causes a number of inconsistencies on the system. It is best to use the *swremove* command.

To uninstall ATDD (from the root file system):

```
# swremove atdd in HP-UX 11.0, 11i v1 and v2
# swremove -x autoreboot=true atdd in HP-UX 11i v3
```

Other Administrative Tasks: To determine whether the current version of ATDD is installed or configured on the default root file system:

```
# swlist -a state atdd
```

To determine the versions of ATDD stored in the default depot:

```
# swlist -d atdd
```

To remove ATDD from the depot:

```
# swremove -d atdd
```

If more than one level of ATDD exists in the depot, explicitly specify the level to remove it. For example:

```
# swremove -d atdd,r=3.5.0.0
```

To view the fileset of ATDD installed on the system:

```
# swlist -l file atdd
```

To display and manage ATDD kernel modules and kernel-tunable parameters, run kmadmin, kmsystem, kmtute and kmadmin commands in HP-UX 11.0 and 11i v1 and kcmodule, kctune commands in HP-UX 11i v2 and v3, or go to the “Kernel Configuration” in sam utility. Refer to the system manual for more detail.

For an example, print out the ATDD driver module status entering the command:

```
# kcmodule -d atdd
Module  State   Cause      Description
atdd    static  explicit   Advanced Tape Device Driver
```

Installation and Configuration of Precision Bus Drivers

This chapter describes how to install, configure, and uninstall the IBM Tape and Medium Changer Device Driver for HP-UX (ATDD) for HP Precision Bus (HP-PB) systems.

Preinstallation Considerations

Make sure your system is ready to install ATDD before you begin. Follow these steps to perform the install:

1. Copy the software from the distribution medium to the depot.
2. Check the README file and verify that your system is configured appropriately for installing ATDD.
3. Ensure that the tape drives to be managed by ATDD are powered on.
4. Install and configure the software.

Note the following facts about the command sequences described in this section:

- In some of the examples, filenames given on the command-line must be referenced with an absolute path. Using *'pwd'/filename* to reference a file instead of *filename* ensures this.
- All the software distributor commands (for example, *swinstall* or *swcopy*) can first be run with the *-p* flag to preview the command. After observing the preview output, you can reissue the command without the *-p* flag to perform the actual operation.
- The software distributor commands are moderately complex scripts that frequently require several steps. The steps are typically *Selection*, *Analysis*, and *Execution*. Each step may produce useful information and error messages. It is a good idea to observe carefully the results of the installation process as it occurs.

If you encounter unexpected results during the installation, check the associated log file.

While using the software distributor commands, you may encounter the following error about mounted file systems:

```
ERROR: "hostname/": One or more filesystems that appear in the
filesystem table are not mounted and cannot be mounted.
```

Many commands (for example, *swinstall* and *swremove*) attempt to mount all file systems in the */etc/fstab* file at the start of the analysis phase. This ensures that all listed file systems are mounted before proceeding. This also helps ensure that files are not loaded into a directory that may be below a future mount point, but it often causes an error with NFS mounted file systems.

This error can be overridden using *'-x'* to set the *mount_all_filesystems* option to FALSE. When this option is used, the command finishes with a warning indicating that no attempt is made to mount all file systems. For example

 # swinstall -x mount_all_filesystems=false -x autoreboot=true atdd

```
WARNING: "hostname:/": There will be no attempt to mount filesystems
that appear in the filesystem table.
```

Create the Configuration File

A configuration file must be created that tells the device driver how to customize your configuration. The file is named *etc/rc.config.d/atdd.cfg* and has the following syntax:

```
ATDD_global_parameter=value
ATDD_device_parameter[index]=value
```

Note:

- Blank lines and lines that start with # are ignored.
- No spaces can appear within each entry.

- No trailing comments can appear on a variable definition line.

Determine the Device Hardware Path for Enterprise Tape System 3590: To determine the device hardware path, follow these steps:

1. Run *ioscan* to determine which SCSI adapter the 3590 drive is connected to.

   # ioscan -f -C ext_bus

2. Record the hardware path entry of the adapter:

   for example, 56/40

3. Determine the SCSI target address of the 3590 drive. The panel on the front of the 3590 drive displays the selected target address.

   HWPATH = ***adapterpath.drivetargetaddress***.0 (3590 one LUN mode)

   or

   HWPATH = ***adapterpath.drivetargetaddress***.0
   HWPATH = ***adapterpath.drivetargetaddress***.1 (3590 random 2 LUN mode)

   For a 3590 at SCSI target address 3 in random 2 LUN mode, the device hardware paths are:

   56/40.3.0
   56/40.3.1

Determine the Device Hardware Path for Magstar MP 3570 Tape Subsystem: The Magstar MP 3570 product family consists of stand alone tape drives (Models B00 and C00) and library models with one or two tape drives. Models B01, B02, B11, B12, C01, C02, C11, and C12 support a 20 cartridge, random access cartridge loader and are known as library units. To control the operation of the library mechanism, SCSI Medium Changer commands are issued to LUN 1 of one or both of the drive ports depending on the library configuration.

Base configuration is intended for attachment of one or two drive libraries on a single SCSI bus. Control of the library is accomplished by sending Medium Changer commands to LUN 1 of drive 1 (the drive on the left as viewed from the rear of the library). The hardware path definitions present in the *atdd.cfg* file must include a definition for drive 1 LUN 1 for operation of the library mechanism.

Models B02, B12, C02, and C12 can operate in a split configuration where each drive and a subset of the cartridge slots are viewed as an independent library. In this case, a definition for LUN 1 for each drive must be present in the *atdd.cfg* file to control each logical library independently. Typically, independent SCSI buses are used for operation of each logical library.

To determine the device hardware path, follow these steps:

1. Run *ioscan* to determine which SCSI adapter the Magstar MP library drive is connected to.

   # ioscan -f -C ext_bus

2. Record the hardware path entry of the adapter:

   for example, *56/40*

3. Determine the SCSI target addresses of the Magstar MP drives. The panel on the front of the Magstar MP library displays the selected target addresses.

   HWPATH = ***adapterpath.drivetargetaddress***.0 (Magstar MP drive)
   HWPATH = ***adapterpath.drivetargetaddress***.1 (Magstar MP library changer)

For a Magstar MP library with a single drive at target address 3 in base configuration, the device hardware paths are:

56/40.3.0
56/40.3.1

Determine the Device Hardware Path for Magstar MP 3575 Tape Libraries: Each drive in a 3575 library can support an automation path for the logical library of which it is a member by sending SCSI Medium Changer commands to LUN 1 of the drive port. The 3575 Model L06 supports one logical library. The other 3575 tape libraries (Models L12, L18, L24, and L32) can support two or three logical libraries depending on the specific model. The first drive in each logical library must have LUN 1 configured to operate the automation for that logical library (drive 1 for logical library 1, drive 3 for logical library 2, and drive 5 for logical library 3). Other drives may have LUN 1 configured, but this is optional.

LUN 1 control of the library mechanism can be configured for any drive by way of the front panel on the library. Conceivably, LUN 1 could be configured for two drives in the same logical library, but this is unnecessary in most instances. In that case, the automation conforms to the SCSI-3 Primary Commands standard for multiple port behavior (except for the handling of reservations).

Access to a logical library is on a first-come, first-served basis for hosts accessing the same logical library on different ports. Each port can accept commands while the library is in use by another host by way of another port. The 3575 deviates from the SCSI-3 specification in that device and element reservations issued on one port apply only to those hosts that access that port. Initiators connected to the same logical library through an alternate port are not subject to those reservations.

Special files for operation of the 3575 automation are based on the hardware path definitions present in the *atdd.cfg* file. Target device LUN 1 also needs to be enabled at the library operator panel for drives other than the first drive in each logical library if they are used to accept automation commands. LUN 1 for the first drive in each logical library is mandatory and is enabled automatically by the library firmware.

To determine the device hardware path, follow these steps:

1. Run *ioscan* to determine which SCSI adapter the Magstar MP library drive is connected to.

   # ioscan -f -C ext_bus
2. Record the hardware path entry of the adapter:

   for example, *56/40*
3. Determine the SCSI target addresses of the Magstar MP drives. The panel on the front of the 3575 library displays the selected target addresses.

   HWPATH = ***adapterpath.drivetargetaddress***1.0 (Magstar MP drive)
   HWPATH = ***adapterpath.drivetargetaddress***2.0 (Magstar MP drive)
   HWPATH = ***adapterpath.drivetargetaddress***1.1 (3575 library control path)

For a 3575 library with two drives in one logical library (drive 1 at SCSI target 3 and drive 2 at SCSI target 4), the device hardware paths are:

```
56/40.3.0
56/40.4.0
56/40.3.1
```

Determine the Drive Hardware Path for IBM 3580 Ultrium Tape Drive, 3581 Tape Autoloader with SCSI Attachment: To determine the hardware path for the 3580 Tape Drive and the 3581 Tape Autoloader with SCSI attachment, follow this procedure:

1. Run *ioscan* to determine which SCSI adapter the IBM Ultrium drive or autoloader is connected to:

   # ioscan -f -C ext_bus
2. Record the hardware path entry of the adapter, for example, 0/4/0/0.
3. Determine the SCSI target address of the IBM Ultrium drive (3580) or the drive in the IBM 3581 Ultrium Tape Autoloader. The switch on the rear of the IBM 3580 Ultrium Tape Drive displays the selected SCSI target address.

   The LCD panel on the front of the unit can display the drive (and medium changer SCSI address) for the IBM 3581 Ultrium Tape Autoloader.

   HWPATH=***adapterpath***.***drivetargetaddress***.0 (3580 drive)

   or

   HWPATH=***adapterpath***.***drivetargetaddress***.0 (3581 drive)

   For an IBM 3581 Ultrium Tape Autoloader at SCSI target address 3, the drive device hardware path is: 0/4/0/0.3.0

Determine the Drive Hardware Paths for IBM Ultrium Tape Libraries with SCSI Attachment: To determine the hardware path for the 3582, 3583, and 3584 Tape Libraries with SCSI attachment, follow this procedure:

1. Run *ioscan* to determine which SCSI adapter the IBM Ultrium Tape Library is connected to:

   # ioscan -f -C ext_bus
2. Record the hardware path entry of the adapter, for example:

   1/8/0/0
3. Determine the SCSI target addresses of the IBM Ultrium drives in the Tape Library. The LCD panel on the front of the unit can display the drive (and medium changer) SCSI addresses.

   HWPATH=***adapterpath***.***drivetargetaddress***.0 (drive 1)

   HWPATH=***adapterpath***.***drivetargetaddress***.0 (drive n)

The drive hardware paths for an IBM Ultrium 3583 Tape Library with a drive at SCSI ID 3 and SCSI ID 5 are:

```
1/8/0/0.3.0
1/8/0/0.5.0
```

Determine the Drive Hardware Paths for IBM 3580 Ultrium Tape Drive, IBM Ultrium Tape Libraries with Fibre Channel Attachment: To determine the hardware path for the 3580 tape drive and the 3582, 3583, and 3584 tape library with fibre channel attachment, follow this procedure:

1. Run *ioscan* to determine which hardware path the IBM Ultrium drive is connected to:

   # ioscan -f -C tape
2. Record the hardware path entry of the IBM Ultrium Fibre Channel drive. For example,

```
0/5/0/0.8.0.255.7.9.0     (3580 stand alone tape drive)
0/8/0/0.8.0.0.0.0.2                 (3583 drive 1)
0/8/0/0.8.0.0.0.0.3                 (3583 drive 2)
```

Create the Hardware Path Entry for Ultrium Tape Drives: If devices are to be configured with settings other than the defaults, entries defining the hardware device paths must be placed in the */etc/rc.config.d/atdd* configuration file. Create an entry for each device that requires further configuration settings. The format for the entries is *HW_PATH[index]=DeviceHardwarePath*. The *index* is used to identify the device for control of the configuration settings in the next section.

For example:

ATDD_HWPATH[0]=0/4/0/0.1.0 (3580 stand alone SCSI drive)
ATDD_HWPATH[1]=1/10/0/0.5.0 (3581 SCSI drive 1)
ATDD_HWPATH[2]=1/10/0/0.6.0 (3581 SCSI drive 2)
ATDD_HWPATH[3]=0/5/0/0.0.0.255.7.9.0 (3580 stand alone FC drive)
ATDD_HWPATH[4]=0/8/0/0.8.0.0.0.0.2 (3583 SCSI drive 1 with SDG)
ATDD_HWPATH[5]=0/8/0/0.8.0.0.0.0.3 (3583 SCSI drive 2 with SDG)

This example shows that four devices are CLAIMED by the ATDD. The first device, 3580 Stand Alone SCSI Tape Drive at SCSI target address 1, LUN 0, has *index=0*. The second device, 3581 with two 3580 SCSI Tape Drives inside at SCSI target addresses 5 and 6, LUN 0, has *index=1 and 2* . The third device, 3580 Stand Alone FC Tape Drive at target 9, LUN 0, has *index=3*. The last one is 3583 with SAN Data Gateway (SDG) Module and two 3580 SCSI Tape Drives at target 0, LUN 2 and 3 (*index=4 and 5)*.

Create the Hardware Path Entry: You must tell the driver which hardware path you want to attach to. Create an entry in the */etc/rc.config.d/atdd.cfg* configuration file for each device you want the driver to attach. For example

ATDD_HWPATH[0]=*56/40.1.0*
ATDD_HWPATH[1]=*56/40.6.0*

Note: This example shows that two devices are CLAIMED by ATDD. If you have a single 3590 device operating in one LUN mode, then you would have only a single ATDD_HWPATH entry. If you have three tape devices attached, three ATDD_HWPATH entries are required in the configuration file. For 3590 devices configured in random 2 LUN mode, two hardware path entries are required. One is needed for the drive (LUN 0), and one is needed for the ACF (LUN 1).

At least one hardware path entry is required in the configuration file.

Create the Device Specific Configuration Entries (Optional): The ATDD is shipped with default settings for all configuration parameters. To change these settings, an entry is made in the configuration file assigning an appropriate value to the desired configuration variable. Each device may have a different value for each configuration variable. The index number associated with the configuration variable associates the parameter setting with the device at the hardware path with the same index.

`Example 1:`

ATDD_IMMEDIATE[0]=1

This allows application control back from the device before rewind is complete for the device at hardware path *56 / 40.1.0* (based on the hardware path entry in "Create the Hardware Path Entry" on page 79).

`Example 2:`

ATDD_TRAILER[1]=1

This allows write operations after an early end-of-tape warning for the device at hardware path *56 / 40.6.0*. Refer to Table 9 on page 84 and Table 10 on page 85 for a description of all configuration parameters.

Note: If you experience difficulties with your tape device, check the */etc/rc.log* for errors and correct the problem.

Power Off the Tape Drives

When the ATDD software is initially installed , it attaches itself to all tape drives that are in the CLAIMED state as shown by entering the command:

 # ioscan -fk -C tape

Before you continue, make sure all devices that report CLAIMED with the preceding command are devices you want this device driver to manage.

To list device file names in the /dev directory and its subdirectories, enter:

 # ioscan -funC tape

To remove a tape drive from the CLAIMED state so it is not be configured by this driver, power Off the tape drive and run *ioscan* without the *'-k'* argument as follows:

 # ioscan -f -C tape

Install the Drive Using the ATDD Installation Script

An installation script (*Install Atdd*) is provided to automate driver installation and perform some checking functions. It copies the latest version of the driver to the software depot and installs the latest driver version.

To install ATDD with the script, download the install script from the ftp site, then run the install script.

When the installation script runs, the customer is prompted with the question, "would you like to activate the Medium Changer support by ATDD (Y/N)?" Enter "Y" to allow the ATDD to configure all attached IBM Medium Changer devices, or answer "N" to allow the HP native changer driver to configure these Medium Changer devices.

Note: If a previous version of ATDD is installed on your system, uninstall it before attempting to install the latest version.

To install ATDD manually, follow the steps in "Install Drivers Manually" on page 70.

Copy the Software to the Software Depot

Attention: If you do not copy the ATDD software into a depot, you will not be able to uninstall the product readily.

 # swcopy -p -s /driver_location/atdd.x.x.x.x atdd (preview option)
 # swcopy -s /driver_location/atdd.x.x.x.x atdd

Use *swlist* to verify that the ATDD software is in the depot:

 # swlist -d atdd

Review the ATDD README File

After copying the ATDD software to the depot, use the *swlist* command to view the README file under the directory */opt/OMImag/Readme*.

 # swlist -d -a readme atdd

The README file lists the system configuration requirements, including required system software patches and required tape related firmware versions. The file also documents any changes in the installation, use, and administration of the software that occurred after this documentation was completed. It is therefore very important that you review it before proceeding with the software installation.

Install the ATDD Software

When the software is in the depot and only the appropriate drives are powered on, the ATDD software can be installed to the root file system using the HP-UX *swinstall* command. If the target root file system is the primary, the ATDD software is configured automatically upon installation.

Configuration requires restarting the system and rebuilding the kernel. This requires you to issue the *swinstall* command with the *-x autoreboot* option set to TRUE, as described below.

If the target is an alternate root file system, the ATDD software is not configured automatically.

Note: If an earlier version of the product is already installed on the target root file system, the existing version is replaced. This is true even if the version already installed is more recent than the version being installed.

The following commands install ATDD from the depot to the default root file system:

 # swinstall -p atdd (preview option)
 # swinstall atdd

If you get an error message stating that this product needs to restart the system, reissue the *swinstall* command with the *autoreboot* option set as follows:

 # swinstall -p -x autoreboot=true atdd (preview option)
 # swinstall -x autoreboot=true atdd

You can use *swlist* to list the software installed on the default root file system as follows:

 # swlist atdd

You can verify correct installation to the default root file system with the *swverify* command as follows:

 # swverify atdd

After the installation is completed, run the *ioscan* command to list the tape drives found:

```
# ioscan -fknC tape
Class I  H/W Path                 Driver     S/W State       H/W Type      Description
============================================================================================
tape  0  0/3/0/0.5.0              atdd       CLAIMED         DEVICE        IBM 03590E11
   /dev/rmt/0m   /dev/rmt/0mn   /dev/rmt/c4t5d0BEST   /dev/rmt/c4t5d0BESTn
   /dev/rmt/0mb  /dev/rmt/0mnb  /dev/rmt/c4t5d0BESTb  /dev/rmt/c4t5d0BESTnb
tape  1  0/3/0/0.6.0              atdd       CLAIMED         DEVICE        IBM 03590B11
   /dev/rmt/1m   /dev/rmt/1mn   /dev/rmt/c4t6d0BEST   /dev/rmt/c4t6d0BESTn
   /dev/rmt/1mb  /dev/rmt/1mnb  /dev/rmt/c4t6d0BESTb  /dev/rmt/c4t6d0BESTnb
tape  4  0/5/0/0.1.23.232.0.0.2   atdd       CLAIMED         DEVICE        IBM ULT3580-TD1
   /dev/rmt/4m   /dev/rmt/4mn   /dev/rmt/c6t0d2BEST   /dev/rmt/c6t0d2BESTn
   /dev/rmt/4mb  /dev/rmt/4mnb  /dev/rmt/c6t0d2BESTb  /dev/rmt/c6t0d2BESTnb
```

If you have enabled ACDD, run the following *ioscan* command to view the medium changers found:

```
# ioscan -fnkC autoch
Class   I  H/W Path                 Driver S/W State  H/W Type   Description
=======================================================================
autoch  0  0/3/0/0.5.1              acdd   CLAIMED    DEVICE     IBM   03590E11
     /dev/rmt/0chng
autoch  2  0/6/0/0.1.17.255.5.10.1 acdd   CLAIMED    DEVICE     IBM   03590E11
   /dev/rmt/2chng
autoch  4  0/5/0/0.1.23.232.0.0.1  acdd   CLAIMED    DEVICE     IBM   ULT3583-TL
    /dev/rmt/4chng
```

Adding an IBM Tape Device Using the Currently Installed ATDD

To add an IBM tape device using the currently installed ATDD, perform the following steps:

1. Modify the */stand/system* file, adding a stanza in the form of:

   driver *56/48.3.0* atdd

   with the adapter/drive path for your device.
2. Modify */etc/rc.config.d/atdd.cfg*, adding the hardware path and instance:

   ATDD_HWPATH[#]=*56/48.3.0*

   where # denotes the next instance and the *adapter/drive* path for your device.
3. Build the kernel as root:

   # mk_kernel -o /stand/vmunix -s/stand/system
4. Restart the system:

   # shutdown -ry now

   or

   # reboot
5. After the system is up, run */opt/OMImag/bin/atdd_mksf* to create the new special files for the device.

   atdd_mksf -ti <instance>

   where instance is the number from the *ioscan* output for the newly installed device. This command echoes to console but does NOT create special files. To create the special files, pipe the command to *sh* as follows:

   # atdd_mksf -ti <instance> | sh

Note: A manual process is required for applications using the HP native *spt* or *sctl* drivers as a changer driver (the *sctl* driver for the ext_bus C700 or C720, the *spt* for the other ext_bus). This manual process requires having the right driver in the kernel, possibly a binding statement in the kernel, and making a device file. Related man pages are scsi_ctl (7) and scsi_pt (7).

Other Administrative Tasks

To determine the versions of ATDD currently installed on the default root file system:

 # swlist -a state atdd

To determine the versions of ATDD stored in the default depot:

 # swlist -d state atdd

To view the set of files that is installed with ATDD:

 # swlist -l file atdd

To remove ATDD from the depot:

 # swremove -d atdd

If more than one level of ATDD exists in the depot, explicitly specify the level to remove it. For example:

 # swremove -d atdd,r=4.9.0.0

Supported Configuration Parameters

This chapter describes the supported configuration values for multiple device types being configured on the same system. The configuration parameters are global. Because the configuration parameters are global to all devices, it is necessary to use configuration values that perform reliably on all devices attached to your system. If you are experiencing difficulties, ensure that your driver is configured properly. To check your current configuration, use the *atdd_cfg* program located in */opt/OMImag/bin* for the static PB driver and in */opt/atdd/bin* for the PCI driver. For program usage, execute as follows:

For PB driver:

 # /opt/OMImag/bin/atdd_cfg -h

For PCI driver:

 # /opt/atdd/bin/atdd_cfg -h

The configuration values for the IBM 3570, 3590, 3592, and 3580 tape drives are by default:

Device	SILI	FORCE_NARROW	DISABLE_COPYIN	COPY_THRESHOLD	DENSITY	COMPRESSION	RESERVE
IBM 3570	1	0	0	0	0	1	1
IBM 3590	1	0	0	0	0	1	1
IBM 3592	1	0	0	0	0	1	1
IBM 3580	1	0	-	-	0	1	1

Configuration Parameter Definitions

This section describes the configuration parameters and values. Modifying the default settings is not recommended unless instructed to do so. The ATDD is shipped with default values that allow the most reliable execution across various device types.

Device-Specific Parameters

Some of the configuration parameters are device-specific, while others are driver-specific. Table 9 shows the parameters that are device-specific, what they mean, and their values.

Table 9. Device-Specific Parameter Definitions

Parameter	Meaning	Values
DENSITY	Density to use when writing/reading tape	In hexadecimal
SILI	Suppress Incorrect Length Indicator	• 0=Off (do not suppress) • 1=On (suppress, by default)
FORCE_NARROW (Precision Bus Driver only)	Width at which the driver will negotiate for data transfers	• 0=Wide (16 bit wide data transfers, by default) • 1=Narrow (8 bit wide data transfers)
DISABLE_COPYIN_THRESHOLD (Precision Bus Driver only)	Protect against nonaligned buffers	• 0=Off (check buffers for alignment, by default) • 1=On (use buffers without checking alignment)
COPY_THRESHOLD (Precision Bus Driver only)	Minimum buffer size to start checking for proper alignment	Size in bytes to force alignment check, 0 by default.
BLOCKSIZE	Block Size	Size in bytes, 0 by default.
COMPRESSION	Compression Mode	• 0=Off (do not use compression at drive) • 1=On (use compression at drive, by default)
BUFFERING	Buffering Mode	• 0=Off (do not buffer data) • 1=On (buffer data to hardware buffers, by default)
IMMEDIATE	Immediate Mode	• 0=Off (wait for rewind completion, by default) • 1=On (return before rewind is complete)
TRAILER	Trailer Label Processing	• 0=Off (do not allow writes past early EOT warning) • 1=On (allow writes past early EOT warning, by default)
ERRNO_LEOT	Error Number return for Logical End Of Tape)	Value returned for writes past EOM, 28 by default.
RESERVE	Multiple open for the device	• 1=On (reserve the device after to open, by default) • 0=Off (do not reserve the device after to open) (multiple open support)

Table 9. Device-Specific Parameter Definitions (continued)

Parameter	Meaning	Values
WFM_IMMEDIATE	Immediate Mode for Write File Mark	• 0=Off (wait for write file mark completion, by default) • 1=On (return before write file mark is complete)
PRE_V3_COMPAT	Use to share tapes with blocksize > 256kb with HP-UX 11i v2 or earlier. Supported in HP-UX 11i v3 only.	• 0=Off (always write or read the date using the blocksize defined by user, by default) • 1=On (split the blocksize less than 256 kb during the read or write)

Driver-Specific Parameters

Some of the configuration parameters are device-specific, while others are driver-specific. Table 10 shows the parameters that are driver-specific, what they mean, and their values.

Table 10. Driver-Specific Parameters (Global)

Parameter	Meaning	Values
INSTANCE	ATDD Device Number	(Read Only)
DEVICES	Number of Configured Devices	(Read Only)
MINPHYS (Precision Bus Driver only)	Check/Adjust Transfer Count	• 0=Use OMI minphys • 1=Use HP-UX minphys

Modifying Configuration Parameters

To change a configuration parameter, use the *atdd_cfg* program located in */opt/OMImag/bin* for the PB driver and in */opt/atdd/bin* for the PCI driver. You can update the current value.

- In HP-UX 11.0, 11i v1 and v2:

  For example, if you want to change the COMPRESSION parameter to 0 for the device at hardware path *0/4/0/0.1.0*, do the following:

  # atdd_cfg -g INSTANCE *0/4/0/0.1.0*
  INSTANCE: 2

  The ATDD instance number for this device is returned. You must use this INSTANCE value (1) to set the COMPRESSION value for this device.

  # atdd_cfg -s COMPRESSION 0 2

  Or, to get or modify COMPRESSION by using the hardware path:

  # atdd_cfg -g COMPRESSION 0/4/0/0.1.0

  # atdd_cfg -s COMPRESSION 0 0/4/0/0.1.0
- In HP-UX 11i v3:

  The usage of atdd_cfg command is:

  ```
  atdd_cfg -d special_file [-o parameter [-w value]]
  ```

  All configurable parameters for the specified device are displayed by entering `#atdd_cfg -d special_file only`

  For instance:

```
# /opt/atdd/bin/atdd_cfg -d /dev/rtape/tape14_BEST
DENSITY=0
SILI=1
BLOCKSIZE=0
COMPRESSION=1
BUFFERING=1
REWIND_IMMEDIATE=0
TRAILER=1
SCALING=0
WRITEPROTECT=0
ERRNO_LEOT=28
WFM_IMMEDIATE=0
RESERVE=1
PRE_V3_COMPAT=0
```

To display a specific parameter, run `#atdd_cfg -d special_file -o parameter`

For example,

```
#/opt/atdd/bin/atdd_cfg -d /dev/tape/tape14_BESTn -o COMPRESSION
COMPRESSION=1
```

To set a specific parameter, run `#atdd_cfg -d special_file -o parameter -w value`

For example,

```
# atdd_cfg -d /dev/rtape/tape14_BESTn -o  COMPRESSION -w 0
Success
```

To set a new BOOT default value for the configuration parameter, you **must** modify or add an entry in the *atdd.cfg* configuration file located in */etc/rc.config.d* for the static driver or the atdd configuration file located in */sbin/init.d*.

For example, if you want the COMPRESSION default set to 0 for the device at ATDD_HWPATH[0] or tape14_BESTn, add the following line to your configuration file:

- In HP-UX 11.0, 11i v1 and v2:

  `ATDD_COMPRESSION[0]=0`
- In HP-UX 11i v3:

  `/opt/atdd/bin/atdd_cfg -d /dev/rtape/tape14_BESTn -o COMPRESSION -w 0`

Note: The configuration parameters are prefixed with *ATDD_* in the configuration file. For additional instructions about using the configuration program, use the *-h* option as follows:

 # atdd_cfg -h

Special Files

For each drive configured by ATDD, twelve or thirteen special files are created: twelve special files are generated for a 3590 with ACF set to *Manual*, *Automatic*, *System*, *Accumulate*, *Random*, or *Library* mode. A thirteenth special file is created for independent operation of the ACF, if the 3590 mode is set to Random 2 LUN operation (*/dev/rmt/<instance#>chng*).

ATDD creates the tape device special files in three forms: the standard or long file name, an alternative short file name based on the instance number, and an alternative short file name ***not*** based on the instance number. BEST represents the operational capabilities likely to be required, including the highest density/format and data compression, if supported by the device. Each set of special file names (long and short) contains four special files that have all combinations of *n* and *b* options, as shown in Table 11 and Table 12. For more information, refer to the mt(7) man pages.

Table 11. Special Files

Special File Name	BSD Compatibility	Rewind on Close
/dev/rmt/<instance#>m	No	Yes
/dev/rmt/<instance#>mb	Yes	Yes
/dev/rmt/<instance#>mn	No	No
/dev/rmt/<instance#>mnb	Yes	No
/dev/rmt/s<#>m	No	Yes
/dev/rmt/s<#>mb	Yes	Yes
/dev/rmt/s<#>mn	No	No
/dev/rmt/s<#>mnb	Yes	No
/dev/rmt/<instance#>t<target>d<LUN>BEST	No	Yes
/dev/rmt/<instance#>t<target>d<LUN>BESTb	Yes	Yes
/dev/rmt/<instance#>t<target>d<LUN>BESTn	No	No
/dev/rmt/<instance#>t<target>d<LUN>BESTnb	Yes	No
/dev/rmt/<instance#>chng	N/A	N/A

Table 12. New Special Files in HP-UX 11i v3

Special File Name	BSD Compatibility	Rewind on Close
/dev/rtape/tape#_BEST	No	Yes
/dev/rtape/tape#_BESTb	Yes	Yes
/dev/rtape/tape#_BESTn	No	No
/dev/rtape/tape#_BESTnb	Yes	No
/dev/rchgr/autoch#	N/A	N/A

- In HP-UX 11.0, 11i v1 and v2:

  atdd_lssf in */opt/OMImag/bin* for the static driver and in */opt/atdd/bin* for the DLKM driver lists information about a special file. For example:

  ```
  #atdd_lssf  /dev/rmt/393mnb
  atdd card instance 30 available at address  Major = 79  Minor = 0x1E90C0
  settings = No Rewind, Berkeley, Best Format,
  IBM 03590H1A /dev/rmt/393mnb
  ```
- In HP-UX 11i v3:

  ATDD supports the HP-UX **lssf**, **insf** and **rmsf** commands. To display special file information, enter the **lssf** command. For example,

  ```
  # lssf /dev/rtape/tape14_BESTnb
  atdd Berkeley No-Rewind best density available at address 64000/0xfa00/0x32
  /dev/rtape/tape14_BESTnb

  # lssf /dev/rchgr/autoch1
  acdd section 0 at address 64000/0xfa00/0x21 /dev/rchgr/autoch1
  ```

  If the device special files are not created on the system, use the **# insf** command to generate them.

Persistent Naming Support

Persistent naming support is used to ensure that attached devices will always be configured with the same logical name across system reboots. For this purpose, the HP-UX tape driver creates an additional set of device special files with a prefix of “s” for the configured tape drives in HP-UX 11.0, 11i v1 and v2, such as s55m in the following example.

```
# ioscan -funC tape
Class  I    H/W Path              Driver S/W State    H/W Type         Description
==================================================================================
tape   103  1/0/0/0.97.25.255.2.7.0     atdd     CLAIMED    DEVICE   IBM   03592E05
            /dev/rmt/103m         /dev/rmt/c10t7d0BESTn            /dev/rmt/s55mb
            /dev/rmt/103mb        /dev/rmt/c10t7d0BEST             /dev/rmt/s55mn
            /dev/rmt/103mn        /dev/rmt/c10t7d0BESTb            /dev/rmt/s55m
            /dev/rmt/103mnb       /dev/rmt/c10t7d0BESTnb           /dev/rmt/s55mnb
```

In HP-UX 11i v3, HP introduces the native multi-pathing and path-independent persistent Device Special Files (DSFs) and the auto discovery of devices greatly to enhance the overall manageability. The ATDD does not create the additional set of device special files with a prefix of “s”.

The persistent DSF format for tape devices introduced with the Agile view in HP-UX 11i v3 is:

```
# ioscan -funNC tape
Class     I  H/W Path  Driver S/W State   H/W Type     Description
==================================================================
tape      1  64000/0xfa00/0x7   estape   CLAIMED     DEVICE  IBM 03592E05
                      /dev/rtape/tape1_BEST    /dev/rtape/tape1_BESTn
                      /dev/rtape/tape1_BESTb   /dev/rtape/tape1_BESTnb

# ioscan -funNC autoch
Class     I  H/W Path  Driver S/W State   H/W Type     Description
================================================================
autoch    8  64000/0xfa00/0xa   acdd   CLAIMED     DEVICE  IBM 03584L22
                      /dev/rchgr/autoch8
```

To map a persistent DSF to its legacy DSF, enter the following:

```
# ioscan -m dsf /dev/rtape/tape1_BESTb
Persistent DSF          Legacy DSF(s)
========================================
/dev/rtape/tape1_BESTb   /dev/rmt/c3t15d0BESTb
                         /dev/rmt/c9t6d0BESTb
```

Control Path Failover Support for Tape Libraries

Notes:

1. The library control path failover feature code must be installed prior to enabling the path failover support in the HP-UX ATDD device driver. Refer to “Supported Devices and Feature Codes” on page 14 for what feature code may be required for your machine type.
2. It is a requirement that applications must open the primary path for CPF

Configuring and Unconfiguring Path Failover Support

Path failover support is enabled automatically when the device driver is installed; however, the driver provides a kernel parameter you can use to enable library control path failover. To enable the failover support in the ATDD driver, perform the following steps after installing the driver:

- In HP-UX 11.0, 11i v1 and v2:

1. Set the alternate_pathing parameter:

```
#  /opt/atdd/bin/atdd_failover -a 1
```

2. Reconfigure CPF for the devices:

```
#  /opt/atdd/bin/atdd_claim
```

- In HP-UX 11i v3:

  Run kctune to enable the kernel tunable of atdd_alternate_pathing. ATDD will handle the CPF in the next open.

```
# kctune atdd_alternate_pathing=1
       * The automatic 'backup' configuration has been updated.
       * The requested changes have been applied to the currently
         running configuration.
Tunable                             Value    Expression       Changes
atdd_alternate_pathing  (before)      0              0          Immed
                        (now)         1              1
```

  You can check if the driver has recognized multiple control paths for your library by running:

```
#ioscan -funC autoch
```

If you see .altpath next to one of the special files in HP-UX 11.0, 11i v1 and v2, then you have successfully enabled the control path failover feature for your library. In HP-UX 11i v3, use the # ioscan -m dsf command and the option 53 of "Display Path" in tapeutil to verify that the multiple path is configured by the system and ATDD.

After the path failover support is enabled, it remains set until atdd_claim is run with the alternate_pathing driver parameter set to OFF . The path failover setting is retained even if the system is rebooted. If you want to turn off the control path failover feature in the ATDD driver, perform the following steps:

- In HP-UX 11.0, 11i v1 and v2:

1. Clear the alternate_pathing parameter:

```
#  /opt/atdd/bin/atdd_failover -a 0
```

2. Reconfigure CPF for the devices:

```
#  /opt/atdd/bin/atdd_claim
```

- In HP-UX 11i v3:

  Run kctune to disable the kernel tunable of atdd_alternate_pathing. ATDD will manage the CPF in the next open.

```
# kctune atdd_alternate_pathing=0
       * The automatic 'backup' configuration has been updated.
       * The requested changes have been applied to the currently
         running configuration.
Tunable                             Value  Expression        Changes
atdd_alternate_pathing  (before)      1            1           Immed
                        (now)         0            0
```

Primary and Alternate Paths

When ATDD is loaded into the kernel, the first logical medium changer device that it sees in the system will be the primary path for that medium changer. The other logical medium changers that ATDD attached for the same medium changer will be configured as alternate paths. The device driver supports up to 16 physical paths for a single device.

On HP-UX 11i v3, the HP-UX system always selects the best path for ATDD to open so the primary path is dynamically assigned from the system. ATDD does not provide any .altpath special file.

The primary and alternate path information can be obtained by using the following command on HP-Ux 11.0, 11i v1 and v2:

```
#  ioscan -funC autoch
```

The following is an example of what you might see:

```
autoch    0  0/1/0/0.101.22.255.1.1.1  acdd   CLAIMED     DEVICE       IBM     ULT3582-TL
                            /dev/rmt/0chng
autoch    1  0/1/0/0.101.22.255.1.2.1  acdd   CLAIMED     DEVICE       IBM     ULT3582-TL
                            /dev/rmt/1chng.altpath
```

Querying Primary and Alternate Path Configurations

You can display the primary and alternate path configuration for all devices by running the **ioscan** command, as explained in the previous section.

You can also display the primary and alternate paths configuration for any device by running the following command:

```
#  tapeutil -f /dev/rmt/Xchng -o phs -v
```

where Xchng is the logical name of any device. This command shows specific information for the primary path and all alternate paths, such as the logical name of the device, the attached host bus adapter, the channel id, the target id, the logical unit number under the target, the current enabled status, and how many paths are configured for the device

Disable and Enable Primary and Alternate Paths

When you load the ATDD device driver with the `alternate_pathing` parameter set to ON, by default, all the available paths for a physical device are enabled. If, for some maintenance reason, you need to disable a path and do not want to fail over to this path, you can run the following commands, where the -f flag is for the primary path and the -p flag is for the path that you want to enable or disable.

This will disable the path:

```
#  tapeutil -f /dev/rmt/Xchng -o dis -p /dev/rmt/Xchng -v
```

This will ensure that the path is labeled as ″disabled.″

To enable a path from a disabled state, you can run the following command:

```
#  tapeutil -f /dev/rmt/Xchng -o ena  -p /dev/rmt/Xchng -v
```

This will enable the path.

Data Path Failover and Load Balancing Support for Tape Drives

The HP-UX device driver path failover support will configure multiple physical paths to the same device within the device driver and provides two basic functions:

1. Automatic failover to an alternate physical path when a permanent error occurs on one path.
2. Dynamic load balancing for devices using multiple Host Bus Adapters (HBA). On HP-UX 11i v3, HP-UX system always pickups a best path for ATDD to open; ATDD doesn't perform a dynamic load balancing in open.

Notes:

1. Data Path Failover is not supported when using block sizes greater than 256K, except in HP-UX 11i v3.
2. It is required that applications must open the primary path for Data Path Failover (not in HP-UX 11i v3).
3. Data Path Failover is supported for 3592 tape drives on HP-UX 11.0, 11i v1, v2 and v3, and for Ultrium-3 and Ultrium-4 drives on HP-UX 11i v1, v2 and v3.

Configuring and Unconfiguring Path Failover Support

Path failover support is turned on automatically when the device driver is installed. However, you will need to provide the data path failover feature key and reclaim the devices using the atdd_claim script. The feature key is a string with 16 characters and numbers.

To enable the failover support in the IBMtape device driver software, perform the following steps after installing the driver:

1. Edit the following line in your `/sbin/init.d/atdd` file:

   ```
   DPF_KEYS="1234567890123456; 0987654321123456 "
   ```

   Notes:

   a. "1234567890123456" is an example of a data path failover feature key. If you have multiple 3584 libraries and multiple data path failover feature keys, input your keys as follows:`DPF_KEYS="`*`key1`*`; `*`key2`*`; ..."`
   b. A DPF key is required for Ultrium-3 tape drives only. No DPF key needs to be added into the `/sbin/init.d/atdd` file if you are running the most recent drive code.
2. Save the file, then run the following command in HP-UX 11.0, 11i v1 and v2:

   ```
   #  /opt/atdd/bin/atdd_claim
   ```
3. In HP-UX 11.0, 11i v1 and v2, you can check whether the driver has recognized multiple data paths for your drives by running:

   ```
   #ioscan -funC tape
   ```

If you see a `".altpath"` next to one of the special files, then you have successfully enabled control path failover feature for your library.

In HP-UX 11i v3, use the **# ioscan -m dsf** command and the option 53 of "Display Path" in tapeutil to verify that the multiple path is configured by the system and ATDD.

After path failover support is enabled, it remains set until `atdd_claim` or `kctune` is run with the `alternate_pathing` driver parameter set to OFF. The path failover setting is retained even if the system is rebooted. If you want to turn off the data path failover feature in the ATDD driver, perform the following steps:

In HP-UX 11.0, 11i v1 and v2:

1. Clear the `alternate_pathing` parameter. (You can ignore the message about rebooting the system.)`#  /opt/atdd/bin/atdd_failover -a 0`
2. Reconfigure DPF for the devices:
   `#  /opt/atdd/bin/atdd_claim`

In HP-UX 11i v3:

Run kctune to disable the kernel tunable of atdd_alternate_pathing. ATDD will handle the DPF in the next open.

```
# kctune atdd_alternate_pathing=0
       * The automatic 'backup' configuration has been updated.
       * The requested changes have been applied to the currently
         running configuration.
Tunable                                         Value  Expression  Changes
atdd_alternate_pathing  (before)      1              1          Immed
                                      (now)          0             0
```

If you want to turn the data path failover feature back on in the ATDD driver, perform the following steps:

In HP-UX 11.0, 11i v1 and v2:

1. Set the `alternate_pathing` parameter. (You can ignore the message about rebooting the system.)
   `#  /opt/atdd/bin/atdd_failover -a 1`
2. Reconfigure DPF for the devices:
   `#  /opt/atdd/bin/atdd_claim`

In HP-UX 11i v3:

Run kctune to enable the kernel tunable of atdd_alternate_pathing. ATDD will handle the DPF in the next open.

```
# kctune atdd_alternate_pathing=1
       * The automatic 'backup' configuration has been updated.
       * The requested changes have been applied to the currently
         running configuration.
Tunable                                         Value  Expression  Changes
atdd_alternate_pathing  (before)      0              0          Immed
                                      (now)          1             1
```

Primary and Alternate Paths

When ATDD is loaded into the kernel, the first logical device that it sees in the system will be the primary path for that medium changer. The other logical paths that ATDD attached for the same tape drive will be configured as alternate paths. The device driver supports up to 16 physical paths for a single device. In HP-UX 11i v3, the HP-UX system always pickups a best path for ATDD to open so the primary path is dynamically assigned from the system. ATDD does not provide any .atlpath special file.

The primary and alternate path information can be obtained by issuing the following command except in HP-UX 11i v3:

```
#  ioscan -funC tape
```

The following is an example of the output for this command:

```
/home/root# ioscan -funC tape
Class  I  H/W Path                 Driver S/W State  H/W Type     Description
=============================================================================
tape   0  0/1/0/0.101.22.255.1.1.1  atdd   CLAIMED    DEVICE    IBM ULT3580-TD3 /dev/rmt/0m
tape   1  0/1/0/0.101.22.255.1.2.1  atdd   CLAIMED    DEVICE    IBM ULT3580-TD3 /dev/rmt/1m.altpath
```

Querying Primary and Alternate Path Configuration

You can display the primary and alternate path configuration for all devices by running `ioscan`, as explained “Primary and Alternate Paths.”

You can also display the primary and alternate path configuration for any device by running the following command:

```
#  tapeutil -f /dev/rmt/Xm -o phs -v
```

Where: `Xm` is the logical name of any device

This command shows specific information for the primary path and all alternate paths, such as the logical name of the device, the attached host bus adapter, the channel ID, the target ID, the logical unit number under the target, the current enabled status, and the number of paths configured for the device.

Disable and Enable Primary and Alternate Paths

Once you load the ATDD device driver with the `alternate_pathing` parameter set to ON, by default, all the available paths for a physical device are enabled. If for some maintenance reason you need to disable a path and do not want to fail over to this path, then run the following commands:

1. To disable the path:
   ```
   #  tapeutil -f /dev/rmt/Xm -p /dev/rmt/Xm.altpath -o dis -v
   ```
2. To make sure the path is labeled as ″disabled″
   ```
   #  tapeutil -f /dev/rmt/Xm -o phs -v
   ```

The string `/dev/rmt/Xm` is the primary path and `/dev/rmt/Xm.altpath` is the alternate path you want to disable.

To enable a path from a disabled state, run the following command:

```
#  tapeutil -f /dev/rmt/Xm -p /dev/rmt/Xm.altpath -o ena -v
```

Tape Utility Program (tapeutil)

This chapter describes how to install, uninstall and use the IBM HP-UX Tape Utility Program, *tapeutil*.

Command Sequence Information

Take note of the following facts about the command sequences described in this section:

- In some of the examples, filenames given on the command-line must be referenced with an absolute path. Using *′pwd′/filename* to reference to a file instead of *filename* ensures this.
- All the software distributor commands (for example, *swinstall* or *swcopy*) can first be run with the *-p* flag to preview the command. After observing the preview output, reissue the command without the *-p* flag to perform the actual operation.
- The software distributor commands are moderately complex scripts that frequently require several steps. The steps are typically *Selection*, *Analysis*, and *Execution*. Each step may produce useful information and error messages, so it is a good idea to carefully observe the results of the installation process as it occurs.

If you run into unexpected results during the installation, check the associated log file.

Product Requirements

The following hardware and software components are required and supported by IBM and are necessary for implementation of the *tapeutil*.

Tapeutil Implementation

The *tapeutil* program contains utility program implementations for operation in the following HP-UX platform environments:

- HP Precision Bus - HP-UX 10.20 and 11.00 - versions of this program are identified by levels *tapeutil.hpux.3.x.x.x*
- HP PCI Bus - If you are using the ATDD version 1.7.7.2 or later, on HP-UX 11.00, use *tapeutil.hpux.4.0.0.0* or later. If you are using the ATDD version 3.0.0.1 or later, on HP-UX 11i, use *tapeutil.hpux.4.0.0.0* or later. Otherwise, use tapeutil level *tapeutil.hpux.3.x.x.x.*

Hardware Requirements

Refer to "Hardware Requirements" on page 64.

Install tapeutil

Installing the product manually requires the three steps detailed in the following sections:

1. "Copy the Software to the Software Depot" on page 94
2. "Review the tapeutil README File" on page 94
3. "Install tapeutil" on page 94

Copy the Software to the Software Depot

Copy the software from its location to the Software Depot.

 # swcopy -p -s /tapeutil_location/tapeutil.hpux.x.x.x.x tapeutil (preview option)
 # swcopy -s /tapeutil_location/tapeutil.hpux.x.x.x.x tapeutil

Use *swlist* to verify that the *tapeutil* software is in the depot:

 # swlist -d tapeutil

Review the tapeutil README File

After copying the *tapeutil* software to the depot, use the *swlist* command to view the README file in the directory */opt/tapeutil*:

 # swlist -d -a readme tapeutil

The README file lists the system configuration requirements, including required system software patches and required tape related firmware versions. The file also documents any changes in the installation, use, and administration of the software that occurred after this documentation was completed. Review it before proceeding with the software installation.

Install tapeutil

When the software is in the depot, the *tapeutil* software can be installed to the root file system using the HP-UX *swinstall* command.

The following commands install *tapeutil* from the depot to the default root file system:

 # swinstall -p tapeutil (preview option)
 # swinstall tapeutil

You can use *swlist* to list the software installed on the default root file system as follows:

 # swlist tapeutil

You can verify correct installation to the default root file system with the *swverify* command:

 # swverify tapeutil

Uninstalling tapeutil

Attention: Do not try to uninstall the ATDD software by simply deleting the files that make up the ATDD fileset. This causes a number of inconsistencies on the system. It is best to use the *swremove* command.

To remove the *tapeutil* software from the root file system, enter:

 # swremove -p tapeutil (preview option)
 # swremove tapeutil

Other Administrative Tasks

To determine the version of *tapeutil* currently installed on the default root file system:

 # swlist -a state tapeutil

To determine the version of *tapeutil* stored in the default depot:

 # swlist -d state tapeutil

To view the set of files installed with *tapeutil*:

 # swlist -l file tapeutil

To remove *tapeutil* from the depot:

 # swremove -d tapeutil

Using the tapeutil Program

A Tape and Medium Changer Utility Program called *tapeutil* is provided with the IBM Tape and Medium Changer Device Driver for HP-UX and installed in the */usr/bin* directory. The *tapeutil* program fulfills several purposes:

- The program provides the following service aids for IBM tape systems:
  - Query Device Type/Verify Device Attachment
  - Query Device Serial Number
  - Query Device Microcode Level
  - Force Device Diagnostic Dump
  - Store Device Diagnostic Dump
  - Download Device Microcode

- The program provides a menu driven test tool for exercising or testing IBM tape and medium changer devices with a full suite of supported operations:
  - Reading/Writing Data
  - Tape Motion Commands
  - Setting/Displaying Device Information/Status
  - Mounting/Demounting Cartridges
  - Cartridge Inventory
- In addition to the menu driven front end, the *tapeutil* program provides a command-line interface, convenient for use in shell scripts.
- The source code for the *tapeutil* program is provided for example purposes and is installed in the */opt/tapeutil* directory during the *tapeutil* package installation. This source code is commented and demonstrates calls to all the supported device driver entry points and *ioctl* commands, giving the application developer a starting point for interfacing to the HP-UX device driver.

Interactive Mode

The *tapeutil* program provides both an interactive mode and a command-line mode. If the *tapeutil* program is called with no command-line parameters, the interactive mode version is started. In the interactive mode, the device to be operated on should first be opened using option 1. Other options may then be selected. The user is prompted for additional information if required for the specific options selected. The results of a command are displayed after it is executed. If an error occurs for the command, error information and device sense data are displayed. The device can be closed using option 2, or it is closed automatically when the *Quit* option is selected. The menu is displayed once automatically when the program is first called. To prevent unnecessary scrolling of the screen, the menu is not displayed again automatically after each command but instead is refreshed only after the *M* (*menu refresh*) command is entered.

These commands can issue the MTIOCTOP *ioctl* with the MT opcodes defined in */usr/include/sys/mtio.h* (MT mode), or issue the STIOC_TAPE_OP *ioctl* with the ST_OP opcodes defined in */usr/include/sys/st.h* (ST mode). For detailed information, refer to the MTIOCTOP and STIOC_TAPE_OP sections in the *IBM Tape Device Drivers Programming Reference*, GA32-0566.

The default for *tapeutil* is the ST mode. Toggle between the MT or ST mode by using option 8. The following commands run in the two modes:
- Write File Mark
- Erase Tape
- Rewind
- Offline
- Forward/Backward Space File
- Forward/Backward Space Record
- Locate End of Data

Command-Line Interface

If command-line parameters are provided when the program is started, the program runs in command-line mode. When in command-line mode, the device is first opened, the specific command is issued, and the device is then closed. The program can be invoked from within a shell script if desired. Results of the operation are displayed only when executed in verbose mode. No information is displayed when not in verbose mode. This is particularly useful for quiet shell script implementations. A completion code, as defined in */usr/include/sys/errno.h*, for the operation requested, is always returned from the program upon exit (in both verbose and quiet mode)

The usage of the *tapeutil* program in command-line mode is as follows:

`tapeutil -f` **`device`** `-o` **`operation`** `[`**`options`**`]`

where **device** is the name of the tape device special file (for example, */dev/rmt/1m*) and **operation** is one of the values listed below. The device special file and the operation are required. The specific **options** associated with a particular operation are indicated below. Parameters enclosed in square brackets are optional. All others are required.

Service Commands

Query Serial Number `tapeutil -f f -o qsn [-w w] [-v]`

Query Microcode Level `tapeutil -f f -o qmc [-w w] [-v]`

Force Dump `tapeutil -f f -o fdp [-w w] [-v]`

Store Dump `tapeutil -f f -o sdp [-w w] [-v] -z z`

Download Microcode `tapeutil -f f -o dmc [-w w] [-v] -z z`

Query Device Type `tapeutil -f f -o chk [-w w] [-v]`

Basic SCSI Commands

Test Unit Ready `tapeutil -f f -o tur [-w w] [-v]`

Inquiry/Inquiry Page `tapeutil -f f -o inq [-w w] [-v] [-t t] [-x x]`

Request Sense `tapeutil -f f -o req [-w w] [-v]`

Log Sense Page `tapeutil -f f -o log [-w w] [-v] [-x x]`

Mode Page `tapeutil -f f -o mod [-w w] [-v] -x x`

Reserve on Close `tapeutil -f f -o mod [-w w] [-v] -x x`

Reserve `tapeutil -f f -o res [-w w] [-v]`

Release `tapeutil -f f -o rel [-w w] [-v]`

Prevent/Allow Medium Removal
`tapeutil -f f -o rem [-w w] [-v]-x x`

Medium Changer Commands

Move Medium
`tapeutil -f f -o mov [-w w] [-v] -s s -d d`

Position To Element
`tapeutil -f f -o pos [-w w] [-v] -s s`

Element Information
`tapeutil -f f -o ele [-w w] [-v]`

Read Element Status with Drive ID
`tapeutil -f f -o dvc[-w w] [-v]`

Inventory `tapeutil -f f -o inv [-w w] [-v]`

To Output the Inventory Data into a File
`tapeutil -f f -o inv -t 1 [-w w] [-v] > filename`

Audit `tapeutil -f f -o aud [-w w] [-v]`

Lock/Unlock Door
`tapeutil -f f -o lck [-w w] [-v] -x x`

Note: Exchange Medium and Initial Element Status Range commands are not supported by IBM TotalStorage and System Storage tape devices.

CPF Commands

Display Paths	`tapeutil -f f -o phs[-v]`
Enable Path	`tapeutil -f f -o ena -p path [-v]`
Disable Path	`tapeutil -f f -o dis -p path [-v]`

Tape Drive Commands

Read `tapeutil -f f -o rea [-w w] [-v] -b b -n n -m m`

Write `tapeutil -f f -o wri [-w w] [-v] -b b -n n -m m [-r r] [-z z]`

Write File Mark	`tapeutil -f f -o eof [-w w] [-v] -c c`
Erase Tape	`tapeutil -f f -o era [-w w] [-v]`
Rewind	`tapeutil -f f -o rew [-w w] [-v]`
Retension	`tapeutil -f f -o ret [-w w] [-v]`
Offline	`tapeutil -f f -o off [-w w] [-v]`
Load/Unload Tape	`tapeutil -f f -o lod [-w w] [-v] -x x`
Forward Space File	`tapeutil -f f -o fsf [-w w] [-v] -c c`
Backward Space File	`tapeutil -f f -o bsf [-w w] [-v] -c c`
Forward Space Record	`tapeutil -f f -o fsr [-w w] [-v] -c c`
Backward Space Record	`tapeutil -f f -o bsr [-w w] [-v] -c c`
Locate End of Data	`tapeutil -f f -o eod [-w w] [-v]`
Report Tape Density	`tapeutil -f f -o den [-w w] [-v]`
Check Device Type	`tapeutil -f f -o chk [-w w] [-v]`
Get Record Size	`tapeutil -f f -o grs [-w w] [-v]`
Set Record Size	`tapeutil -f f -o srs [-w w] [-v] -c c`
Get Device Status	`tapeutil -f f -o gds [-w w] [-v]`
Get Device Information	`tapeutil -f f -o gdi [-w w] [-v]`
Get Media Information	`tapeutil -f f -o gmi [-w w] [-v]`
Get Position	`tapeutil -f f -o gpo [-w w] [-v] -t t`
Set Position	`tapeutil -f f -o spo [-w w] [-v] -t t -x x`
Get Parameter	`tapeutil -f f -o gpa [-w w] [-v] -t t`
Set Parameter	`tapeutil -f f -o spa [-w w] [-v] -t t -x x`
Sync Buffer	`tapeutil -f f -o syn [-w w] [-v]`
Display Message	`tapeutil -f f -o msg [-w w] [-v] -t t -y y1,y1`

Note: Invoking the *tapeutil* program with the **-h** flag (for example, **tapeutil -h**) or **-?** flag (for example, **tapeutil -?**) will display the usage help information.

Flag Description

The supported flags, their meanings, their associated operations, and their acceptable ranges are as follows:

-? Usage Help (stand-alone flag) {no value required}

-b Block Size (rea, wri)

{0 < (block size x blocking factor) < 2097152}

-c Operation Count (eof, fsf, fsr, bsf, bsr, srs) {0–65535}

-d Destination Address (mov)

{device specific, determine range from Element Info}

-f Device Special File Name (always required)

{*/dev/rmt/1st* or similar}

-h Usage Help (stand alone flag) {no value required}

-m Multiples to Read or Write (rea, wri) {0–2097152}

-n Blocking Factor (rea, wri)

{0 > (block size x blocking factor) < 2097152}

-o Operation (always required) {refer to the previous list}

-p The path that will be enabled or disabled

-r Random Seed (wri) {0–65535}

-s Source Address (mov, pos)

{device specific, determine range from Element Info}

-t Type of Parameter Value

- (gpo) {1=logical block, 2=physical block}
- (spo) {1=logical block, 2=physical block}
- (gpa) {1=block size, 2=compression, 3=buffering, 4=immediate, 5=trailer, 6=write protect, 7=acf mode, 8=capacity, 9=sili}
- (spa) {1=block size, 2=compression, 3=buffering, 4=immediate, 5=trailer, 6=write protect, 8=capacity, 9=sili}
- (msg) {1=display msg0, 2=display msg1, 3=flash msg0, 4=flash msg1, 5=alternate msg1/msg2}
- (inq) {0=standard data, 1=page code}
- {0=mt mode, 1=st mode}

-v Verbose Mode (optional for all commands, stand-alone flag)

{no value required, absence of flag means quiet mode}

-w Open Mode (optional for all commands)

{1=read/write, 2=read only (default), 3=write only, 4=append}

-x Parameter Value

- (lck) {1=lock, 2=unlock}
- (lod) {1=load, 2=unload}

- (spo) {0–65535}
- (spa) {0–65535}
- (inq) {0x0–0xFF}
- (log) {0x00–0xFF}
- (mod) {0x00–0xFF}
- (rem) {1=prevent, 2=allow}

-y Messages (msg) {message1,message2}

-z Input/Output File Name
- (*sdp*) {path and name of the file in which to store dump}
- (*dmc*) {path and name of the microcode image file}
- (*wri*) {path and name of the file containing write data pattern}

Notes:
1. For read and write operations, the size of one buffer of data transferred during a single SCSI read or write command is determined by the product of the *Block Size* value and the *Blocking Factor* value. The number of these buffers transferred is determined by *Multiplier* value. The actual total number of bytes transferred is then (Block Size) x (Blocking Factor) x (Multiplier). If the device is set to fixed block mode (block size not equal to zero), the product of *Block Size* and *Blocking Factor* must be a multiple of the device block size setting.
2. For further information on the Get Parameter (*gpa*) and Set Parameter (*spa*) operations, refer to the STIOC_GET_PARM and STIOC_SET_PARM *ioctl* commands in the *IBM Tape Device Drivers Programming Reference*, GA32-0566.
3. All media types for the 3590 tape drive support capacity scaling. Capacity scaling is only supported for the standard 300 GB re-writable Data Cartridge for the 3592 tape drive. Attempting to set capacity scaling that is not supported by a device or the current media loaded will return a failure.
4. If the current microcode on a tape drive is FIPS (Federal Information Processing Standard) code, then a window will display `Warning: The drive is currently using FIPS code.  Do you want to continue with downloading new drive code? [y/n]:`

   Enter `n` to exit without downloading new code; otherwise, enter `y` to continue with the download code procedure.

The following examples should help to demonstrate and clarify the command-line usage of the *tapeutil* program. For all examples, substitute the actual value of the special file associated with the target device.

- To query the serial number of the device:
  ```
  tapeutil -f /dev/rmt/0m -o qsn -v
  ```
- To request inquiry data from the device:
  ```
  tapeutil -f /dev/rmt/0m -o inq -v
  ```
- To request inquiry page data from the device:
  ```
  tapeutil -f /dev/rmt/0m -o inq -t 1 -x 0x03 -v
  ```
- To request log sense page from the device:
  ```
  tapeutil -f /dev/rmt/0m -o log -x 0x00 -v
  ```
- To request mode page from the device:
  ```
  tapeutil -f /dev/rmt/0m -o mod -x 0x02 -v
  ```
- To move a cartridge from cell 32 to the tape drive (16) for TotalStorage 3590:
  ```
  tapeutil -f /dev/rmt/4chng -o mo -s 32 -d 16 -v
  ```

- To write one hundred 64K blocks of data to the tape device:
  `tapeutil -f /dev/rmt/0mn -w 1 -o wri -b 65535 -n 1 -m 100 -v`
- To write two file marks to the tape device:
  `tapeutil -f /dev/rmt/0mn -w 1 -o eof -c 2 -v`
- To rewind the tape device:
  `tapeutil -f /dev/rmt/0mn -o rew -v`
- To read one hundred 64K blocks of data from the tape device:
  `tapeutil -f /dev/rmt/0mn -o rea -b 65535 -n 1 -m 100 -v`
- To report the tape density:
  `tapeutil -f /dev/rmt/0mn -o den -v`
- To prevent tape removal from the tape drive:
  `tapeutil -f /dev/rmt/0mn -o rem -x 1 -v`
- To allow tape removal from the tape drive:
  `tapeutil -f /dev/rmt/0mn -o rem -x 2 -v`
- To read the element status with the drive ID:
  `tapeutil -f /dev/rmt/4chng -o dvc -v`
- To output the inventory information into a file:
  `tapeutil -f /dev/rmt/4chng -o inv -t 1 > inv.txt`

Problem Determination

A set of tools is provided with the device driver to determine if the device driver and the tape device are functioning correctly.

Error Logging

Read the system log file (typically */var/adm/syslog/syslog.log*) and the ATDD log file (*/var/adm/atdd/atdd_d.log*) if you are having problems. The ATDD logs messages to this file which provide information regarding the problem.

Sense Data Logging

When the tape drive responds with CHECK CONDITION status and associated sense keys of 0x1 (Recovery Error), 0x3 (Medium Error), 0x4 (Hardware Error) and 0xB (Aborted Command) for a hardware or medium error, the sense data is logged into the system log file (typically */var/adm/syslog/syslog.log*). The sense data logging
| feature is disabled by default on 11.0, 11i v1 and 11i v2 and enabled by default on
| 11i v3. To enable the support dynamically, run `/opt/atdd/bin/atdd_cfg` with the -c option.

```
# /opt/atdd/bin/atdd_cfg -c 1
       * The requested changes have been applied to the currently running system.
Tunable                         Value  Expression  Changes
atdd_log_trace  (before)        0      Default     Immed
                (now)           1         1
```

Support_info Script

Run the *support_info* script, located in the */opt/OMImag/bin* directory for the static driver and in */opt/atdd/bin* for the DLKM driver. This script gathers important system and configuration information. There are several sections with the keyword VERIFY, indicating information that should be verified for correctness.

Log in as userid ″root″ on the system which experienced a problem. Then run the script and redirect the output to an end of a file:

Static:

```
/opt/OMImag/bin/support_info > support_info.out
```

DLKM

```
/opt/atdd/bin/support_info > support_info.out
```

Tracing Facility

Running the ATDD_trace utility, located in the */opt/OMImag/bin* for the static driver and in */opt/atdd/bin* for the DLKM driver, displays debug information. The utility traces can be started at any time before an operation on a tape device. The default debug level is 0x00001003, which shows driver configuration and per instance drive attribute configuration and errors. You can increase the amount of debug information displayed by enabling more trace flags. Issuing *atdd_trace -f 0x1fffffff* turns on every trace.

Enter the following commands to start the trace:

1. First, determine which items need to be traced and set the trace flags using the *atdd_trace* command. For example, to trace everything:

   For the Static driver:**/opt/OMImag/bin/atdd_trace -f 0x1fffffff**

   For the DLKM driver:**/opt/atdd/bin/atdd_trace -f 0x1fffffff**

   The set of trace flags is displayed in Table 13.

Table 13. Trace Flags

COMMAND	VALUE
Configuration	0x00000001
General errors	0x00000002
Routine entry points	0x00000004
Returns	0x00000008
TRACE_DBG information	0x00000010
Open/Close routine	0x00000020
Trace motion commands	0x00000040
Read/Write	0x00000100
Strategy	0x00000200
Open/Close states	0x00000400
IOCTLs	0x00000800
Trace device config routines	0x00001000
Trace SCSI level stuff	0x00004000
Configuration	0x10000000
Entry point	0x01000000
General information	0x00400000
Memory	0x00200000

2. Clear the trace buffer:

   For the Static driver:**/opt/OMImag/bin/atdd_trace -c**

   For the DLKM driver:**/opt/atdd/bin/atdd_trace -c**

3. To have trace data displayed immediately in a readable format as it is collected:

   For the Static driver:**/opt/OMImag/bin/atdd_trace -b**

For the DLKM driver:**/opt/atdd/bin/atdd_trace -b**

4. Start the operations on a tape device.
5. To display collected trace data in a readable format at any time:

   For the Static driver:**/opt/OMImag/bin/atdd_trace -d**

   For the DLKM driver:**/opt/atdd/bin/atdd_trace -d**
6. The atdd_trace output can be redirected to place the trace data in a readable format into a file, by entering:

   For the PB driver:**/opt/OMImag/bin/atdd_trace >trace.txt**

   For the PCI driver:**/opt/atdd/bin/atdd_trace >trace.txt**
7. In HP-UX 11i v3, the ATDD trace is logged into syslog of /var/adm/syslog/syslog.log once the kernel tunable of atdd_direct_trace is enabled. To set a value (0 or 1) to atdd_direct_trace, run **# kctune atdd_direct_trace=value**.

Atdd_d Log Daemon

The device driver provides a log daemon (*atdd_d*) facility for the Enterprise Tape System 359x to automatically store drive dumps, to retrieve and store SIM/MIM (Service and Media Information Messages) data, and to log error messages into the */var/adm/atdd*directory. The *atdd_d* log daemon is not automatically started in 11.0, 11i v1 and v2 but not in 11i v3, when the driver is installed.

The following steps document how to start and configure the daemon:

1. Start the daemon by running:

   For the Static driver:

   ```
   #/opt/OMImag/bin/atdd_d
   ```

   For the DLKM driver:

   ```
   #/opt/atdd/bin/atdd_d
   ```
2. Check whether the daemon is running by entering:

   ```
   #ps -ef | grep atdd_d
   ```
3. To view and/or modify the settings for the daemon, enter:

   For the Static driver:

   ```
   # /opt/OMImag/bin/atdd_d -h
   usage: atdd_d [options]
   ```

   For the DLKM driver:

   ```
   # /opt/atdd/bin/atdd_d -h
   usage: atdd_d [options]

      [-d <log directory>]              default is /var/adm/atdd
      [-n <maximum number of dumps>]    default is 10
      [-z <maximum size of a dump>]     default is 1048576
      [-s <maximum size of a log>]      default is 100000
   ```

   For example, to decrease the number of drive dumps to store from 10 to 6 and change the log directory to */tmp/drive_dump*, do the following:

   For the Static driver:

   ```
   #/opt/OMImag/bin/atdd_d -n 6 -d/tmp/drive_dump
   ```

   For the DLKM driver:

   ```
   #/opt/atdd/bin/atdd_d -n 6 -d/tmp/drive_dump
   ```
4. View the *atdd_d.log*, which provides information about the daemon start time, message type (1 for the drive dump, 2 for SIM/MIM data, and 3 for Error message), time stamp, H/W path, and size. Refer to an example in Figure 7 on page 104

page 104.

```
# more /var/adm/atdd/atdd_d.log ↲
Tue Dec 18 14:29:36 2001    Daemon started ↲
Tue Dec 18 14:30:36 2001    Got message type 1, created on Tue Dec 18 14:30:36 2001 ↲
Tue Dec 18 14:30:36 2001    Got dump from device 235, 0x0640c0, path 1/10/0/1.4.0,
                                      size 487624 ↲
Tue Dec 18 14:30:36 2001   dump saved as /var/adm/atdd/dump.00000 ↲
Tue Dec 18 14:31:29 2001   Got message type 1, created on Tue Dec 18 14:31:29 2001 ↲
Tue Dec 18 14:31:29 2001   Got dump from device 235, 0x0640c0, path 1/10/0/1.4.0,
                                      size 487624 ↲
Tue Dec 18 14:31:29 2001   dump saved as /var/adm/atdd/dump.00001 ↲
Tue Dec 18 14:33:50 2001   Got message type 1, created on ↲
Tue Dec 18 14:33:50 2001 ↲
Tue Dec 18 14:33:50 2001   Got dump from device 235, 0x1282c0, path 0/5/0/0.1.17.239.0.8.2,
                                     size 186068 ↲
Tue Dec 18 14:33:50 2001   dump saved as /var/adm/atdd/dump.00002 ↲
Tue Jan 22 16:24:58 2002   Got message type 2, created on Mon Jan 21 13:18:45 2002 ↲
Tue Jan 22 16:24:58 2002   Got SIM/MIM from device 2, 0x040040, path 0/12/0/0.0.0 ↲
Tue Jan 22 16:24:58 2002   0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF ↲
Tue Jan 22 16:24:58 2002   ---------------------------------------------------------------- ↲
Tue Jan 22 16:24:58 2002   31 00 00 44 00 00 61 40 02 30 30 30 30 30 30 30 1..D..a@.0000000 ↲
Tue Jan 22 16:24:58 2002   32 32 32 37 36 32 35 35 36 30 31 30 30 30 33 41 222762556010003A ↲
Tue Jan 22 16:24:58 2002   32 41 00 00 00 00 00 00 30 38 30 30 30 49 42 2A.......08000IB ↲
Tue Jan 22 16:24:58 2002   4D 31 33 2D 30 30 30 30 30 30 30 35 31 30 30 37 M13-000000051007 ↲
Tue Jan 22 16:24:58 2002   30 33 35 39 30 48 31 41                         03590H1A00051007 ↲
Tue Jan 22 16:24:58 2002   Got message type 2, created on Mon Jan 21 14:39:00 2002 ↲
Mon Feb  4 17:06:45 2002   Daemon started ↲
Mon Feb  4 17:06:45 2002   Got message type 2, created on Mon Feb  4 17:02:57 2002 ↲
Mon Feb  4 17:06:45 2002   Got SIM/MIM from device 2, 0x064040, path 1/10/0/1.4.0 ↲
Mon Feb  4 17:06:45 2002       SIM/MIM saved as /var/adm/atdd/simmim.00000
```

A250188

Figure 7. Example of atdd_d.log Output

Note: No entry is made in the log file when the daemon stops. Run #ps -ef | grep *atdd_d* to check if the daemon is stopped.

5. Stop the daemon by using the following commands:

```
Get the daemon process id:     #ps -ef | grep atdd_d
kill the daemon process:       #kill -9 process_id
```

Problems and Solutions

Table 14 describes problems and possible solutions for errors you may encounter with the DLKM driver.

Table 14. Problems, Reasons, and Solutions

Problem	Reason and Solution
The system hangs running rmsf -H on HP-UX 11i.	DLKM Infrastructure does not set the correct flag in the device switch table entry for a DLKM driver that uses. Install HP patch PHKL_29818
ATDD DLKM modules are not loaded in the running kernel. A message in the file */var/adm/syslog/syslog.log* "vmunix: WARNING: modpath: DLKM is not initialized"	The current kernel is not matching the symbol table (DLKM) in */stand/dlkm* Need to rebuild the kernel by running the following commands: **#cd /stand/build** **#/usr/lbin/sysadm/system_prep -s /stand/build/system** **#/usr/sbin/mk_kernel -s /stand/build/system** **#mv /stand/system /stand/system.prev** **#mv /stand/build/system /stand/system** **#kmupdate /stand/build/vmunix_test** **#shutdown -ry now**

Table 14. Problems, Reasons, and Solutions (continued)

Problem	Reason and Solution
A message in the file */var/adm/syslog/syslog.log* "modld: Attempt to load unregistered module "	ATDD DLKM Module's component files are not copied to certain subdirectories of */usr/conf* and */stand* Remove ATDD and try to install it again.
# kmtune | grep atdd kmtune: Cannot open file -- /stand/.kmsystune_lock	The file .kmsystune_lock was removed manually. Use the following command to create the .kmsystune_lock file again: **`# touch /stand/.kmsystune_lock`**
IBM tape devices not claimed in ioscan -fnk display	As "root", verify the drivers loaded in the running kernel using **`` `kmadmin -s` ``** If status is "LOADED" then run **`'atdd_claim -b'`** to claim all IBM devices, then run **`'atdd_claim -t'`** to claim the tape drives, then run **`'atdd_claim -c'`** to claim the changers. If status is "UNLOADED" then run **`'kmadmin -L atdd'`** to load the ATDD, then run **`'kmadmin -L acdd'`** to load the ACDD, then run **`'atdd_claim'`** to claim the tape devices. If status is still "UNLOADED", then check the entries of the files 'atdd' and/or 'acdd' in */etc/loadmods* If no atdd and/or acdd file entries exist, the driver may be statically bound to the kernel or the ATDD package may not be installed. Check the installation running the following command: **`# swlist atdd`**
No special files found in */dev/rmt*	Execute the atdd_mksf and acdd (if you have changers managed by the acdd driver) scripts found in */opt/atdd/bin* **`atdd_mksf and acdd_mksf`** **Note:** The script prints out the commands that are necessary to create the special files. To create the files you must "pipe" the output to shell (sh), using the following commands: **`atdd_mksf | sh`** **`acdd_mksf | sh`**
The special files for a device are duplicates.	Remove all of the special files for the device in */dev/rmt* Create the special files again by running the commands **`atdd_mksf`** or **`acdd_mksf`**
No special files are created by HP Stape and Schgr after installing ATDD	Create the special files by running the command: **`# insf -e`**
Cannot open Special File and the system log has the following message: Invalid SCSI request in data at bit 7 of byte 4	Wrong DENSITY setting. Check the current value by running the command: **`atdd_cfg -g DENSITY`** Try setting value to 0 (zero) by running the command: **`atdd_cfg -s DENSITY 0`**
An attempt to read data times out and returns an error and the system log has the following messages: A SCSI command timed out and was aborted.	Make sure the SILI configuration parameter is "1". This can be checked by running the command: **`atdd_cfg -g SILI`** If the value of SILI = 0, try setting the value to 1, by running the following command: **`atdd_cfg -s SILI 1`**

Table 15 on page 106 describes problems and possible solutions for errors you may encounter with the static driver.

Table 15. Troubleshooting (Problems and Solutions)

Problem	Solution
No special files found in */dev/rmt*	Issue the following commands: **cd /opt/OMImag/bin** **atdd_mksf | sh**
If you are using the PB drivers or are not using the standard device driver defaults with PCI drivers:	Verify that the hardware path for the device is listed in the configuration file: */etc/rc.config.d/atdd.cfg* If the hardware path is missing or incorrect, the driver was not installed properly.
Claimed by HP Stape driver	Run the command: **# lsdev** to check that the atdd is in the current kernel: 1. If the atdd is not in the output, rebuild the kernel or reinstall atdd; 2. If the atdd is in the output: a. Rebuild the system I/O configuration: 1) Remove */stand/ioconfig* and */etc/ioconfig* after to backup the files; 2) Reboot the system; 3) On System Console, enter the command: **# /sbin/ioinit -c** while the autobooting is halted; 4) Run the command: **# /sbin/insf -e** 5) Enter the command: **# ^D** to exit the I/O configuration b. Consider reinstalling the HP-UX operating system if the system is new or this is a new update.
Cannot open Special File. The system log has the following message: Invalid SCSI request in data at bit 7 of byte 4	Wrong DENSITY setting. Check the current value by: **atdd_cfg -g DENSITY <atdd_inst>** Try setting it to 0 (zero): **atdd_cfg -s DENSITY 0 <atdd_inst>**
An attempt to read data times out and returns an error. The system log has the following messages: A SCSI command timed out and was aborted. LLIO: Detected PowerFail (probably caused by SCSI bus or device reset).	Check that the SILI configuration parameter is 1 by executing: **atdd_cfg -g SILI <atdd_inst>** If the SILI parameter is zero, try setting it to 1: **atdd_cfg -s SILI 1 <atdd_inst>**
An attempt to read data times out and returns an error. The system log has the following messages: A SCSI command timed out and was aborted. scsi3: ....date code... scsi3: ....timestamp...	Check that the FORCE_NARROW parameter is valid for this device by executing: **atdd_cfg -g FORCE_NARROW <atdd_inst>** If FORCE_NARROW is 1, try setting it to 0: **atdd_cfg -s FORCE_NARROW 0 <atdd_inst>**

Chapter 5. Linux Tape and Medium Changer Device Driver

| This chapter describes the IBM Linux Tape and Medium Changer Device Driver (lin_tape).

Purpose

| The lin_tape and medium changer device driver is designed specifically to take advantage of the features provided by the IBM tape drives and medium changer devices. The goal is to give applications access to the functions required for basic tape operations (such as backup and restore) and medium changer operations (such as mount and demount the cartridges), as well as to the advanced functions needed by full tape management systems. Whenever possible, the driver is designed to take advantage of the device features transparent to the application.

Data Flow

The software described in this chapter covers the Linux Device Driver (lin_tape device driver) and the interface between the application and the tape device.

Figure 8 illustrates a typical data flow process.

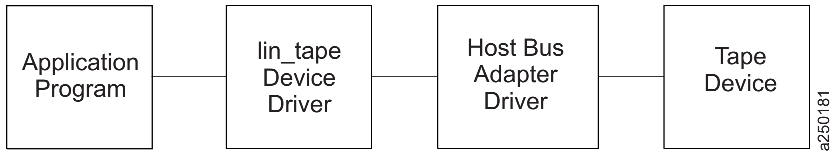


Figure 8. Data Flow for Linux Device Driver (lin_tape)

Product Requirements

More current information on supported hardware and software configurations for lin_tape will be in the README files. The most current information is found in the README files on the IBM Device Drivers **FTP** site, which is located at ftp://ftp.software.ibm.com/storage/devdrvr/Linux. Information specific to certain distributions are in the README files in the distribution-specific directories under the Linux directory.

Information there will include:
- Specifics on Linux distributions and kernel levels
- Supported Host Bus Adapter cards and required firmware and HBA device driver levels
- Other important information that is not included in this manual

Hardware Requirements for Intel and AMD Opteron Processors

One or more of the following processors is required by the lin_tape device driver:
- 32-bit Intel-compatible processors (uniprocessor or SMP) capable of running the Linux operating system
- 64-bit Intel Itanium processors (uniprocessor or SMP) capable of running the Linux operating system
- Intel processors that support Extended Memory 64 Technology (EM64T)
- AMD Opteron processors capable of running the Linux operating system

One or more of the following host bus adapters:
- Adaptec 2940U2W (LVD; Ultrium drives only)
- Adaptec 2944UW (HVD) SCSI Host Bus Adapter
- Adaptec SCSI Adapter 29160 or IBM P/N 19K4646 for IBM System x™ - Single Port LVD
- Adapted SCSI Adapter 39160 - Dual Port LVD
- Adaptec SCSI Adapter 29320 - Single Port LVD
- Adaptec SCSI Adapter 39320 - Dual Port LVD
- LSI Logic SAS3800X SAS Adapters
- IBM SAS Host Bus Adapter Controller Part Number 25R8060

- QLogic QLA2200F, QLA2300F, QLA2310FL, QLA2340L, QLA2342L, QLA2460, and QLA2462 Fibre Channel Adapters
- Emulex LP1150, LP9002L, LP9802, LP10000, LP11000, and LP11002 Fibre Channel Adapters

Hardware Requirements for IBM System p Models

One or more of the following processors is required by the lin_tape device driver:

- IBM low, middle, or high range System p (also known as pSeries) servers running a 64-bit Linux operating system
- IBM POWER5™ servers running a 64-bit Linux operating system

One or more of the following host bus adapters:

- PCI-X DDR Dual -x4 SAS Adapter (FC 5900)
- System p (also known as pSeries) Feature Code 5712 PCI-X Dual Channel Ultra320 SCSI Adapter
- System p (also known as pSeries) Feature Code 5736 (571A) or 1912 PCI-X DDR Dual Channel Ultra320 SCSI Adapter
- System p (also known as pSeries) Feature Code 6228 Fibre Channel Adapter
- System p (also known as pSeries) Feature Code 6239 Fibre Channel PCI-X Adapter
- System p (also known as pSeries) Feature Code 5716 Fibre Channel PCI-X Adapter
- System p (also known as pSeries) Feature Code 5758 and 5759 Fibre Channel PCI-X Adapter
- System p (also known as pSeries) Feature Code 5773 and 5774 Fibre Channel PCI-E Adapter

Hardware Requirements for IBM System z Models

One or more of the following processors is required by the lin_tape device driver running Linux on System z in 31-bit or 64-bit:

- IBM System z 800 (z800) or 900 (z900) models
- IBM System z 890 (z890) or 990 (z990) models
- IBM System z9™ models

One or more of the following host bus adapters with Fibre Channel Protocol support:

- FICON® Express card (feature 2315 or 2318)
- FICON Express card (feature 2319 or 2320)
- FICON Express2 card (feature 3319 or 3320)
- FICON Express4 card (feature 3321, 3322, or 3324)

Note: Using a single Fibre Channel host bus adapter (HBA) for concurrent tape and disk operations is not recommended. Tape and disk devices require incompatible HBA settings for reliable operation and optimal performance characteristics. Under stress conditions (high I/O rates for tape, disk, or both) where disk and tape subsystems share a common HBA, stability problems have been observed. These issues are resolved by separating disk and tape I/O streams onto separate HBAs and using SAN zoning to minimize contention. IBM is focused on assuring server and storage configuration interoperability. IBM strongly recommends that your implementation plan includes provisions for separating disk and tape workloads.

Software Requirements for Intel and AMD Opteron Processors

The lin_tape device driver supports the following Linux distributions:

- Red Hat Enterprise Linux Server
- SUSE LINUX Enterprise Server
- Asianux

Software Requirements for IBM System p Models

The lin_tape device driver supports the following Linux distributions for Linux for System p (also known as pSeries) (64-bit kernel):

- Red Hat Enterprise Linux
- SUSE LINUX Enterprise Server
- Asianux

Software Requirements for IBM System z Models

The lin_tape device driver supports the following Linux distribution for Linux on System z 31-bit and 64-bit:

- Red Hat Enterprise Linux
- SUSE LINUX Enterprise Server

Installation and Configuration Instructions

The lin_tape device driver for Linux is provided in a source *rpm* package. The utility tools for lin_tape are supplied in binary *rpm* packages. They can be downloaded from the following FTP site: ftp://ftp.software.ibm.com/storage/devdrvr/Linux.

The following sections describe installation, configuration, uninstalling, and verification procedures for lin_tape and its utility tools. Refer to Linux documentation for *tar* command information and any Linux distribution supporting *rpm* for *rpm* command information. You must have *root* authority to proceed with the installation of the driver. See the README file at ftp://ftp.software.ibm.com/storage/devdrvr/Linux/README, which contains the latest driver information and supercedes the information in this publication.

Conventions Used

In subsequent pages, you will see file names with x.x.x in them. The x.x.x refers to the version of the driver, which will change as IBM releases new driver levels. Use the actual driver version numbers as you perform the instructions.

Commands that you are to type are indicated with a leading ″>″ character, which indicates the shell prompt. Some of the commands will have a ″—some text″ string after them, for example, ″>cd IBMtapeutil.x.x.x.i386 — to change the directory″. The ″—some text″ items are explanatory comments to the reader and should not be specified when you enter these commands, for example, ″cd IBMtapeutil.x.x.x.i386″ is all you would specify on the previous example.

Components Created During Installation

The lin_tape package consists of the device driver and a number of associated files. Components created during lin_tape installation (from the *rpm* package) are listed in Table 16.

Table 16. Components Created During lin_tape Installation

Component	Description
/lib/modules/(Your system's kernel name)/kernel/drivers/scsi/lin_tape.ko	Device driver module for current kernel version
/usr/bin/lin_taped	lin_taped daemon
/etc/lin_taped.conf	lin_taped daemon configuration file
/usr/share/doc/lin_tape-xxx/ lin_tape_359X.Readme (for Red Hat) */usr/share/doc/packages/lin_tape/ lin_tape_359X.Readme* (for SUSE LINUX)	ReadMe file for lin_tape using IBM TotalStorage devices
/usr/share/doc/lin_tape-xxx/lin_taped.Readme (for Red Hat) */usr/share/doc/packages/lin_tape/ lin_taped.Readme* (for SUSE LINUX)	ReadMe file for lin_taped daemon
/usr/share/doc/lin_tape-xxx/copying (for Red Hat) */usr/share/doc/packages/lin_tape/copying* (for SUSE LINUX)	License documentation for lin_tape

The IBMtapeutil package consists of *IBMtapeutil*, *IBMtapeconfig*, and the source files for *IBMtapeutil*. Components created during lin_tape installation (from the *rpm* package) are listed in Table 17.

Table 17. Components Created During IBMtapeutil Installation

Component	Description
/usr/bin/IBMtapeconfig	Utility to create special files
/usr/bin/IBMtapeutil	Utility and service aid program

Installation Procedure

If lin_tape is already installed on your system, refer to the "Updating Procedure" on page 112 in this chapter. This section assumes you are installing the lin_tape device driver onto a system where it is not currently installed.

If you are installing lin_tape on a system running Linux for S/390® or Linux for zSeries®, ensure that the OpenFCP adapter device driver zfcp is loaded in the kernel. Please refer to the section "Configuring Tape and Medium Changer Devices on IBM System z Models" on page 114 in this chapter for how to configure and install zfcp.

Make sure that the C/C++ development and kernel development packages are installed on your system. To install the lin_tape driver with all the added value of the lin_taped daemon, perform the following steps:

1. Download the appropriate level of the source RPM package to a directory of your choice on the Linux kernel for which you wish to install it.

2. Run rpmbuild --rebuild <filename>, where filename is the name of the RPM file. This will create a binary RPM package for your kernel from the source RPM package.

   For example:

   ```
   >rpmbuild --rebuild lin_tape-1.x.x.x.0-1.src.rpm
   ```

3. Output from the build will be printed to your screen. Near the end of the output, you will find a line that indicates the filename and location of your binary RPM package. For example, a line similar to the following will be output to your screen:

   ```
   Wrote: /usr/src/redhat/RPMS/i386/lin_tape-1.x.x.x.0-1.i386.rpm
   ```

4. To install the lin_tape driver from the binary package, run >rpm -ivh <filename>

   For example:

   ```
   >rpm -ivh /usr/src/redhat/RPMS/i386/lin_tape-1.x.x.x.0-1.i386.rpm
   ```

5. To install the lin_taped daemon, download it to your Linux filesystem and run rpm -ivh on the daemon RPM file.

   For example:

   ```
   >rpm -ivh /usr/src/redhat/RPMS/i386/lin_taped-1.x.x.x.0-rhel5.i386.rpm
   ```

Updating Procedure

Before using this procedure to update your device driver to a different level, use the following command to obtain your existing lin_tape device driver version, if there is at least one tape device attached to your system:

```
> IBMtapeutil -f /dev/IBMtape0 qryversion
```

If your current lin_tape device driver was installed from a *rpm* package previously, you may uninstall the driver first, then install the newer version. For example:

```
>rpm -e lin_tape
>rpmbuild -rebuild lin_tape.x.x.x.i386.rpm
>rpm -i lin_tape.x.x.x.i386.rpm
```

Note that all tape devices that use the lin_tape device driver must be closed and cannot be in use when lin_tape is uninstalled.

Querying the Installed Package

The query is supported for the lin_tape device driver *rpm* package only.

The installed *rpm* package can be queried by running the following commands to display information associated with the package.

To display information about lin_tape:

```
>rpm -qi lin_tape
```

To display the file list, enter the command:

```
>rpm -ql lin_tape
```

To display the states of files in the package, for example, *normal*, *not installed*, or *replaced*:

```
>rpm -qs lin_tape
```

Verifying Installation and Updating

You can run the following command to verify your lin_tape device driver version if there is at least one tape device attached to your system:

```
IBMtapeutil -f /dev/IBMtape0 qryversion
```

Configuring Tape and Medium Changer Devices on Intel-Compatible Systems

If you have not physically attached your tape and medium changer devices to your Linux server, perform the following steps:

1. Halt the Linux server.
2. Power down the Linux server.
3. Physically attach the tape and medium changers to the host bus adapter cards. Consult your hardware documentation for details.
4. Power on the tape and medium changer devices. Verify that they have completed the initialization process.
5. Power on and boot the Linux server.

Note: Hot plugging SCSI devices while the Linux server is running may cause hardware damage.

After the driver software is installed and a tape device is connected to the adapter, the device can be configured and made available for use. Access to the device is not provided until the device is configured.

If your system is attached to an IBM 3583 Tape Library with the integrated router, before installing the Qlogic driver, set the host type of the router to `solaris` and make sure that the logical unit numbers of the control unit, medium changer, and the connected tape drives are contiguous (otherwise, the Qlogic device driver will not recognize all of the attached devices). To view the LUNs of attached devices, log onto the router and use the *fcShowDevs* command. If the LUNs are not contiguous, use the *mapCompressDatabase* command to delete the invalid LUNs and make the valid LUNs contiguous.

If you have the StorWatch™ Specialist installed, you can use the Specialist to do this configuration. For further information about the StorWatch Specialist, refer to the *IBM Storage Area Network Data Gateway Installation and User's Guide*. You can download the guide from the Web at:

http://www.storage.ibm.com/hardsoft/products/sangateway/support/cdr/Document/sdgdoc.htm

When you run the lin_tape daemon, it will create special files under the /dev directory for you.

To configure the tape devices, use the *IBMtapeutil* application program. Choose **Query/Set Tape Parameters** in the menu.

Configuring Tape and Medium Changer Devices on IBM System p Models

Follow the same instructions as documented in the previous section. You will need to configure the Emulex Linux device driver if you have fibre channel tape devices attached to your System p (also known as pSeries) system. For instructions on how to configure the Emulex device driver, refer to the lin_tape_359X_pSeries.ReadMe at ftp://ftp.software.ibm.com/storage/devdrvr/Linux.

Configuring Tape and Medium Changer Devices on IBM System z Models

The fibre channel topology supported for System z is point-to-point and fabric. Please refer to the Linux on System z fibre channel documents for more details on the supported configurations for fibre channel device attachment. The Linux fibre channel adapter device driver zfcp is available in the kernel that supports zSeries Fibre Channel Protocol. The zfcp device configuration methods in 2.6 and 2.4 kernels are different. For 2.6 kernels, refer to appropriate chapter in the Linux on System z document entitled ″Linux on System z: Device Drivers, Features, and Commands″.

For 2.4 kernels, there are three ways to load the zfcp device driver in order to see the attached tape devices.

1. Create a /etc/zfcp.conf file and make a ramdisk to statically attach tape devices into your system. You can use this method only if you have a persistent mapping in a SAN environment. Every time you reboot the system, the zfcp will be automatically loaded and the tape devices can be seen from the system.

   First you need to add the device map into this file. The following is an example of zfcp.conf:

   ```
   0xf1c0 0x1:0x5005076300402733 0x0:0x0000000000000000;\
   0xf1c1 0x1:0x5005076300402733 0x0:0x0001000000000000
   ```

   The zfcp device driver uses the ″map″ module parameter to recognize a physically attached tape device. ″map″ takes the following format:

   ```
   map="<devno><port scsi-id>:<wwpn><unit-scsi-lun>:<fcp-lun>;...."
   ```

   where:

   devno The device number of the host bus adapter (16 bits, see /proc/subchannels). It is ″0xf1c0″ or ″0xf1c1″ in the previous example.

   port scsi-id
   Linux internal SCSI ID assigned to the Fibre Channel port of the SCSI target device (32-bit, must not be 0, must be a unique one-to-one mapping for each World Wide Port Name. It is ″0x1″ in the previous example.

   wwpn World Wide Port Name identifying the Fibre Channel port of the SCSI target device (64-bit). It is ″0x5005076300402733″ in the previous example.

   unit scsi-lun
   Linux internal SCSI Logical Unit Number (32-bit). It is ″0x0″ in the previous example.

   fcp-lun
   Logical Unit Number associated with the SCSI target device (64-bit). In the previous example, ″0x0000000000000000″ is the Logical Unit Number 0, and ″0x0001000000000000″ is the Logical Unit Number 1.

   We recommend, for tape attachment, that each logical unit number be associated with a unique devno. If you use the same devno numbers for several logical units, you should ensure that each <unit-scsi-lun> is unique. After /etc/zfcp.conf is created, run the following commands:

   ```
   >mk_initrd>zipl
   ```

Then, reboot the system. After it is booted up, your tape device should be shown in */proc/scsi/scsi* file.

2. Modify the /etc/modules.conf file to add the zfcp module parameters; then run the ″depmod –A″ and ″modprobe zfcp″ command. Do not use this choice together with the first one, otherwise it will cause conflicts. The zfcp map in /etc/modules.conf always takes higher priority than the map in /etc/zfcp.conf.

   The following example demonstrates the zfcp configuration in */etc/modules.conf*:

   ```
   options zfcp map="\
   0xf1c0 0x1:0x5005076300402733 0x0:0x0000000000000000;\
   0xf1c1 0x1:0x5005076300402733 0x0:0x0001000000000000"
   ```

   The map arguments are the same as the ones listed in for the */etc/zfcp.conf* file.

   After modifying the */etc/modules.conf* file, save and close it. Then run the following command:

   ```
   >depmod -A
   >modprobe zfcp
   ```

   This will install the zfcp device driver and all of its prerequisite kernel modules. Now you can check the file */proc/scsi/scsi* to see if all of the attached tape devices are shown in this file. If not, then check the fibre channel connection, such as the fibre cables, or if the devices are powered on, etc.

   Then run the following commands to install zfcp:

   ```
   >rmmod zfcp
   >modprobe zfcp
   ```

3. Run the ″modprobe zfcp″ command first, then dynamically add a tape device into the system after you physically attach a fibre channel tape device to the switch.

   If you physically attach a tape device on the switch and zfcp is already loaded, you do not need to reboot the Linux system in order to add this entry in the */proc/scsi/scsi* file. The zfcp device driver provides an ″add_map″ proc system entry under the directory */proc/scsi/zfcp* to allow you to dynamically add the device into the system. For example, to add two logical units from the example in Step 2 into the system, you may issue the following commands;

   ```
   > echo "0xf1c0   0x1:0x5005076300402733 0x0:0x0000000000000000;\
      0xf1c1 0x1:0x5005076300402733 0x0:0x0001000000000000" > /proc/scsi/zfcp/add_map
   > echo "scsi add-single-device 0 0 1 0" > /proc/scsi/scsi
   > echo "scsi add-single-device 1 0 1 1" > /proc/scsi/scsi
   ```

   The ″scsi add-single-device″ takes four parameters, corresponding to the four parameters ″scsi″, ″Channel″, ″Id″, and ″Lun″ in the */proc/scsi/scsi* file. The value of ″scsi″ is 0 for the first devno, 1 for the second devno (if it is different from the first devno), and so on. The value of ″Channel″ can start from 0 for each different ″scsi″ value. The value of ″Id″ is the one you use for <unit scsi-lun> in the above mapping. The value of ″Lun″ is the logical unit number of the target device, for example, the last number in the above mapping. Currently, the zfcp device driver does not support dynamically removing the attached devices. If you need to remove the tape devices from the system, do ″rmmod zfcp″. Then you can delete the entry in */etc/modules.conf* and reload zfcp, or reload zfcp first and dynamically add the devices you want. After you have done all the mapping, if you can see all of the attached tape devices in */proc/scsi/scsi*, you have successfully attached those devices to your system. Next you may install the lin_tape device driver. Refer to the ""Installation Procedure" on page 111" section in this chapter for the instructions on how to install lin_tape.

Uninstall Procedure

Note: All tape devices that use the lin_tape driver must be closed and cannot be in use when lin_tape is uninstalled or the uninstall fails.
Run the following command:
`>rpm -e lin_tape ---to remove`

Tape Drive, Media, and Device Driver Parameters

This chapter describes the parameters that control the operating modes of the IBM Linux Tape and Medium Changer Device Driver.

Configuration Parameters

The configuration parameters are used to set the operating mode of the tape drive and device driver when a device is opened. The installation defaults are provided for all parameters initially. Some of these parameters can be queried and set by *IBMtapeutil*, **Query/Set Tape Parameters** in the menu. These parameters are kept on reopen, but are always restored back to the default values when the lin_tape device driver is reinstalled.

The nonchangeable configuration parameters are:
- Autoloading
- Density code
- Emulate autoloader
- Hook word
- Maximum block size
- Minimum block size
- Medium type
- Read SILI bit
- Record space mode
- Volume ID for logging
- Write protect

The changeable configuration parameters are:
- Block size
- Buffered mode
- Capacity scaling
- Compression
- Disable auto drive dump
- Disable SIM logging
- Logging
- Logical write protect
- Maximum SCSI transfer length
- Read past filemark
- Rewind immediate
- Trace
- Trailer labels

Nonchangeable Parameters

The configuration parameters are used to set the operating mode of the tape drive and device driver when a device is opened. The nonchangeable parameters are detailed below.

Autoloading

This parameter enables the autoloading feature of the device driver. It is disabled by default and cannot be changed.

Capacity Scaling

This parameter sets the capacity or logical length of the current tape. By reducing the capacity of the tape, the tape drive can access data faster at the expense of data capacity. Capacity Scaling is not supported currently but might be supported in future releases of lin_tape.

Density Code

This parameter is the density setting for the currently loaded tape. Some tape devices support multiple densities and will report the current setting in this field. It cannot be changed by the application.

Emulate Autoloader

This parameter currently is not supported and should be ignored.

Hook Word

This parameter is not supported in the lin_tape device driver.

Logical Write Protect

This parameter sets or resets the logical write protect of the current tape. This feature is not supported currently but might be supported in future releases of the lin_tape.

Maximum Block Size

This parameter is the maximum block size for the device.

Minimum Block Size

This parameter is the minimum block size for the device.

Medium Type

This parameter is the media type of the current loaded tape. Some tape devices support multiple media types and will report different values in this field.

Read SILI Bit

SILI bit currently is not supported due to limitations associated with the Linux environment. SILI bit support may be enabled in future releases of the lin_tape.

Record Space Mode

This parameter specifies how the device driver operates when a forward or backward space record operation encounters a filemark. Only the SCSI mode is supported by lin_tape. When a forward or backward space record operation is issued to the driver and a filemark is encountered, the device driver returns -1 and the *errno* variable is set to input/output error (EIO). On the forward space operation, the tape is left-positioned after the filemark (the end of tape side of the filemark). On the backward space operation, the tape is positioned before the filemark (the beginning of tape side of the filemark).

Volume ID for Logging

This parameter is the volume ID of the currently loaded tape. lin_tape device driver ignores this field.

Write Protect

This parameter is set to TRUE if the currently mounted tape is logically or physically write protected.

Changeable Parameters

The configuration parameters are used to set the operating mode of the tape drive and device driver when a device is opened. The changeable parameters are detailed below.

Block Size

This parameter specifies the block size used for read and write operations. A value of zero means a variable block size. Any other value is a fixed block size. The installation default is zero (variable length block size). Refer to "Maximum SCSI Transfer Length" on page 119 for additional guidance.

Buffered Mode

This parameter specifies if read and write operations should be buffered by the tape device. The default (recommended) value is TRUE.

Capacity Scaling

This parameter sets the capacity or logical length of the current tape on Enterprise Tape System 3590 or 3592 tape subsystems. By reducing the capacity of the tape, the tape drive can access data faster at the expense of data capacity. Capacity scaling can be set at 100% for the entire tape (which is the default), or set at 75%, 50%, or 25% of the 3590 tape cartridge and more available capacity scaling for the 3592 standard 300 GB rewritable data cartridge. Capacity scaling remains with the tape across mounts until it is changed.

Note:

1. The tape position must be at the start of the tape to change this parameter from its current value.
2. Changing this parameter destroys any existing data on the tape.
3. For 3592 media types, capacity scaling is only supported for the standard 300 GB rewritable data cartridge. Attempting to set capacity scaling that is not supported by a device or the current media loaded will always return 100% and cannot be changed. For example, the 60GB (Economy Data) cartridge for the IBM 3592 cannot be capacity scaled and will always be 100%.

Compression

Hardware compression is implemented in the device hardware. This parameter turns the hardware compression feature On and Off. If compression is enabled, the effective performance can increase, based on the compressibility of the data.

The installation default is On (use compression).

Disable Auto Drive Dump

This parameter is provided in the lin_tape version 1.2.2 or later. It is set to FALSE by default. If it is FALSE and the lin_taped daemon is running and if an error occurs in the drive which creates a drive dump, the lin_tape device driver will automatically retrieve the drive dump and save it under the */var/log* directory by default. You may specify another directory in the */etc/lin_taped.conf* file. Please refer to “Configure and Run lin_taped Daemon” on page 139 in this chapter for details.

Disable SIM logging

This parameter is provided in the lin_tape version 1.2.2 or later. It is set to FALSE by default. If it is FALSE and the lin_taped daemon is running and SIM/MIM data is generated by the drive, the lin_tape device driver will automatically retrieve the data and save it in a formatted text file under the */var/log* directory by default. You may specify another directory in the */etc/lin_taped.conf* file. Please refer to “Configure and Run lin_taped Daemon” on page 139 in this chapter for details.

This capacity is not applicable to IBM Ultrium tape drives.

Logging (Volume Logging)

This parameter turns the volume information logging On or Off. With the lin_tape version 1.2.2 and later, the lin_tape device driver provides this support. It is set to On by default. If logging is On and the lin_taped daemon is running, the lin_tape device driver will retrieve the full log sense data from the drive whenever a tape is unloaded, or the drive reaches a log threshold. The log file will be saved in binary format under the directory */var/log* by default. You may specify another directory in */etc/lin_taped.conf* file. Please refer to “Configure and Run lin_taped Daemon” on page 139 in this chapter for details.

Note: This is volume logging, which is different from error logging. lin_tape provides error logging whenever the lin_taped daemon is running. Please refer to “Configure and Run lin_taped Daemon” on page 139 in this chapter for details on error logging.

Logical Write Protect

This parameter sets or resets the logical write protect of the current tape on Enterprise Tape System 3590 tape subsystems. The three types of logical write protect are: associated protect, persistent protect, and write-once read-many (WORM) protect.

1. Associated protect remains only while the current tape is mounted or associated with the tape drive. It is reset when the tape is unloaded or the tape drive is reset.
2. Persistent protect remains or persists with the tape across mounts until it is reset.
3. WORM protect also remains with the tape across mounts, but unlike persistent protect it cannot be reset on the tape. After a tape is WORM protected, it can never be written on again.

Note: The tape position must be at the start of the tape to change this parameter from its current value.

Maximum SCSI Transfer Length

In the lin_tape drivers with level lower than 3.0.3, the maximum transfer length per device per SCSI command is 262144 bytes (256 KB) by default. Variable block read/write requests with transfer length greater than the maximum transfer length will fail [*errno*: EINVAL]. When a fixed block size has been defined, large write requests are subject to both the granularity of the block size and the maximum transfer length. For example, with a fixed block size of 80000 bytes and maximum transfer length of 262144, a write request for 400000 bytes (5 blocks of 80000 each) is written to tape in two transfers. The first transfer is 240000 bytes (3 blocks) and the second transfer is 160000 (the remaining two blocks). You may increase the maximum transfer length to enhance the data throughput. This can be done either using *IBMtapeutil*, option 48 – **Query/Set Tape Parameters**, or a customized STIOCSETP input/output control (*ioctl*) call. However, setting the transfer length greater than the default 256 KB does not guarantee a noticeable increase in data throughput. Maximum transfer length of 256 KB is highly recommended.

In lin_tape driver with level 3.0.5 or higher and the open source driver lin_tape, the maximum transfer length is defined as the minimum length that the host bus adapter and the tape drive can support. This number is usually greater than 256KB. It can not be changed by the IBMtapeutil, option 48 or the STIOCSETP ioctl call any more.

Read Past Filemark

If this parameter is set to true, when a read operation encounters a filemark, it returns the number of bytes read before encountering the filemark and positions the tape head after the filemark. If the read_past_filemark parameter is set to false, when the read operation encounters a filemark, if data was read, the *read* function returns the number of bytes read, and positions the tape head before the filemark. If no data was read, then the *read* returns 0 bytes read and positions the tape head after the filemark.

This installation default is FALSE.

Rewind Immediate

This parameter sets the immediate bit for rewind commands. If it is set to On, the rewind tape operation executes faster, but the next command takes a long time to finish unless the physical rewind operation has completed. Setting this parameter reduces the amount of time it takes to close a device for a Rewind on Close special file.

The installation default is Off (no rewind immediate).

Trace

This parameter turns the trace facility On or Off. With the lin_tape version 1.2.2 and later, the lin_tape device driver provides this support. It is set to On by default. If trace is On and the lin_taped daemon is running, the lin_tape device driver will retrieve the trace from the driver if trace level is set to 1 or 2 in the */etc/lin_taped.conf* file. The trace file will be saved under the directory */var/log* by default. You may specify another directory in */etc/lin_taped.conf* file. Please refer to “Configure and Run lin_taped Daemon” on page 139 in this chapter for details.

Trailer Labels

If this parameter is set to On, then writing records past the early warning mark on the tape is allowed. The first write operation after detecting the early warning mark fails and the *errno* variable is set to ENOSPC. No data is written during the operation. All subsequent write operations are allowed to continue until the physical end of the volume is reached and *errno* EIO is returned.

If this parameter is set to Off, then writing records past the early warning mark is not allowed. *Errno* variable is set to ENOSPC.

The installation default is On (with trailer labels).

Special Files

After the driver is installed and a device is configured and made available for use, access is provided through the special files. These special files, which consist of the standard Linux special files for devices, are in the */dev* directory.

Special Files for the Tape Device

Each tape device has a set of special files providing access to the same physical drive but providing different attributes. Table 18 shows the attributes of the special files.

Note: The asterisk (*) in *lin_tape** represents a number assigned to a particular device, such as *lin_tape0*.

For tape drives with attached medium changer devices, the *IBMchanger** special file provides a separate path for issuing commands to the medium changer. When this special file is opened, the application can view the medium changer as a separate device. Both the tape and changer special file can be opened at the same time.

Table 18. Special Files for IBM TotalStorage Tape Devices

Special File Name	Rewind on Close
*/dev/lin_tape**	YES
*/dev/lin_tape*n*	NO

Special Files for the Medium Changer Device

After the driver is installed and a medium changer device is configured and made available for use, access to the robotic device is provided through the *IBMchanger* special file in the */dev* directory. The asterisk (*) represents a number assigned to a particular device, such as *IBMchanger0*. The term *IBMchanger* is used for a SCSI medium changer device. The *IBMchanger** special file provides a path for issuing commands to control the medium changer robotic device.

The file descriptor that results from opening the *IBMchanger* special file does not support the following operations:

- Read
- Write
- Open in Append mode
- Commands designed for a tape device

Control Path Failover Support for Tape Libraries

Note: The library control path failover feature code must be installed prior to enabling control path failover support in the Linux lin_tape device driver. Refer to "Supported Devices and Feature Codes" on page 14 to determine which feature code is required for your machine type.

The Linux lin_tape device driver control path failover support will configure multiple physical control paths to the same logical library within the device driver and provide automatic failover to an alternate control path when a permanent error occurs on one path. This is transparent to the running application.

Configuring and Unconfiguring Path Failover Support

Control path failover support is not enabled automatically when the device driver is installed. The Linux lin_tape device driver provides a driver parameter alternate_pathing for you to enable the library control path failover. To enable the failover support in the lin_tape device driver software, you need do the following steps after installing the lin_tape rpm package:

1. lin_taped stop (stop the lin_taped daemon)
2. rmmod lin_tape (unload the lin_tape driver from the memory)
3. Add the following line in your */etc/modules.conf* file for 2.4 kernels or */etc/modprobe.conf*.local file for 2.6 kernels:

   options *lin_tape alternate_pathing=1*
4. depmod
5. modprobe lin_tape (re-load the lin_tape driver into memory)
6. lin_taped (re-start lin_taped daemon)

You can ignore the ″Unresolved symbols in */lib/modules/<your kernel name>/drivers/scsi/lin_tape.ko*″ message after the ″depmod″ command. You can check if the lin_tape driver has recognized multiple control paths for your library by reading the /proc/scsi/IBMchanger file.

- *cat/proc/scsi/IBMchanger*

If your library lists ″Primary″ or ″Alternate″ under ″FO Path″, you have successfully enabled control path failover feature for your library. If it is ″NA″ listed under ″FO Path″, then the control path failover is not enabled. After control path failover support is enabled, it remains set until the lin_tape driver is reloaded with the alternate_pathing driver parameter set to OFF . The path failover setting is retained even if the system is rebooted. If you want to turn off the control path failover feature in the lin_tape device driver, you can perform the following steps:

1. lin_taped stop
2. rmmod lin_tape
3. Delete the following line in your */etc/modules.conf* file:

   options lin_tape alternate_pathing=1
4. depmod
5. modprobe lin_tape
6. lin_taped

Primary and Alternate Paths

When lin_tape is loaded into kernel memory, the first logical medium changer device that lin_tape sees in the system will be the primary path for that medium changer. The other logical medium changers that lin_tape attached for the same medium changer will be configured as alternate paths. The device driver supports up to 16 physical paths for a single device. The primary and alternate path information can be obtained by the following command:

- *cat /proc/scsi/IBMchanger*

The following is an example of a /proc/scsi/IBMchanger file:

- lin_tape version: 3.0.3
- lin_tape major number: 253

Attached Changer Devices:

Number	Model	SN	HBA	FO Path
0	03584L22	IBM1234567	qla2xxx	Primary
1	03584L22	IBM1234567	qla2xxx	Alternate
2	03584L22	IBM1234567	qla2xxx	Alternate

The labeling of a logical device as either a primary or alternate path is for information only, in order to:

- Be able to identify the actual number of physical devices configured on the system and a specific logical device associated with them. There will be only one logical device labeled as the primary path for each physical device. However, there can be multiple logical devices labeled as an alternate path for the same devices.
- Provide information about which logical devices configured on the system have path failover support enabled.

Querying Primary and Alternate Path Configuration

You can display the primary and alternate path configuration for all devices by reading the */proc/scsi/*IBMchanger file, as explained in the above section. You can also display the primary and alternate path configuration for any device by running the following command:

- *IBMtapeutil -f /dev/*IBMchangerx path (IBMchangerx is the logical name of any device)

This command shows specific information for the primary path and all alternate paths, such as the logical name of the device, the attached host bus adapter, the channel id, the target id, the logical unit number under the target, the current enabled status, and how many paths are configured for the device.

Disable and Enable Primary and Alternate Paths

Once you load the lin_tape device driver with the *alternate_pathing* parameter to be ON, by default, all the available paths for a physical device are enabled. If for some maintenance reason you need to disable a path and do not want to fail over to this path, you can run the following commands:

- *IBMtapeutil -f /dev/IBMchangerx* path (to know what number of this path is)
- *IBMtapeutil -f /dev/IBMchangerx* disablepath number ("number" will be the number of the path you want to disable)
- *IBMtapeutil -f /dev/IBMchangerx* path (to make sure the path is labeled as "disabled")

Correspondingly, in the */proc/scsi/IBMchanger* file, the disabled path will be listed as "Disabled" under the "FO Path" column.

To enable a path from a disabled state, you can run the following command:

- *IBMtapeutil -f /dev/IBMchangerx* enablepath number ("number" will be the number of the path you want to enable)

Data Path Failover and Load Balancing Support for Tape Drives

Data path failover support is not enabled automatically when the device driver is installed. The Linux lin_tape device driver provides a driver parameter `alternate_pathing` for you to enable the data path failover.

To enable the failover support in the lin_tape device driver software, you need to perform the following steps after installing the lin_tape *rpm* package.

```
>lin_taped stop (stop the lin_taped daemon)
>rmmod lin_tape (unload the lin_tape driver from the memory
```

If you have IBM 3592 tape drives, add the following line in your /etc/modules.conf file for 2.4 kernels or /etc/modprobe.conf.local file for 2.6 kernels:

```
options lin_tape alternate_pathing=1
```

If you have IBM LTO tape drives, the library needs to have path failover feature code. The data path failover license keys are needed to enable the failover if you are running LTO 2 drives or if you are running LTO 3 drives with old levels of drive code. DPF keys do not need to be added if you are running the latest drive code on LTO 3 or LTO 4 drives.

Add the following line in your /etc/modules.conf file for 2.4 kernels or /etc/modprobe.conf.local file for 2.6 kernels:

```
options lin_tape alternate_pathing=1 dpf_keys="abcdefghijklmnop"
```

"abckdefghijklmnop" is an example of a data path failover feature key. If you have multiple libraries and multiple data path failover feature keys, input your keys as follows:

```
dpf_keys="key1;key2;..."
```

Save the file, then run the following commands:

```
>depmod
>modprobe lin_tape (re-load the lin_tape driver into memory)
>lin_taped (re-start lin_taped daemon)
```

You may ignore the "*Unresolved symbols in /lib/modules/<your kernel name>/drivers/scsi/lin_tape.ko*" message after the *depmod* command. You can check if the lin_tape driver has recognized multiple paths for your tape drive by reading the */proc/scsi/lin_tape* file:

```
>cat /proc/scsi/lin_tape
```

If your tape drive lists "Primary" or "Alternate" under "FO Path", you have successfully enabled data path failover feature for your tape drive. If it is "NA" listed under "FO Path", then the data path failover is not enabled. After the path failover support is enabled, it remains set until the lin_tape driver is reloaded with the *alternate_pathing* driver parameter set to OFF . The path failover setting is retained even if the system is rebooted. If you want to turn off the data path failover feature in the lin_tape device driver, you may do the following steps:

```
>lin_taped stop
>rmmod lin_tape
```

Delete the following line in your */etc/modules.conf* file: options lin_tape alternate_pathing=1

```
>depmod
>modprobe lin_tape
>lin_taped
```

Primary and Alternate Paths

When the lin_tape device driver is loaded into kernel memory with path failover support enabled, the first logic device that lin_tape sees always becomes the primary path. The other logical devices that lin_tape sees will be configured as the alternate paths. The device driver supports up to 16 physical paths for a single device.

The primary and alternate path information can be obtained by the following command:

```
>cat /proc/scsi/lin_tape
```

The following is an example of a /proc/scsi/lin_tape file:

```
lin_tape version: 3.0.3
lin_tape major number: 253
Attached Tape Devices:

Number    Model    SN            HBA                            FO Path

0         03592    IBM1234567    qla2xxx                        Primary

1         03592    IBM1234567    qla2xxx                        Alternate
```

The labeling of a logical device as either a primary or alternate path is for information only to:

- Be able to identify the actual number of physical devices configured on the system and a specific logical device associated with them. There will be only one logical device labeled the primary path for each physical device. However, there may be many (multiple) logical devices labeled as an alternate path for the same devices.
- Provide information about which logical devices configured on the system have path failover support enabled.

Querying Primary and Alternate Path Configuration

You can display the primary and alternate path configuration for all devices by reading the */proc/scsi/lin_tape* file, as explained in the above section. You can also display the primary and alternate path configuration for any device by running the following command:

```
>IBMtapeutil -f /dev/IBMtapex path (IBMtapex is the logical name of any device)
```

This command shows specific information for the primary path and all alternate paths, such as the logical name of the device, the attached host bus adapter, the channel id, the target id, the logical unit number under the target, the current enabled status, and how many paths are configured for the device.

Disable and Enable Primary and Alternate Paths

Once you load the lin_tape device driver with the `alternate_pathing` parameter to be ON, by default, all the available paths for a physical device are enabled. If for some maintenance reason you need to disable a path and do not want to fail over to this path, you may run the following commands:

```
>IBMtapeutil -f /dev/IBMtapex path
(to know what number of this path is)
>IBMtapeutil -f /dev/IBMtapex disablepath number
("number" will be the number of the path you want to disable)
>IBMtapeutil -f /dev/IBMtapex path
(to make sure the path is labeled as "disabled")
```

| Correspondingly, in the */proc/scsi/lin_tape* file, the disabled path will be listed as ″Disabled″ under the ″FO Path″ column.

To enable a path from a disabled state, you may run the following command:

```
>IBMtapeutil -f /dev/IBMtapex enablepath number
("number" will be the number of the path you want to enable)
```

Open Source Device Driver - lin_tape

The lin_tape device driver is the new device driver for the Linux 2.6 kernels to replace the closed-source driver IBMtape. In most respects, it behaves the same as the closed-source IBMtape device driver. This section will discuss significant differences between the IBMtape driver and the lin_tape driver.

IBMtape and lin_tape Comparison

Table 19 compares the names for various components of the IBMtape and lin_tape device drivers.

Table 19. IBMtape and lin_tape Comparison

Component	IBMtape	Lin_tape
Driver name	IBMtape	lin_tape
Module name	IBMtape.ko	lin_tape.ko
Special files	/dev/IBMtape0 /dev/IBMchanger0, etc.	No change
proc entry	/proc/scsi/IBMtape /proc/scsi/IBMchanger	No change
Daemon name	IBMtaped	lin_taped
Daemon configuration file	/etc/IBMtaped.conf	/etc/lin_taped.conf
Daemon trace files	/var/log/IBMtape.trace /var/log/IBMtape.errorlog	/var/log/lin_tape.trace /var/log/lin_tape.errorlog

The IBMtapeutil utility is the same for both the IBMtape driver and the lin_tape driver.

Installation

Installation of the lin_tape driver is the same as for the IBMtape driver, except that ″IBMtape″ should be replaced with ″lin_tape″ in all of the installation instructions. Refer to the section “Installation and Configuration Instructions” on page 110 of this chapter for details.

The lin_tape driver cannot be installed if the IBMtape driver is already installed. If the IBMtape driver is installed, first uninstall the IBMtape driver, and then install the lin_tape driver. With RHEL4 and SLES10, driver removal will also require a reboot of the server, since the IBMtape driver module is ″permanent″ in these distributions.

Driver parameters and special device files

The driver parameters have not changed for the lin_tape driver. However, it is important to note that the module parameters, such as ″alternate_pathing″ and ″dpf_keys″, must now be applied to the lin_tape module, instead of the IBMtape module. For example, in the /etc/modprobe.conf (for Red Hat) or /etc/modprobe.conf.local (for SUSE) file, add the following line for LTO library's path failover:

```
options lin_tape alternate_pathing=1 dpf_keys="abcdefghijklmnop"
```

`abckdefghijklmnop` is an example of a data path failover feature key.

The special device files for the lin_tape driver are the same as for the IBMtape driver. Refer to "Special Files for the Tape Device" on page 121 and "Special Files for the Medium Changer Device" on page 121 of this chapter for details on special device files.

Path Failover Support

Path failover support in lin_tape is the same, except that with the lin_tape driver, failover support is provided through the lin_taped daemon. If the lin_taped daemon is not running, neither control path failover nor data path failover will be attempted. The lin_taped daemon is started automatically when the lin_tape driver is loaded.

To check if the lin_taped daemon is running, run the following command:

```
lin_taped status
```

This will indicate if the lin_taped daemon is running or not. If the /proc/scsi/IBMtape and /proc/scsi/IBMchanger files indicate ″NA″ for ″FO Path″, this indicates that failover support for that device is not enabled. If all other settings are correct, but ″FO Path″ is incorrectly indicating ″NA″, confirm that the lin_taped daemon is running.

For details about the path failover support, refer to the sections "“Control Path Failover Support for Tape Libraries” on page 121″ and "“Data Path Failover and Load Balancing Support for Tape Drives” on page 123″ of this chapter.

lin_taped Daemon

The lin_taped daemon uses the same command-line arguments as the IBMtaped daemon. The lin_taped configuration file is the same as the IBMtaped configuration file, but has been renamed to lin_taped.conf. Refer to "Configure and Run lin_taped Daemon" on page 139 of this chapter for detailed information.

System-Managed Encryption

Device Driver Configuration

Note: System-managed encryption (SME) on Linux requires that the lin_taped daemon is running.

The device driver SME settings can be set for all drives at once using the ″**default_sys_encryption_proxy**″ and ″**default_sys_encryption_write**″ module options.

If no options are specified in the registry, the driver uses the default values for the parameters.

- The default value for **default_sys_encryption_proxy** is 1.

This value causes the device driver to handle encryption key requests, if the drive is set up for system-managed encryption. This value should not need to be changed. A value of 0 causes the device driver to ignore encryption key requests for system-managed encryption drives, and is not desirable.

- The default value for **default_sys_encryption_write** is 2.

  This value causes the device driver to leave the encryption write-from-BOP settings alone. It does not turn on or turn off encryption writing, but instead uses the settings that are already in the drive. If encryption has not been set up previously, then the drive writes unencrypted data. A value of 0 causes the device driver to write unencrypted data. A value of 1 causes the device driver to write encrypted data.

The module options can be specified in the /etc/modprobe.conf or /etc/modeprobe.conf.local files, the same as other lin_tape module parameters.

For example, to turn on SME to write/read encrypted data:

1. Add the following line:

   options lin_tape default_sys_encryption_write=1
2. Then run the following commands:

   ```
   >lin_taped stop
   >rmmod lin_tape
   >depmod
   >modprobe lin_tape (reinstall back)
   >lin_taped (restart the daemon)
   ```

The default settings are used to initialize the settings for all connected drives.

To modify the settings for individual drives, the settings are ″**sys_encryption_write**″ and ″**sys_encryption_proxy**″, and have the same definitions and values as the similarly named ″default″ parameters, except that the settings only apply to individual drives.

These settings are available as part of the sysfs infrastructure. For each drive, there are two files, named ″**sys_encryption_write**″ and ″**sys_encryption_proxy**″, in the /sys/class/lin_tape/{DEVICE}/ directory, where {DEVICE} is the device name, such as IBMtape0. The contents of these files indicate the current setting for the parameter for that particular drive. The setting can be changed by writing a different value for the parameter to the file.

For example, to change the ″**sys_encryption_write**″ setting for IBMtape0 to ON (which has a value of 1), enter the following at a command prompt:

```
echo 1">/sys/class/lin_tape/IBMtape0/sys_encryption_write
```

Note: The driver encryption parameters for individual drives are not persistent between loads of the lin_tape driver. If you remove the lin_tape driver, and then reload it, the individual settings for all drives will be the same as the ″default″ settings in /etc/modprobe.conf or /etc/modprobe.conf.local.

The `ibmekm.conf` file, which contains the configuration that the EKM servers use, is installed in the /etc/ directory when lin_tape is installed. Instructions for modifying this file are found within the file itself.

Querying Tape Drive Configuration

There is an IBMtapeutil command to query the encryption settings of a tape drive. The IBMtapeutil command is menu option 57, “Query Encryption Status”.

The following is an example of the output when the drive is configured for system-managed encryption, with encryption turned on:

```
issuing query encryption status...
encryption capable......Yes
encryption method.......METHOD_SYSTEM
encryption state........ON
```

Testing Data Encryption Configuration and Connectivity

There is an IBMtapeutil command to validate the `ibmekm.conf` file server entries and test tape drive to server connectivity operations. This test can be run using the IBMtapeutil menu option 58, “EKM Test”. The process involves three tests.

1. The first test checks the server configuration defined in the `ibmekm.conf` file and then communication to the configured servers. This test reports back the number of servers available.
2. The second test runs a basic diagnostic that checks the tape drive to server communication and reports success or fail.
3. The third test runs an enhanced diagnostic that checks a key operation between the tape drive and server then reports success or fail.

The following is an example of the output of a successful test:

```
Testing server configuration and connections...
Testing complete, servers available 1
Running basic drive to server encryption test...
Testing complete, completion code 3
Running full drive to server encryption test...
Testing complete, completion code 3
```

If the first server test fails with one of the following errors, then the remaining tests will not be run. Perform the recommended problem determination to resolve the problem.

- “Invalid argument or Operation not supported on device”

  System encryption is not configured for the tape drive in the device driver. Use SMIT to set system encryption to “yes” for the tape drive. If system encryption can not be set then the tape drive is not encryption capable.
- “Can’t assign requested address”

  Either the `ibmekm.conf` file is missing or is invalid. Check and correct.
- “Network is down”

  All configured servers in the `ibmekm.conf` file are not available. Check all servers configured in the `ibmekm.conf` file are currently running on the server IP address and configured correctly.

Tape Utility Program (IBMtapeutil)

Installed with the device driver is a tape utility program (*IBMtapeutil*) that exercises or tests the functions of the tape device and the device driver. It also performs basic tape and medium changer operations. The tape utility program can be used in either the interactive or the command-line mode.

The C source code for the *IBMtapeutil* program contains a sample of the interface to the device driver and the *ioctl* commands supported by the device driver. *IBMtapeutil.c* may be found in the tar file *IBMtapeutil.x.x.x.tar* on the device driver down load ftp site: ftp://ftp.software.ibm.com/storage/devdrvr/Linux/.

Interactive Mode

The interactive mode for the tape utility program can be invoked from the command line using the *IBMtapeutil* command. A list of general subcommands, medium changer subcommands, tape subcommands, and service aid subcommands is displayed. You must open a device before using these commands and operations.

To open a device:

1. Select **Open a Device** from General Commands.
2. Enter the name of the device special file. Use any special file that exists for the device, for example, */dev/IBMtape0, /dev/IBMtape0n*, or */dev/IBMchanger0*.
3. Enter the *Read/Write*, *Read Only*, *Write Only*, or *Append* mode to open a device. These modes apply only to the tape devices. *Append* mode uses *Write Only* for file access permission. After you open a device, select a command using the appropriate number for the command from the window. Some commands require additional information after they are selected from the window. If an error occurs running the command, the error number, the error text, and the device sense data (if applicable) are displayed.

Command-Line Mode

The command-line mode for the tape utility program (*IBMtapeutil*) provides the same basic tape and changer commands as the interactive mode. Invoke the *IBMtapeutil* command from the Linux command line or within a shell script. If you enter the *IBMtapeutil* command without any arguments, the interactive mode is invoked. The syntax for the command-line mode of the tape utility program is:

```
IBMtapeutil -f Device Subcommand [Subcommand ...]
```

Note:

1. *Device* is the name of the device special file (for example, */dev/IBMtape0*).
2. *Subcommand* is any valid command for the device. Multiple subcommands can be combined in a single command to perform more than one operation. The subcommands are processed one at a time in the order specified on the command line. For help information about the subcommands and their syntax, enter the *IBMtapeutil ?* command on the Linux command line. Figure 9 on page 131 is an example of the help information that is displayed.

```
General Subcommands:

   tur              inquiry [Page]     print "Text"
   reserve          release            reqsense
   qryversion       logpage "Page"     modepage "Page"

Tape Subcommands:

   bsf [Count]      bsr [Count]        eof [Count]
   fsf [Count]      fsr [Count]        weof [Count]
   fsfm [Count]     bsfm [Count]       asf [Count]
   compress         tell               seek [Count]
   nocompress       rewind             sync
   load             erase              display "Message"
   unload           retension          read -d Destination [-c Count]
   qrypos           seod               write -s Source
   setpos [Blockid] status             rtest  [-b Blocksize] [-c Count] [-r Repetition]
   offline          parms              wtest  [-b Blocksize] [-c Count] [-r Repetition]
   rewoffl          list               rwtest [-b Blocksize] [-c Count] [-r Repetition]
   prevent          lock               setblk [Count]
   allow            unlock             density
   qryinquiry       qrysense           append   mtdevice   encryption

Medium Changer Subcommands:

   allow            prevent            audit [Address [Count]]
   inventory        mount [Slot]       position "Destination"
   elementinfo      unmount [Slot]     move "Source" "Destination"
   devids           exchange "Source" "Dest1" "Dest2"

Service Aid Subcommands:

   dump [Filename]  forcedump     ucode "Name"     resetdrive
```

Figure 9. Linux Command Line Help Display

General Subcommands

The following general subcommands are available for the tape and medium changer devices. Items shown enclosed in square brackets, for example [Page], are optional.

disablepath ″primary″ | number: The disablepath subcommand is used to disable a path to a medium changer. IBM recommends that you leave all paths enabled, but this subcommand can be used to disable a path. If ″primary″ is specified, the primary path is disabled. If a numeric value is specified, that specific path will be disabled. Use the path command to obtain a list of the paths. If there is only one path available and you attempt to disable it, this subcommand will fail with an illegal argument. If you attempt to disable a data path (to the tape drive), a ″function not implemented″ error will be returned. This subcommand is only available in IBMtapeutil version 1.2.0 or higher, and the lin_tape device driver, version 1.4.1 or higher, on libraries that have the Control Path Failover feature enabled.

enablepath ″primary″ | number: The enablepath subcommand is used to enable a path to a medium changer that has been previously disabled with the disablepath subcommand. If ″primary″ is specified, the primary path is enabled. If a numeric value is specified, that specific path will be enabled. Use the path command to obtain a list of the paths. If you attempt to enable a data path (to a tape drive), a ″function not implemented″ error will be returned. This subcommand is only available in IBMtapeutil version 1.2.0 or higher, and the lin_tape device driver, version 1.4.1 or higher, on libraries that have the Control Path Failover feature enabled.

inquiry [Page]: This subcommand issues the SCSI Inquiry command to the device for either standard inquiry data, when no *page* parameter is specified, or for a specific inquiry page. The *page* parameter must be specified as a hex value. For example:

```
# Get standard inquiry data
IBMtapeutil -f /dev/IBMtape0 inquiry

# Get inquiry page x'83'
IBMtapeutil -f /dev/IBMtape0 inquiry 83
```

logpage "Page": This subcommand issues the SCSI Log Sense Page command to the device for the specified *page* and displays the log sense data. The *page* parameter must be specified as a hex value. For example:

```
# Get log page x'31'
IBMtapeutil -f /dev/IBMtape0 logpage 31
```

modepage "Page": This subcommand issues the SCSI Mode Sense Page command to the device for the specified *page* and displays the mode sense data. The *page* parameter must be specified as a hex value. For example:

```
# Get mode page x'1D'
IBMtapeutil -f /dev/IBMtape0 modepage 1d
```

path: This subcommand shows all the available paths to a medium changer. This subcommand is only available in IBMtapeutil version 1.2.0 or higher, and the lin_tape device driver, version 1.4.1 or higher, on libraries that have the Control Path Failover feature enabled.

print "Text": This subcommand prints the associated text to standard output. It can be used at any time to display the progress of the subcommands. For example:

```
# Rewind, erase current tape and backup myfile.tar
IBMtapeutil -f /dev/IBMtape0 rewind \
erase \
print "Writing myfile.tar" \
write -s myfile.tar
```

qrypath: This subcommand shows the primary path and the first alternate path for a medium changer. This subcommand is only available in IBMtapeutil version 1.2.0 or higher, and the lin_tape device driver, version 1.4.1 or higher, on libraries that have the Control Path Failover feature enabled.

qryversion: This subcommand issues the QUERY_DRIVER_VERSION *ioctl* command. It prints out the current version of the lin_tape driver.

release: This subcommand explicitly releases a device and makes it available for other hosts by issuing the SCSI Release command.

reqsense: This subcommand issues the SCSI Request Sense command and prints out the sense data.

reserve: This subcommand explicitly reserves a device by issuing the SCSI Reserve command.

tur: This subcommand issues the SCSI Test Unit Ready command to the device.

Tape Subcommands

The following tape subcommands are available for the tape devices.

allow: This subcommand issues the SCSI Allow Medium Removal command to the device to allow medium removal by an operator. It is normally used after the *prevent* subcommand to restore the device to the default state.

append: This subcommand opens the device in *Append* mode. The file access permission is set to *Write Only*.

asf [Count]: This subcommand places the tape at the beginning block of *count* files from the beginning of the tape. Positioning is done by first rewinding the tape and then spacing forward over count filemarks. The default count is 1.

bsf [Count]: This subcommand backward spaces *count* filemarks. The tape is positioned on the beginning of the last block of the previous file. An optional count can be specified. The default is 1.

bsfm [Count]: This subcommand backward spaces *count* filemarks then positions the tape on the end of tape (EOT) side of the filemark. An optional count can be specified. The default is 1.

bsr [Count]: This subcommand backward spaces *count* records. An optional count can be specified. The default is 1.

compress and nocompress: These subcommands turn the hardware compression On or Off.

density: This subcommand issues the SCSI Report Density Support command for all supported media and for the current media loaded in the drive, and displays the results. If the drive is not loaded, the current media density is not reported.

display "Message": This subcommand displays a message on the display panel of the tape device. Up to 16 characters can be used for the message. If the message is longer than eight characters, the display alternates between the first eight characters and the remainder of the message. For example:

```
# Display "Test 1" and "Running" messages
IBMtapeutil -f /dev/IBMtape0 display "Test 1 Running"
```

encryption: This subcommand displays the encryption state of the drive.

eof [Count] and weof [Count]: These subcommands write *count* filemarks. An optional count can be specified. The default is 1.

erase: This subcommand erases the tape from the current position through the physical end of the tape.

fsf [Count]: This subcommand forward spaces *count* filemarks. The tape is positioned on the first block of the next file. An optional count can be specified. The default is 1.

fsfm [Count]: This subcommand forward spaces *count* filemarks, then positions the tape on the beginning of tape (BOT) side of the filemark. The tape is positioned on the last block of the previous file. An optional count can be specified. The default is 1.

fsr [Count]: This subcommand forward spaces *count* records. An optional count can be specified. The default is 1.

list: This subcommand rewinds the tape to the beginning of tape and displays the contents. The output lists filemarks and the size of each record found on the tape until the end of data is reached. If a record is larger than 256 KB, a read error is returned. When the command completes, the tape is at end of data. The output generated from this subcommand can be large, depending on the amount of data on the tape, and in most cases should be directed to a file. For example:

```
# List tape contents to file
IBMtapeutil -f /dev/IBMtape0 list >  tape.list
```

load: This subcommand issues the SCSI Load command to load a tape.

lock: This subcommand locks the tape drive door.

mtdevice: This subcommand issues the MTDEVICE *ioctl* command and displays the library device number. The device number consists of the control unit serial number with a 1 digit logical unit number appended to it. Both the control unit serial number and the 1 digit logical unit number are hexadecimal. The device number is only valid when an Enterprise Tape System 3590 is installed in a 3494 Enterprise Tape Library.

offline, rewoffl, and unload: These subcommands rewind and unload the tape.

parms: This subcommand issues the STIOCQRYP *ioctl* command and displays the current tape drive, media, and device driver parameters. Then the subcommand prompts users to set parameters to new values, and issues the STIOCSETP *ioctl* command to do so.

prevent: This subcommand issues the SCSI Prevent Medium Removal command to the device to prevent medium removal by an operator until the *allow* subcommand is issued or the device is reset.

qryinquiry: This subcommand issues the STIOCQRYINQUIRY *ioctl* command and displays the output.

qrypos: This subcommand issues the STIOCQRYPOS *ioctl* command for the current logical and physical tape position and displays the data. In addition, the current tape position is saved and can be restored using a subsequent *setpos* subcommand. For example:

```
# Append myfile.tar to the end of tape and then read back
IBMtapeutil -f /dev/IBMtape0 append \
qrypos \
write -s myfile.tar \
setpos \
read -d temp.tar
# Verify myfile.tar was written correctly
diff myfile.tar temp.tar
```

qrysense: This subcommand issues the STIOCQRYSENSE *ioctl* command to obtain new sense data.

read -d Destination [-c Count]: This subcommand reads a file or a specified number of records from the current position on the tape to the destination file name specified with the *-d* flag. If the optional *count* parameter is used, only the number of records specified with the *-c* flag are read unless a filemark is encountered before the number of specified records. If the *count* parameter is not used, all records up to the next filemark on tape are read. For example:

```
# Restore myfile.tar from tape
IBMtapeutil -f /dev/IBMtape0 read -d myfile.tar
# Read 3 records from the tape into myfile
IBMtapeutil -f /dev/IBMtape0 read -d myfile -c 3
```

rewind and retension: These subcommands rewind the tape to the beginning of the tape (BOT).

rtest [-b Blocksize] [-c Count] [-r Repetition]: This subcommand performs a read test by reading a random data pattern from the tape and verifying that it matches the written data. The *rtest* subcommand can be used after the *wtest* subcommand to verify the data. An optional *block size*, *count*, and *repetition* can be specified with the *-b*, *-c*, and *-r* flags, respectively. If the *block size* is fixed, the *count* specifies the number of blocks to read on each *repetition*. If the *block size* is zero (variable), the *count*specifies the number of bytes to read on each *repetition*. The default is a *block size* of 10240, a *count* of 20 blocks, and a *repetition* of 1. For example:

```
# R/W test using 256K blocks, 5 megabytes per write, 100 times
IBMtapeutil -f /dev/IBMtape0 rewind \
wtest -b 262144 -c 20 -r 100 \
rewind \
rtest -b 262144 -c 20 -r 100
```

Note: If the number of bytes read is not equal to the number of bytes specified, or there is a data mismatch, an error message is printed and *errno* is set to 999.

rwtest [-b Blocksize] [-c Count] [-r Repetition]: This subcommand performs a read and write test by writing a random data pattern on the tape, reading it, and verifying that it matches the written data. An optional *block size*, *count*, and *repetition* can be specified with the *-b*, *-c*, and *-r* flags, respectively. If the *block size* is fixed, the *count* specifies the number of blocks to write on each *repetition*. If the *block size* is zero (variable), the *count* specifies the number of bytes to write on each *repetition*. The default is a *block size* of 10240, a *count* of 20 blocks, and a *repetition* of 1. For example:

```
# R/W test using 256K blocks, 5 megabytes per write, 10 times
IBMtapeutil -f /dev/IBMtapeutil rwtest -b 262144 -c 20 -r 10
```

Note: If the number of bytes written/read is not equal to the number of bytes specified, or there is a data mismatch while reading, an error message is printed and *errno* is set to 999.

seek [Count]: If the optional *count* parameter is specified, the tape position is set to the block whose ID is *count*. Otherwise, if the *count* parameter is omitted, the tape position is set to the last position saved using the *tell* subcommand. The *count* can be specified in decimal or in hex (with a leading "x").

seod: This subcommand spaces to the end of data on the tape.

setblk [Block Size]: This subcommand sets the block size of the drive to *block size* bytes per record. A *block size* of zero sets the drive to variable block mode. If no *block size* is specified, the block is set to variable block mode.

setpos [Blockid]: This subcommand issues the SCSI Locate command to the device to set the tape position. If the optional *Blockid* parameter is specified, the tape position is set to the *Blockid*. Otherwise, if the *Blockid* parameter is omitted, the tape position is set to the last position saved using the *qrypos* subcommand. The *Blockid* can be specified in decimal or in hex (with a leading "x"). For example:

```
# Append myfile.tar to the end of tape and then read back
IBMtapeutil -f /dev/IBMtape0n append \
qrypos \
write -s myfile.tar \
setpos \
read -d temp.tar
# Verify myfile.tar was written correctly
diff myfile.tar temp.tar
# Set tape position to block 32 and leave positioned on close
IBMtapeutil -f /dev/IBMtape0n setpos 32
# Set tape position to block 32 and leave positioned on close
IBMtapeutil -f /dev/IBMtape0n setpos x20
```

status: This subcommand issues the MTIOCGET command and prints out status information about the tape unit.

sync: This subcommand synchronizes or flushes the tape buffers to tape.

tell: This subcommand reports the current block position on the tape and saves the position for use by a subsequent *seek* subcommand.

unlock: This subcommand unlocks the tape drive door.

write -s Source: This subcommand writes the source file specified with the *-s* flag on the tape. For example:

```
# backup myfile.tar to tape
IBMtapeutil -f /dev/IBMtape0 write -s myfile.tar
```

wtest [-b Blocksize] [-c Count] [-r Repetition]: This subcommand performs a write test by writing a random data pattern on the tape. The *rtest* subcommand can be used after the *wtest* subcommand to verify the data that was written. An optional *block size*, *count*, and *repetition* can be specified with the *-b*, *-c*, and *-r* flags, respectively. If the *block size* is fixed, the *count* specifies the number of blocks to write on each *repetition*. If the *block size* is zero (variable), the *count* specifies the number of bytes to write on each *repetition*. The default is a *block size* of 10240, a *count* of 20 blocks, and a *repetition* of 1. For example:

```
# R/W test using 256K blocks, 5 megabytes per write, 100 times
IBMtapeutil -f /dev/lin_tape rewind \
wtest -b 262144 -c 20 -r 100 \
rewind \
rtest -b 262144 -c 20 -r 100
```

Note: If the number of bytes written is not equal to the number of bytes specified, an error message is printed and *errno* set to 999.

Medium Changer Subcommands

The following medium changer subcommands are available for the integrated and independent medium changer devices.

allow

This subcommand is not supported for TotalStorage medium changer devices.

audit [Address[Count]]

This subcommand with no parameters issues the SCSI Initialize Element Status command to the device. The Address and Count parameters are not supported for IBM TotalStorage devices.

```
# Initialize all elements
IBMtapeutil -f /dev/IBMtape0 audit
```

devids

This subcommand issues the SCSI Read Element Status command to the device with the DVCID option for all drive elements and displays the element status information.

elementinfo

This subcommand displays the information returned from the SMCIOC_ELEMENT_INFO *ioctl* command that contains the number of and the address of the first of each element type.

exchange Source Dest1 Dest2

This subcommand is not supported for TotalStorage devices.

inventory

This subcommand issues the SCSI Read Element Status command for each element type and displays the element status information. No device identifier information is obtained.

mount [Slot]

This subcommand mounts a tape from the specified *slot* address into the drive or from the first full slot into the drive if the slot is omitted. For example:

```
# Mount cartridge from slot 3
IBMtapeutil -f /dev/IBMchanger0 mount 3
# Mount cartridge from first full slot
IBMtapeutil -f /dev/IBMchanger0 mount
```

move Source Destination

This subcommand issues the SCSI Move Medium command using the *source* and *destination* addresses specified. The element addresses can be obtained using the *elementinfo* subcommand. For example:

```
# Get slot and drive addresses
IBMtapeutil -f /dev/IBMchanger0 elementinfo
# Move cartridge in slot 20 to drive at address 16
IBMtapeutil -f /dev/IBMchanger0 move 20 16
```

position Destination

This subcommand issues the SCSI Position to Element command using the destination specified. For example:

```
# Position to slot at address 20
IBMtapeutil -f /dev/IBMchanger0 position 20
```

prevent

This subcommand is not supported for TotalStorage medium changer devices.

unmount [Slot]

This subcommand moves a tape from the drive to the specified *slot* address or the first empty one if *slot* is omitted. For example:

```
# Move tape from drive to slot address 4 (tape is already unloaded)
IBMtapeutil -f /dev/IBMchanger0 unmount 4
# Unload tape and move to the first empty slot
IBMtapeutil -f /dev/IBMchanger0 unmount
```

Service Aid Subcommands

The following service aid subcommands are available for the devices.

dump [Filename]

This subcommand is implemented using SCSI Pass Through *ioctl*. It stores the dump in the specified *Filename* or, if *Filename* is omitted, in *dump0001.dmp* in the current directory. It should follow a *forcedump* subcommand. For example:

```
# Force a dump and then stores the dump into file /tmp/dump1.dmp
IBMtapeutil -f /dev/IBMtape0 forcedump dump /tmp/dump1.dmp
```

Note: If the device is not supported, or the returned status from SCSI Pass Through is not correct, this operation fails and *errno* is set to 999.

forcedump

This subcommand is implemented using SCSI Pass Through *ioctl*. It forces a dump on a tape device. It is usually followed by the *dump* subcommand.

resetdrive

This subcommand issues a STIOC_RESET_DRIVE *ioctl* command to reset the device.

ucode [Filename]

This subcommand is implemented using SCSI Pass Through *ioctl*. It downloads microcode to the device. *Filename* is a file that contains the *ucode*. For example:

```
# download microcode
IBMtapeutil -f /dev/IBMchanger0 ucode /temp/device.ucode
```

Note: If the device is not supported, or the returned status from SCSI Pass Through is not correct, this operation fails and *errno* is set to 999.

Automatic Cartridge Facility (ACF) Mode

If the Enterprise Tape System 3590 tape subsystem has an ACF installed, the ACF mode can be queried by the *IBMtapeutil* application using the **Query/Set Parameters** option in the window.

This parameter is currently not supported for IBM Ultrium devices.

Block Size and SCSI Transfer Size

The minimum and maximum block sizes for the tape device and the maximum SCSI transfer size can be queried using either the interactive mode of the tape utility program and selecting **Query/Set Parameters** under Tape Commands, or by issuing the *parms* subcommand when in command-line mode.

Configuration Parameters

The configuration parameters can be queried using either the interactive mode of the tape utility program and selecting **Query/Set Parameters** under Tape Commands, or by issuing the *parms* subcommand when in command-line mode. Some of the parameters can be changed using the interactive mode of the tape utility program and selecting **Query/Set Parameters** under lin_tape Commands.

Note: All configuration parameters are reset to their default values whenever the lin_tape device driver is reinstalled.

Reserve and Release Commands

The device driver reserves the device automatically on the *open* call and releases the device on the *close* call. This prevents other applications and hosts from accessing the device. However, there may be situations when the *reserve* should be maintained after the *close* call. For example, some backup programs (such as *tar*) can open and close the device multiple times, or a device may be shared by multiple initiators or hosts. In these cases, the reservation must be retained between the *close* call and the next *open* call. This is done by explicitly reserving the device. After all work is completed on the device, the user will explicitly release the device.

A device can be reserved and released explicitly using either the interactive mode of the tape utility program and selecting *reserve* or *release* under General Commands, or by issuing *reserve* and *release* commands when in command-line mode. For example:

```
# Reserve device, run tar, and then release device
IBMtapeutil - f /dev/IBMtape0 reserve
tar ... /dev/IBMtape0 ...
IBMtapeutil -f /dev/IBMtape0 release
```

After the *reserve* command is used, the device driver retains the reservation until a *release* command is issued.

Tape Drive Service Aids

The service aids described here are accessible through both the interactive and command-line mode of the `IBMtapeutil` command, for more information refer to "Service Aid Subcommands" on page 137. For information on using tape drive service aids with the IBM Virtualization Engine TS7510, please refer to the *IBM System Storage Virtualization Engine TS7510 Installation and Planning Guide* at:

http://www-03.ibm.com/systems/storage/index/tape.html.

Create Special Files

The interactive mode of the tape utility program provides a selection to create special files based on attached devices. It invokes a command, *lin_tapeconfig*, to remove current */dev/lin_tape* and */dev/IBMchanger* files and generate new ones. The command *lin_tapeconfig* can also be used directly from the Linux command line, assuming this utility program has been installed.

Note: Running *lin_tapeconfig* requires *root* authority.

Problem Determination

A set of tools is provided with the device driver to determine if the device driver and the tape device are functioning correctly.

Configure and Run lin_taped Daemon

Starting with lin_tape version 1.2.5, the lin_tape device driver provides an error diagnostic daemon (lin_taped) which provides the following capabilities:

1. Full error logging and tracing of the lin_tape device driver
2. When drive dumps, log sense data, and/or SIM/MIM error information are created by the tape drive, the daemon will automatically retrieve that data and save it to the hard drive on your Linux system.

Because lin_taped requires a minimal amount of system resource and because it provides these necessary diagnostic capabilities, IBM recommends that you leave the daemon enabled at all times.

Install lin_taped

lin_taped is automatically installed at */usr/bin/lin_taped* when you install the lin_tape device driver using the *rpm* or *tar* package. Refer to "Installation and Configuration Instructions" on page 110 for instructions on installing the lin_tape device driver.

Configure lin_taped

You can customize the operation of lin_taped by modifying its configuration file, which is located at */etc/lin_taped.conf*. The daemon only reads the configuration file when it starts; consequently, if you make modifications to the configuration file, stop the daemon and restart it so that your updates will be recognized by the daemon.

Tracing: Three levels of tracing are supported for the lin_tape device driver and are defined as follows:

0 With tracing set to 0, very minimal tracing is recorded from the lin_tape device driver.

1 With tracing set to 1, lin_taped records information associated with each ioctl called. If a device error occurs and SCSI sense data is obtained from the device, a subset of that sense data will also be recorded. This is the default setting for tracing.

2 With tracing set to 2, lin_taped will record tracing messages for each SCSI command. If a device error occurs and SCSI sense data is obtained from the device, all sense data will also be recorded. This tracing level should only be used when a specific problem is being diagnosed due to the potential for huge amounts of data being generated.

Set the *lin_tapeTrace* variable in the */etc/lin_taped.conf* file to 0, 1, or 2, depending on what level of tracing you desire. If the *lin_tapeTrace* variable is set to an invalid number, the lin_taped daemon will not start.

Tracing information is written to a file named */var/log/lin_tape.trace*, by default. Information is written into the file until it is 1 MB in size, by default. After 1 MB of information is written, the file is archived (using the Linux *ar* command) into file *lin_tape.a* in the same directory. In the archive, the filename will be renamed to *lin_tape.trace.timestamp*, where *timestamp* reflects the time that the file was archived.

You may change the directory to which the tracing information is written or the default maximum size of the trace file by modifying settings in the *lin_taped.conf* file. Refer to the instructions in the *lin_taped.conf* file for details.

Error Logging: lin_taped records certain error messages from the lin_tape device driver in a file named */var/log/lin_tape.errorlog*, by default. Information is written into the file until it is 1 MB in size, by default. After 1 MB of trace information is written, the file is archived (using the Linux *ar* command) into file *lin_tape.a* in the same directory. In the archive, the filename will be renamed to *lin_tape.errorlog.timestamp*, where *timestamp* reflects the time that the file was archived.

You may change the directory to which the error logging information is written or the default maximum size of the error log file by modifying settings in the *lin_taped.conf* file. Refer to the instructions in the *lin_taped.conf* file for details.

Whenever the lin_taped daemon is running, error logging is enabled if tracing is enabled. Following is an example an error log record:

```
IBMtape0---E0001  Tue Sep 10 14:04:57 2002
  Scsi Path   :  03 00 00 00
  CDB Command :  01 00 00 00 00 00
  Status Code :  08 00 00 01
  Sense Data  :  70 00 04 00 00 00 00 58 00 00 00 00 00 00 FF 0B
                 C4 77 00 00 00 06 01 40 00 00 00 00 00 00 01 00
                 10 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  Description :  Hardware Error
```

The first line indicates the tape device special file name and the device serial number, and the timestamp when the error message was recorded. ″Scsi Path″ is the SCSI path for this logical unit. It matches the order of the scsi/Channel/Id/Lun information in the */proc/scsi/scsi file*. ″CDB Command″ is the command data block of the SCSI command. ″Status Code″ is the returned result from the Linux SCSI middle layer device driver (*scsi_mod.o*). The four bytes represent driver_byte, host_byte, msg_byte, and status_byte, respectively. ″Sense Data″ is the full SCSI sense data returned from the target. ″Description″ is a person-readable text string obtained by parsing the sense key field of the sense data.

| The following circumstances are not logged in the lin_tape.errorlog file:

1. Sense key is 0, and the sense data indicates an overlength or an underlength read, or encountering a file mark or the end of data
2. Sense key is 2, and the ASC/ASCQ indicates the device is becoming ready
3. Sense key is 6, indicating a unit attention
4. Sense key is 8, and the ASC/ASCQ indicates the end of data

| **Volume Logging:** The lin_tape device driver retrieves the full log sense data from the tape drive whenever the drive reaches a log threshold, or a tape is unloaded from the drive, or the drive is reset through an application. This data is stored in binary in a file named *lin_tape.timestamp.log*, where lin_tapex is the device special file (for example, lin_tape1, lin_tape2, etc.) and *timestamp* reflects the time the file was created. Each time log sense data is obtained, it is written to a new file. Use the appropriate tape drive hardware reference manual to decode the log sense data.

The volume logging data is stored in the */var/log* directory by default. You may specify another directory in the */etc/lin_taped.conf* file.

| There are two configuration parameters in the */etc/lin_taped.conf* file that you can tailor to affect the number of log sense files that are kept on your system: *lin_tapeMaxLogSenseFiles* and *lin_tapeAutoLogSenseFileOverWrite*. *lin_tapeMaxLogSenseFiles* can be 0 or a positive decimal number. *lin_tapeAutoLogSenseFileOverWrite* can be 0 or 1. If *lin_tapeMaxLogSenseFiles* is 0, *lin_tapeAutoLogSenseFileOverWrite* is ignored, and each time log sense data is obtained, it is written to a new file. If *lin_tapeMaxLogSenseFiles* is a positive number and *lin_tapeAutoLogSenseFileOverWrite* is 0, each time log sense data is created, lin_taped will write that data to a file until *lin_tapeMaxLogSenseFiles* have been created; then lin_taped will stop creating new files, even if new log sense data is produced. If *lin_tapeMaxLogSenseFiles* is a positive number and *lin_tapeAutoLogSenseFileOverWrite* is 1, each time log sense data is created, lin_taped will write that data to a file until *lin_tapeMaxLogSenseFiles* have been created; then when new log sense data is detected, lin_taped will delete the oldest log sense file and create a new file with the newest log sense data; thus, only the newest data will be kept.

By default, *lin_tapeMaxLogSenseFiles* is 0 and *lin_tapeAutoLogSenseFileOverWrite* is 1, which means that every time log sense data is created, it will be written to a new file.

Automatically Retrieve a Drive Dump: If a condition occurs in the drive such that a drive dump is created, lin_taped will retrieve the drive dump and save it in a file named *lin_tapex.timestamp.dmp*, where lin_tape**x** is the device special file (for example, lin_tape1, lin_tape2, etc.) and *timestamp* reflects the time the file was created. Each time a drive dump is obtained, it is written to a new file. IBM service may request that you forward drive dumps to them for analysis.

The drive dumps are stored in the */var/log* directory by default. You may specify another directory in the */etc/lin_taped.conf* file.

There are two configuration parameters in the */etc/lin_taped.conf* file that you can tailor to affect the number of drive dumps that are kept on your system: *lin_tapeMaxDumpFiles* and *lin_tapeAutoDriveDumpFileOverWrite*. *lin_tapeMaxDumpFiles* can be 0 or a positive decimal number. *lin_tapeAutoDriveDumpFileOverWrite* can be 0 or 1. If *lin_tapeMaxDumpFiles* is 0, *lin_tapeAutoDriveDumpFileOverWrite* is ignored, and each time a drive dump is obtained, it is written to a new file. If *lin_tapeMaxDumpFiles* is a positive number and *lin_tapeAutoDriveDumpFileOverWrite* is 0, each time a dump is obtained, lin_taped will write that data to a file until *lin_tapeMaxDumpFiles* have been created; then lin_taped will stop creating new files, even if new drive dumps are produced. If *lin_tapeMaxDumpFiles* is a positive number and *lin_tapeAutoDriveDumpFileOverWrite* is 1, each time a dump is obtained, lin_taped will write that data to a file until *lin_tapeMaxDumpFiles* have been created; then when a new drive dump is detected, lin_taped will delete the oldest drive dump file and create a new file with the newest drive dump data; thus, only the newest data will be kept.

By default, *lin_tapeMaxDumpFiles* is 0 and *lin_tapeAutoDriveDumpFileOverWrite* is 1, which means that every time a drive dump is obtained, it will be written to a new file.

Automatically Retrieved SIM/MIM Data: If a condition occurs in the drive such that a drive SIM/MIM data is created, lin_taped will retrieve the data and save it in a file named *lin_tapex.timestamp.simmim*, where lin_tape**x** is the device special file (for example, lin_tape1, lin_tape2, etc.) and *timestamp* reflects the time the file was created. Each time SIM/MIM data is obtained, it is written to a new file. IBM service may request that you forward SIM/MIM data to them for analysis.

The SIM/MIM data is stored in the */var/log* directory by default. You may specify another directory in the */etc/lin_taped.conf* file.

There are two configuration parameters in the */etc/lin_taped.conf* file that you can tailor to affect the number of SIM/MIM files that are kept on your system: *lin_tapeMaxSimMimDataFiles* and *lin_tapeAutoSimMimDataOverWrite*. *lin_tapeMaxSimMimDataFiles* can be 0 or a positive decimal number. *lin_tapeAutoSimMimDataOverWrite* can be 0 or 1. If *lin_tapeMaxSimMimDataFiles* is 0, *lin_tapeAutoSimMimDataOverWrite* is ignored, and each time SIM/MIM data is obtained, it is written to a new file. If *lin_tapeMaxSimMimDataFiles* is a positive number and *lin_tapeAutoSimMimDataOverWrite* is 0, each time SIM/MIM data is obtained, lin_taped will write that data to a file until *lin_tapeMaxSimMimDataFiles* have been created; then lin_taped will stop creating new files, even if new SIM/MIM data are created. If *lin_tapeMaxSimMimDataFiles* is a positive number and *lin_tapeAutoSimMimDataOverWrite* is 1, each time SIM/MIM data is obtained, lin_taped will write that data to a file until *lin_tapeMaxSimMimDataFiles* have been created; then when new SIM/MIM data is detected, lin_taped will delete the oldest SIM/MIM file and create a new file with the newest SIM/MIM data; thus, only the newest data will be kept.

By default, *lin_tapeMaxSimMimDataFiles* is 0 and *lin_tapeAutoSimMimDataOverWrite* is 1, which means that every time SIM/MIM data is obtained, it will be written to a new file.

Selective Tracing: lin_tape provides facilities by which you can disable and enable tracing, error logging, auto-retrieving drive dumps, and auto-retrieving SIM/MIM data. You may selectively enable/disable them through the *IBMtapeutil* Query/Set Tape Parameters operation or through an application program which uses the STIOC_SETP *ioctl*. These settings persist until the device driver is restarted, or the host system is rebooted. The parameters and their definitions are as follows:

trace
: This parameter is set to On by default, which enables lin_tape tracing of activities and error logging on a particular tape drive. Set this parameter to off to stop tracing and error logging.

logging
: This parameter is set to On by default and enables logging of log sense data. Setting this flag to Off suppresses volume logging for this device.

disable_sim_logging
: This parameter controls the logging of SIM/MIM data for a device. By default it is set to Off which causes SIM/MIM data to be logged. Set this flag to On to suppress the logging of SIM/MIM records.

disable_auto_drive_dump
: This parameter controls the saving of drive dumps for a device. By default it is set to Off which causes drive dumps to be saved. Set this flag to On to suppress the saving of drive dumps.

Run lin_taped

If you are running the lin_tape device driver, version 1.4.1 or higher, after installing lin_tape lin_taped will start running even if your system does not have a tape device attached. If you add a new tape device into your Linux system, lin_taped will automatically create a special file under the /dev directory. If you are running the lin_tape device driver, version 1.3.x or less, lin_taped will not automatically start if there is no tape device attached. After you attach a new tape device, you'll need to start the lin_taped daemon.

You can invoke lin_taped from the command line. lin_taped takes zero or more of the parameters as listed below:

```
lin_taped [start stop restart status]
```

lin_taped or lin_taped start
: Starts the daemon. If there is already a lin_taped running, the new one will be aborted. (Use "lin_taped restart" if lin_taped is already running.)

lin_taped stop
: Terminates the daemon and frees all the resources associated with the daemon. When the daemon is stopped, no information is saved.

lin_taped restart
: Terminates the currently running daemon and starts a new one. The new

daemon will read the */etc/lin_taped.conf* file. This command should be used after modifying the */etc/lin_taped.conf* file while lin_taped is running.

lin_taped status
Prints a message on stdout indicate whether the daemon is running or not.

Note: If you run ″rmmod lin_tape″ command to remove the lin_tape device driver from the running kernel, you need to stop the lin_taped daemon first; otherwise you will get a ″Device or Resource Busy″ error.

Tape Drive Service Aids

The service aids described here are accessible through both the interactive and command-line mode of the *IBMtapeutil*. Refer to “Service Aid Subcommands” on page 137. For information on using tape drive service aids with the IBM Virtualization Engine TS7510, please refer to the IBM System Storage Virtualization Engine TS7510 Installation and Planning Guide at: http://www-03.ibm.com/systems/storage/index/tape.html

Details of Tape Drive Service Aids

The following service aid utilities are available through the utility program *IBMtapeutil* installed with the lin_tape device driver.

- Force Microcode Dump
- Read Dump
- Load Microcode
- Reset Drive

Force Drive Dump: This utility forces a dump operation on the tape drive. After the dump operation is performed, the dump data can be saved to a file using the Read Dump utility. To access this utility, invoke *IBMtapeutil*, then choose ″Force Dump″ under ″Service Aids Commands″ in the menu or issue the following command:

```
IBMtapeutil -f /dev/IBMtape0 forcedump
```

Read Dump: This utility transfers the dump data from the device to a file, a diskette, or a tape cartridge. It follows a *force drive dump* operation. To access this utility, invoke *IBMtapeutil*, then choose ″Dump Device″ under ″Service Aids Commands″ in the menu or issue the following command:

```
IBMtapeutil -f /dev/IBMtape0 dump [filename]
```

If the filename is not specified, *dump0001.dmp* is used as the default. All the dump files are stored under the current directory.

Load Microcode: This utility downloads microcode to the tape drive or medium changer from a file. To access this utility, invoke *IBMtapeutil*, then choose ″Load Ucode″ under ″Service Aids Commands″ in the menu or issue the following command:

```
IBMtapeutil -f /dev/IBMtape0 ucode filename
```

Note: The filename has to be specified. Otherwise, *IBMtapeutil* returns an error.

Reset Drive: This utility resets the tape drive. To access this utility, invoke *IBMtapeutil*, then choose ″Reset Drive″ under ″Service Aids Commands″ in the menu or issue the following command:

```
IBMtapeutil -f /dev/IBMtape0 resetdrive
```

Chapter 6. Solaris Tape and Medium Changer Device Driver

This chapter provides an overview of the IBM SCSI Tape and Medium Changer Device Driver for Solaris, also known as *IBMtape*.

Purpose

This device driver product provides attachment for IBM Magnetic Tape and Library System products to Sun Microsystems SPARC Servers running the Solaris operating system.

It is designed specifically to take advantage of the features provided by IBM tape and library systems, including control of the random access medium changer facility (move, element information, and inventory) present in some devices. The goal is to give applications access to the functions required for basic operations (such as backup and restore), as well as the advanced functions needed by full tape management systems. Whenever possible, the device driver is designed to take advantage of the IBM tape system features transparent to the application.

Data Flow

Both data and commands flow between the application program and the tape subsystem by way of IBMtape. Figure 10 on page 146 shows the data flow between IBMtape, the application program, the SCSI adapter device driver, and the IBM tape system.

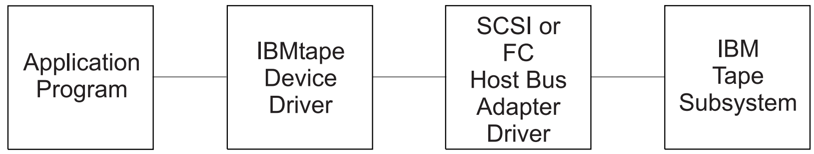


Figure 10. Data Flow for Solaris Device Driver (IBMtape)

Product Requirements

The following hardware and software components are required and supported by IBMtape.

Hardware Requirements

The following hardware is required and supported by IBMtape:

- One of the following SCSI-2 Differential Host Bus Adapters:
  - Sun Microsystems S-Bus Differential SCSI-2 Host Adapter
  - Sun S-Bus Ultra Differential SCSI-2 Fast/Wide Host Adapter
  - Sun Dual-Channel Differential Ultra SCSI Host Adapter PCI (requires FC 9799 for 3590 or FC 5099 for IBM Magstar MP 3570 or 3575)
  - Sun Microsystems Differential Fast/Wide SCSI-2 Host Adapter (DWIS/S) Host Adapter (UDWIS/S) (P/N X1062A)
- One or more of the following SCSI High Voltage Differential (HVD) host bus adapters (359x only):
  - Sun Microsystems Differential Fast/Wide SCSI-2 Host Adapter (DWIS/S) Host Adapter (UDWIS/S) (P/N X1062A)
  - Sun Microsystems SBus Ultra Differential Fast/Wide Intelligent SCSI-2 Host Adapter (UDWIS/S) (P/N X1065A)
  - Sun Microsystems Dual-channel Differential UltraSCSI Host Adapter, PCI (P/N X6541A)
- One or more of the following SCSI Low Voltage Differential (LVD) host bus adapters on Solaris Version 8, or later (Ultrium only):
  - Sun Microsystems PCI Adapter with two fast Ethernet Interfaces and two SCSI Interfaces (P/N X2222A)
  - Sun Microsystems Dual Fast Ethernet + Dual SCSI PCI Adapter (Dual VHDCI)
  - Sun Microsystems PCI Dual Channel Gigabit Ethernet UTP & Dual Channel SE Ultra-2 SCSI Adapter (P/N X4422A)
  - Sun Microsystems PCI-X Single and Dual Ultra320 SCSI Host Adapter SCSI Adapter (P/N SG-XPCI1SCSILM320-Z and SG-XPCI2SCSILM320-Z)
- One or more of the following Serial Attached SCSI (SAS) host bus adapters on Solaris Version 10, or later (Ultrium only):
  - Sun StorageTek PCI-X SAS HBA (SG-XPCI8SAS-E-Z)
  - Sun StorageTek PCI-Express SAS HBA (SG-XPCIE8SAS-E-Z)
- The IBM 2108–G07 (IBM SAN Data Gateway) and IBM 2108–R03 (IBM SAN Data Gateway Router) can be used to attach SCSI models of the IBM 357x, 3590, and VTS of products with any of the supported FC–AL host bus adapters.
- The Ultrium tape drives and Enterprise Tape drives 3590 (Model E) and 3592 with Fibre Channel Attachment are supported with the following adapters:

  - QLogic QLA2200F, QLA2310FL, QLA2340, QLA2340L, QLA2342, QLA2342L, QLA2460, QLA2462, PCI Fibre Channel Adapters
  - Emulex LightPulse LP8000(PCI), LP8000S(S-Bus), LP9002C(cPCI), LP9002L(PCI), LP9002S(S-Bus), LP9042DC(PCI-X), LP9802(PCI-X), LP10000(PCI-X), LP11000, and LP11002 Fibre Channel Adapters
  - AMCC FCX/2-6562(PCI-X), FCX-6562(PCI-X), FCC/2–6562(cPCI), FCC-6562(cPCI), FCE-6460(PCI) and FCE-1473(S-Bus) Fibre Channel Adapters
  - Sun StorEdge 2 Gb FC PCI Single Channel network adapter (X6767A)
  - Sun StorEdge 2 Gb FC PCI Dual Channel network adapter (X6768A)
  - Sun StorageTek Enterprise Class 4 Gb FC PCI-X HBA (SG-XPCI2FC-QF4 and SG-XPCI1FC-QF4)
- Table 20 indicates which SCSI-attached devices are supported through the IBM 2108-G07 (IBM SAN Data Gateway) and IBM 2108-R03 (IBM SAN Data Gateway Router) with which fibre-channel host bus adapters.

Table 20. SCSI Supported Devices

DEVICE	QLogic QLA2100	QLogic QLA2200F	Emulex LP8000	Emulex LP8000S
Magstar MP 3570	X			
Magstar MP 3575	X			
Enterprise 3590, models B11, B1A, E11, and E1A	X	X	X	X
VTS model B18	X	X	X	X
VTS models B10, B20		X	X	X

IBM Enterprise Tape Drive 3590, Models B11, B1A, E11, and E1A

IBM Magstar MP 3570 Tape Subsystem Model C

IBM Magstar MP 3575 Tape Library Dataserver

IBM TotalStorage Virtual Tape Server (Models B10, B18, and B20)

Notes:

1. Using a single Fibre Channel host bus adapter (HBA) for concurrent tape and disk operations is not recommended. Tape and disk devices require incompatible HBA settings for reliable operation and optimal performance characteristics. Under stress conditions (high I/O rates for tape, disk, or both) where disk and tape subsystems share a common HBA, stability problems have been observed. These issues are resolved by separating disk and tape I/O streams onto separate HBAs and using SAN zoning to minimize contention. IBM is focused on assuring server and storage configuration interoperability. It strongly recommends that your implementation plan includes provisions for separating disk and tape workloads.
2. For IBM Ultrium drives with the Fibre Channel attachment, the Sun Solaris operating system requires that the Fibre Channel addressing mode of the drive be set to hard addressing.
3. The most recent supported HBA information can be found at:

   http://www-03.ibm.com/systems/support/storage/config/hba/index.wss .

Software Requirements

The following software is required and supported by IBMtape:

- Sun Microsystems Solaris operating system Version 2.6, 7, 8, 9, or 10 on SPARC servers. For Solaris Version 2.6, Sun patch 105867-01 is required.
- SCSI Host Bus Adapter (HBA) driver as supplied by the HBA manufacturer.

  Refer to the manufacturer's documentation for the SCSI adapter driver to determine which adapter driver is required.

Installation and Configuration Instructions

IBM SCSI Tape Drive and Medium Changer Device Driver for Solaris is an installable kernel module, supplied as a standard Solaris software package. When installed, its package name is *IBMtape*. The following sections describe installation, removal, configuration, and verification procedures for IBMtape. Refer to the Solaris documentation for general information about installable packages.

The IBMtape package consists of the device driver and a number of associated files and utilities. For components created during IBMtape installation, refer to Table 21.

Table 21. IBMtape Components

Component	Description
/etc/ibmekm.conf	SME configuration file, working version
/etc/tmd.conf	TMD configuration file, working version
/opt/IBMtape	Package subdirectory
/opt/IBMtape/tapeutil	Utility and service aid program
/opt/IBMtape/tapeutil.c	Utility/service program sample source code
/opt/IBMtape/diags_info	Diagnostic script
/opt/IBMtape/ibmekm.conf	SME configuration file, reference version
/opt/IBMtape/tapelist	Utility program
/opt/IBMtape/tmd	Tape Monitor Daemon (TMD) program
/opt/IBMtape/tmd.conf	TMD configuration file, reference version
/opt/IBMtape/IBMtape.conf	Configuration file, reference version
/opt/IBMtape/ztapelist	Shell script for Solaris zones use
/opt/IBMtape/tapedtrc	Dynamic tracing utility program
/usr/kernel/drv/IBMtape	Kernel module device driver
/usr/kernel/drv/IBMtape.conf	Configuration file, working version
/usr/include/sys/smc.h	Medium changer application programming interface (API) header file
/usr/include/sys/st.h	Tape drive API header file
/usr/include/sys/svc.h	Service aid API header file
/usr/include/sys/oldtape.h	Compatibility API header file
Note: When updating IBMtape, the working copies of IBMtape.conf, tmd.conf and ibmekm.conf are not overwritten by the package file contents. This allows tape drive configuration options to be preserved across IBMtape updates. The reference copies of IBMtape.conf, tmd.conf and ibmekm.conf are always installed in the /opt/IBMtape directory.	

Examples of installation commands and their results throughout this chapter use a percent sign (%) to indicate the shell prompt.

Preventing Conflicts with Other Device Drivers

IBMtape only attempts to claim and operate the devices described in "Hardware Requirements" on page 146. However, the Solaris operating system includes a SCSI tape device driver named *st*, which claims any SCSI-compliant tape drive that it detects, including devices that IBMtape manages. In order to avoid conflicts between IBMtape and *st*, you must prevent *st* from claiming and attempting to operate IBMtape-owned devices. Likewise, other suppliers' SCSI tape device drivers that you have installed must be prevented from claiming IBMtape-owned devices.

Attention: Failure to prevent more than one device driver from operating the same tape drive may cause system panics or data loss on the tape drive.

The following installation and update steps describe how to prevent conflicts between IBMtape and other SCSI tape device drivers.

Preinstallation Considerations

Follow these steps to install or update IBMtape. Before starting the step-by-step procedure, note the following general considerations:

- Differential SCSI-2 support must already exist on the machine before installing IBMtape. Install and configure one of the supported differential SCSI adapters first, then return to this section. Refer to the differential SCSI adapter documentation for instructions on installing the adapter and adapter driver.
- You must have root authority to install or remove IBMtape.
- You may restart the system as part of the IBMtape installation. Take appropriate precautions that this does not adversely affect users or active processes on the system.
- As a consequence of installing or reinstalling IBMtape, device special file numbers under */dev/rmt* may change. These numbers are assigned by Solaris during the driver attachment process, and the sequencing cannot be specified by the device driver or installer.

Installing and Updating IBMtape

There are several steps that must be taken prior to installing or updating IBMtape on your system to ensure proper installation and system integrity. These are:

1. Notify users that system maintenance and a restart is performed.
2. Select a time when all system activity can be stopped to perform the installation.
3. Login to the target system as *root*.
4. Ensure that all user and tape drive activity on the system is halted.
5. If tape drives not owned by IBMtape are installed on the system, list the low-density device special files and find the SCSI addresses with which they are currently associated.

   ```
   % ls -l /dev/rmt/*l
   lrwxrwxrwx   1 root     root          72 Aug 26 15:47 /dev/rmt/5l ->
   ../../devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,10000/st@2,0:l
   ```

   In the preceding example, */dev/rmt/5l* and the related *5m*, *5h*, and so on, are controlled by the *st* device driver and are associated with the device at SCSI address 2, LUN 0. Record the device type, */dev/rmt* special file number, owning driver, SCSI target address and LUN. You will need this information later in the installation.

For example, suppose an installation has two non-IBM devices owned by *st* at SCSI addresses 2 and 8. The low-density devices are accessed as special files */dev/rmt/5l* and */dev/rmt/6l*. For the equipment listing after recording the device information, refer to Table 22.

Table 22. IBMtape Install or Update

DEVICE	Old Special File	Old Driver	SCSI Address/LUN (Old)
QIC	*/dev/rmt/5l*	st	2/0
QIC	*/dev/rmt/6l*	st	8/0

6. If this is an update to IBMtape, there are already IBMtape-owned devices installed. In that case, list the primary device special files and find the SCSI addresses with which they are currently associated.

```
% ls -l /dev/rmt/*st /dev/rmt/*smc
lrwxrwxrwx   1 root     other        46 Aug 26 16:36 /dev/rmt/0st ->
../../devices/pci@6,4000/scsi@3/IBMtape@b,0:st

lrwxrwxrwx   1 root     other        47 Aug 26 16:36 /dev/rmt/1smc ->
../../devices/pci@6,4000/scsi@3/IBMtape@b,1:smc
```

In this example, */dev/rmt/0st* (a SCSI tape drive) is controlled by IBMtape and is associated with the device at SCSI address b, LUN 0. The address is reported in hexadecimal format: The file */dev/rmt/1smc* (a SCSI medium changer) is associated with the device at SCSI address b, LUN 1. Record the device type, */dev/rmt* special file number, owning driver, SCSI target address and LUN. You will need this information later in the installation.

For example, suppose an installation has only an IBMtape-owned device installed at SCSI address 8. The device consists of both a tape drive and SCSI medium changer. The tape drive is accessed as */dev/rmt/2st*, and the medium changer as */dev/rmt/3smc*. For a similar equipment listing after recording the device information, refer to Table 23.

Table 23. Solaris Device Driver - IBMtape - Equipment Listing Example 1

DEVICE	Old Special File	Old Driver	SCSI Address/LUN (Old)
3590-B11 drive	*/dev/rmt/2st*	IBMtape	8/0
3590-B11 changer	*/dev/rmt/3smc*	IBMtape	8/1

7. Select one of the following methods to prevent conflicts between IBMtape and other SCSI tape device drivers, depending on the equipment that is attached to your system.

   Attention: Failure to prevent more than one device driver from operating the same SCSI tape drive may cause system panics or data loss on the tape drive.

   a. If the system will have only IBMtape-owned devices attached, follow these steps to prevent *st* and other non-IBM SCSI tape device drivers from claiming the IBM devices.

      1) Edit */kernel/drv/st.conf*, and comment out all SCSI target entries by placing a pound sign (#) in the first column of each target entry. The following example shows the entries for SCSI target addresses 0 and 1 commented out. Repeat this operation for all target entries.

```
#name="st" class="scsi"
#target=0 lun=0;

#name="st" class="scsi"
#target=1 lun=0;
```

2) For other non-IBM installed SCSI tape device drivers, remove the drivers if they are not needed. If a driver is for SCSI tape devices only, it should not be needed. If a driver is for both tape and disk devices, follow the suppliers' instructions to disable its access to all SCSI tape devices.

b. If the system will have a mixture of IBMtape-owned devices and other tape drives, follow these steps to configure *st* and other non-IBM SCSI tape device drivers so that they control a range of target addresses distinct from the range that IBMtape will use. These steps leave target addresses 7 and 15 unclaimed by all target device drivers, because SCSI adapters typically use one of those two addresses.

1) Edit */kernel/drv/st.conf*, and comment out SCSI target entries for addresses 7–15 by placing a pound sign (#) in the first column of each target entry. In the following example, the entries for SCSI address 7 and 8 have been commented out. Repeat this operation for all entries in the target address range 7–15.

```
#name="st" class="scsi"
#target=7 lun=0;

#name="st" class="scsi"
#target=8 lun=0;
```

2) For other non-IBM installed SCSI tape device drivers, follow the suppliers' instructions to disable their access to all SCSI tape devices in the address range 7–15.

3) Later, after the IBMtape package has been installed, you must alter its configuration file so it does not use SCSI target addresses in the range 0–7 or address 15.

Now *st* and other non-IBM SCSI tape device drivers have been configured to avoid conflicting with IBMtape.

8. Remove all special file entries under */dev/rmt*. This ensures that stale entries do not exist after the system is restarted. New entries are created when the system is restarted.

```
% rm /dev/rmt/*
```

9. If you are updating the level of IBMtape, remove the currently installed IBMtape package. If this is a new installation of IBMtape, skip this step.

a. Use **pkgrm** to remove the current level.

```
% /usr/sbin/pkgrm IBMtape
```

Respond to the pkgrm prompts.

b. Examine the results from **pkgrm**. If you see these messages, one or more IBMtape-owned tape drives or tape monitor daemon (TMD) were still in use. Identify the drives and TMD process ID (pid), and end the processes that are using them. If you cannot identify the processes, you must restart the system to free the tape drive, then continue with the installation from this point.

```
...
Device busy
Cannot unload module: IBMtape
Will be unloaded upon reboot.
...
```

10. Select one of the following methods to install the IBMtape package, depending on the package distribution medium and the location of system resources.

    Note: If this is a new install of IBMtape, IBM devices are not yet attached to the system, and you will see *pkgadd* error messages similar to the following:

    ```
    ...
    drvconfig: Driver (IBMtape) failed to attach
    Warning: Driver (IBMtape) successfully added to system
    but failed to attach
    ## The device driver was unable to detect any supported devices!
    ## Verify that the device(s) are properly connected and powered on.
    ## Ensure that the SCSI adapter device driver is installed/configured.
    ## Then try reinstalling the device driver as follows:
    ##   -enter the command: rem_drv IBMtape
    ##   -enter the command: add_drv -m '* 0666 bin bin' IBMtape
    ## If problems persist, contact your IBM service representative.
    pkgadd: ERROR: postinstall script did not complete successfully
    ...
    ```

    Later, after you have cabled IBM drives to the system and restarted, the driver will attach normally.

    If the distribution medium is a package file in a Unix file system, perform the following steps. You may have obtained a package file by extracting it from a distribution diskette or by downloading it from IBM's anonymous FTP site. This example presumes a package file named *IBMtape.4.0.2.7* located in the */tmp* directory.

    a. If necessary, FTP the package file to the target system. Use binary transfer mode. Place the package file in the target system's */tmp* directory.
    b. Use **pkgadd** to install the driver:

       ```
       % /usr/sbin/pkgadd -d /tmp/IBMtape.4.0.2.7
       ```

11. If your system environment includes a mixture of IBMtape-owned devices and devices owned by *st* or another third-party SCSI tape device driver, you should have already modified the configuration files for the non-IBM device drivers and restricted them to target addresses in the range 0–6.

    Edit *IBMtape.conf*, located in */usr/kernel/drv*, and comment out entries for SCSI target addresses 0–7 and 15 by placing a pound sign (#) in the first column of each line making up the entries. In the following example, the entries for address 0, LUN 0 and address 0, LUN 1 have been commented out. Repeat the operation for all stanzas in the address range 0–7 and address 15. Note that each SCSI target address has a stanza for both LUN 0 and LUN 1.

    ```
    #name="IBMtape" class="scsi"
    #target=0 lun=0
    #block_size=0
    #buffering=1
    #immediate=0
    #trailer=0
    #sili=0;

    #name="IBMtape"  class="scsi"
    #target=0 lun=1
    #block_size=0
    #buffering=1
    #immediate=0
    #trailer=0
    #sili=0;
    ```

    In SAN environment, the fibre channel HBA driver may map the SCSI target address out of the range 15 and LUN over the number 1. You create a new entry with the mapped SCSI target in IBMtape.conf. In the following example, a IBM tape device is mapped to the SCSI target 32 and LUN 15.

```
name="IBMtape" class="scsi"
target=32 lun=15
block_size=0
buffering=1
immediate=0
trailer=0
sili=0;
```

Alternatively, you can modify the configuration file or use the utility provided by the fibre channel HBA driver to persistently bind the tape device to the expected SCSI target address.

12. Shut down the system. One common method to perform a shutdown is shown here but use your installation's normal procedures.

```
% /usr/sbin/shutdown -y -g0 -i0
```

13. Address or readdress devices as determined by your installation:
    a. If the system will have only IBMtape-owned devices attached, you may select addresses in the range 0–6 or 8–14. Leave addresses 7 and 15 unused, because these addresses are used typically by the SCSI adapter.
        1) For each device, refer to the appropriate IBM hardware reference for any special instructions about addressing. Then set the address and record the device type, SCSI address, and LUN. For example, suppose an installation will have only IBMtape-owned devices attached. An IBM device with tape drive and medium changer is added. It is addressed at target 4 and the information is recorded. For the results, refer to Table 24

Table 24. Solaris Device Driver - IBMtape - Equipment Listing Example 2

DEVICE	Old Special File	Old Driver	SCSI Address/LUN (Old)	SCSI Address/LUN (New)
3590-B11 drive	–	–	–	4/0
3590-B11 changer	–	–	–	4/1

    b. If you are using distinct address ranges to separate tape drives that are IBMtape-owned from devices that are owned by *st* or another supplier's driver, readdress the tape drives now.
        1) For each device to be owned by *st* or another SCSI tape device driver, refer to the manufacturer's hardware reference for any special instructions about readdressing. Then readdress each device to an address in the range 0–6. For each tape drive that is readdressed, record the new SCSI address along side the special file number and old SCSI address that you recorded previously.
        2) Readdress all tape drives that are owned by IBMtape to addresses in the range 8–14. Refer to the appropriate IBM hardware references for any special instructions about readdressing. For each tape drive that is readdressed, record the new SCSI address along side the special file number and old SCSI address, if any, that you recorded previously.

           For example, suppose an installation has two non-IBM devices owned by *st* at SCSI addresses 9 and B (12 in decimal). An IBM device with tape drive and medium changer is added. To prevent conflicts between IBMtape and *st*, the non-IBM devices are all placed at addresses in the range 0–6. The new IBM device is addressed in the range 8–14, at address 10, or X'0A'. Depending on the addresses chosen for the non-IBM devices, after readdressing and recording device information, refer to Table 24 on page 153 for the possible equipment listing.

Table 25. Solaris Device Driver - IBMtape - Equipment Listing Example 3

DEVICE	Old Special File	Old Driver	SCSI Address/LUN (Old)	SCSI Address/LUN (New)
QIC	*/dev/rmt/2l*	st	9/0	3/0
QIC	*/dev/rmt/3l*	st	b/0	5/0
3590-B11 drive	–	–	–	a/0
3590-B11 changer	–	–	–	a/1
Note: The SCSI target address of fibre channel tape device may be over 15.				

14. Cable the tape drives to the system, if not yet done. Refer to the manufacturer's hardware references for any special instructions about cabling. Ensure that each SCSI bus is terminated properly.
15. Start the system according to your installation's normal procedures.
16. Log on as *root* and list the device special files in */dev/rmt* as you did earlier during the install.

```
% ls -l /dev/rmt/*l
% ls -l /dev/rmt/*st /dev/rmt/*smc
```

Compare the SCSI addresses obtained from *ls* with the readdressed SCSI targets you recorded. Write the new device special file numbers and owning driver next to the matching new SCSI addresses.

For example, suppose an installation previously had two non-IBM devices owned by *st* at SCSI addresses 2 and–8. An IBM device with tape drive and medium changer is added. To prevent conflicts between IBMtape and *st*, the non-IBM devices are all placed at addresses in the range 0–6. The new IBM device is addressed in the range 8–14. Depending on the addresses chosen, after completing installation and recording device information, refer to Table 26 for the possible equipment listing entries.

Table 26. Solaris Device Driver - IBMtape - Equipment Listing Example 4

DEVICE	Old Special File	Old Driver	SCSI Address/ LUN (Old)	SCSI Address/ LUN (New)	New Driver	New Special File
QIC	*/dev/rmt/5l*	st	2/0	2/0	st	*/dev/rmt/0l*
QIC	*/dev/rmt/6l*	st	8/0	0/0	st	*/dev/rmt/1l*
3590-B11 drive	–	–	–	8/0	IBMtape	*/dev/rmt/2st*
3590-B11 changer	–	–	–	8/1	IBMtape	*/dev/rmt /3smc*
Note: Based on the listing, you can see that the tape drive accessed previously as */dev/rmt/5* will now be accessed as */dev/rmt/0*, the new medium changer is accessible as */dev/rmt/3smc*, and so on.						

17. Verify operation of the newly installed or readdressed equipment.
18. Notify users of any changed device special files numbers.

Configuring IBM tape devices with Fibre Channel and SAS HBAs

Users may experience difficulty attaching an IBM tape device with an IBM tape driver on a fibre network. The following sections describe how to configure IBM tape devices with QLogic, Emulex, Sun and AMCC Fibre Channel HBAs:

- "Configuring IBM tape devices with QLogic FC HBAs"
- "Configuring IBM tape devices with Emulex FC HBAs"
- "Configuring IBM tape devices with Sun FC and SAS HBAs" on page 156
- "Configuring IBM tape devices with AMCC FC HBAs" on page 157

Configuring IBM tape devices with QLogic FC HBAs

To configure an IBM tape device with a QLogic FC HBA, complete the following steps:

1. Run the QLogic SANSurfer Control FX utility to find and record the mapped target and LUN of the tape device.
2. Remove the comment at the beginning of the entry for the QLogic HBA in the */usr/kernel/drv/IBMtape.conf* file.

   For example, the following command opens the entry for QLogic QLA2462 running QLogic HBA driver qla2300:

   ```
   name="IBMtape" parent="qla2300" target=0;  # for qla2300 only
   ```
3. Update the entry for the device in the *IBMtape.conf* file, if necessary. The current entry in the *IBMtape.conf* file is added to target 255 with LUN 0 and 1.

   For instance, the following command adds an entry for a mapped device with target 200 and LUN 3:

   ```
   name="IBMtape" class="scsi" target=200 LUN=3;
   ```
4. Unload and reload the IBMtape driver:

   ```
   # /opt/IBMtape/tmd -s
   # rem_drv IBMtape
   # add_drv -m '* 0666 bin bin' IBMtape
   # /opt/IBMtape/tmd
   ```
5. Display information on the configured devices by running **/opt/IBMtape/tapelist -l**.

Configuring IBM tape devices with Emulex FC HBAs

To configure an IBM tape device with an Emulex FC HBA, complete the following steps:

1. Run the Emulex HBAnyware utility to find and record the mapped target and LUN of the tape device.
2. Remove the comment at the beginning of the entry for the Emulex HBA in */usr/kernel/drv/IBMtape.conf*:

   ```
   name="IBMtape" parent="lpfc" target=0;
   ```

   If this fails to configure the changer, you may need to add the entries for LUN 0 and 1:

   ```
   name="IBMtape" parent="lpfc" target=X lun=0;
   name="IBMtape" parent="lpfc" target=X lun=1;
   ```
3. Update the entry for the device in the *IBMtape.conf* file, if necessary. The current entry in *IBMtape.conf* adds target 255 with LUN 0 and 1.

   For instance, the following command adds an entry for a mapped device with target 200 and LUN 3:

   ```
   name="IBMtape" class="scsi" target=200 lun=3;
   ```
4. Unload and reload the IBMtape driver:

```
# /opt/IBMtape/tmd -s
# rem_drv IBMtape
# add_drv -m '* 0666 bin bin' IBMtape
# /opt/IBMtape/tmd
```

5. Display information on the configured devices by running **/opt/IBMtape/tapelist -l**.

Configuring IBM tape devices with Sun FC and SAS HBAs

To configure an IBM tape device with a Sun FC HBA, complete the following steps:

1. Attach the IBM tape devices on the host.
2. Install the appropriate patch for the Sun HBA driver.
3. Run **# cfgadm -al** to display the configuration between the HBA and the tape device.

   Run **# cfgadm -al -o show_FCP_dev Ap_Id** to show the medium changer configuration with the HBA.

   Run **# cfgadm -c configure device** to configure the tape device with the HBA if needed.
4. Install the IBMtape tape driver by running the **# pkgadd -d IBMtape** command.
5. Enter the following appropriate lines in */etc/driver_aliases*:

   ```
   IBMtape "scsiclass,01.vIBM.pXXX"  for tape drive.
   IBMtape "scsiclass,08.vIBM.pXXX" for medium changer.
   ```

   where *01* and *08* stand for the type of tape drive and medium changer and *XXX* is the product ID string in the standard inquiry data. For example, ULT3580-TD2 is the product ID of the IBM LTO-2 drive. The following entry will be added in the file for the IBM LTO-2 drive:

   ```
   IBMtape "scsiclass,01.vIBM.pULT3580-TD2"
   ```

   The following is a list of the entries for the supported IBM tape devices.

   For the tape drives:

   ```
   IBMtape "scsiclass,01.vIBM.pULT3580-TD1"
   IBMtape "scsiclass,01.vIBM.pULTRIUM-TD1"
   IBMtape "scsiclass,01.vIBM.pULT3580-TD2"
   IBMtape "scsiclass,01.vIBM.pULTRIUM-TD2"
   IBMtape "scsiclass,01.vIBM.pULT3580-TD3"
   IBMtape "scsiclass,01.vIBM.pULTRIUM-TD3"
   IBMtape "scsiclass,01.vIBM.pULT3580-HH3"
   IBMtape "scsiclass,01.vIBM.pULTRIUM-HH3"
   IBMtape "scsiclass,01.vIBM.pULT3580-TD4"
   IBMtape "scsiclass,01.vIBM.pULTRIUM-TD4"
   IBMtape "scsiclass,01.vIBM.p03592J1A"
   IBMtape "scsiclass,01.vIBM.p03592E05"
   IBMtape "scsiclass,01.vIBM.p03592E06"
   ```

   For the medium changers:

   ```
   IBMtape "scsiclass,08.vIBM.p03584L32"
   IBMtape "scsiclass,08.vIBM.p03584L22"
   IBMtape "scsiclass,08.vIBM.pULT3582-TL"
   IBMtape "scsiclass,08.vIBM.pULT3583-TL"
   IBMtape "scsiclass,08.vIBM.pULT3581-TA"
   IBMtape "scsiclass,08.vIBM.pULT3581-TA2"
   IBMtape "scsiclass,08.vIBM.p3576-MTL"
   IBMtape "scsiclass,08.vIBM.p3573-TL"
   IBMtape "scsiclass,08.vIBM.p3577-TL"
   ```

   Note: The entry is also added running the **# update_drv** command on Solaris 8 (patch 111804-03 is required), Solaris 9, and later versions of the operating system. For example:

```
# update_drv -av -i '"scsiclass,01.vIBM.pULT3580-TD2"' IBMtape
```

6. Reboot the system by running **# reboot -- -r**.
7. Run **# /opt/IBMtape/tapelist -l** to display the configured tape device information.

   Note: All of the added entries will be removed by the operating system automatically after IBMtape is unloaded from the kernel by running the **# pkgrm**, **rem_drv**, or **modunload** commands. It is strongly recommended that you back up these entries in a file, and that you re-enter the entries when you upgrade the IBMtape driver before running the **#pkgadd** command.

Configuring IBM tape devices with AMCC FC HBAs

To configure an IBM tape device with an AMMC FC HBA, complete the following steps:

1. Modify and add the following parameters in `/kernel/drv/jnic146x.conf`:
   ```
   CmdTaskAttr=1;
   lun_throttle=1;
   tape-device="IBMtape";
   tape-changer="IBMtape";
   ```
2. Update the change in `jnic146x.conf`.
3. Run the EZ Fibre utility to find and record the mapped target and LUN of the tape device.
4. Remove the comment from the beginning entry for AMCC HBA in */usr/kernel/drv/IBMtape.conf*:
   ```
   name="IBMtape" parent="jnic146x" target=0;
   ```
5. Update the entry for the device in *IBMtape.conf* if necessary. The current entry in *IBMtape.conf* adds target 255 with LUN 0 and 1. For instance, use the following command to add an entry for a mapped device with a target 200 and LUN 3:
   ```
   name="IBMtape" class="scsi" target=200 lun=3;
   ```
6. Unload and reload the IBMtape driver:
   ```
   # /opt/IBMtape/tmd -s
   # rem_drv IBMtape
   # add_drv -m '* 0666 bin bin' IBMtape
   # /opt/IBMtape/tmd
   ```
7. Display information on the configured devices by running **/opt/IBMtape/tapelist -l**.

Solaris Zones Support

The Solaris Zones partitioning technology on Solaris 10 is used to virtualize operating system services and provide an isolated and secure environment for running applications. Every Solaris system contains a global zone with ID 0, where the IBMtape driver is installed. Zones hosted by a global zone are known as *non-global zones*, which have their own node name, virtual network interface, and storage assignment.

The IBMtape driver supports the Solaris Zones environment. To install IBMtape on the system with the virtualized zones, run the **pkgadd** system command from the global zone:

```
#pkgadd -G -d IBMtape.x.x.x.x
```

The IBMtape install script installs the driver in the global zone and installs some of IBMtape utilities running a non-root user in all zones.

Since the tape devices in non-global zones are configured from the global zone, a script program called **ztapelist** was developed to help the user display the IBM tape devices and to dynamically assign or remove IBM tape devices in non-global zones without the non-global zone reboot requirement. The utility (available in IBMtape 4.1.5.2 or later) is installed on Solaris 10 and runs in the global zone only.

Use the **ztapelist** utility on the command line as follows:

Synopsis

```
/opt/IBMtape/ztapelist [-l] [-c] [-z zonename] [-a] [-d] [-h]
```

Options and Usage

ztapelist recognizes the following options:

-l
: Displays IBM tape device information with the column headers for all zones

-c
: Displays IBM tape device information without the column headers for all zones

-z zonename
: Shows IBM tape devices in a zone

-a
: Dynamically adds IBM tape devices to each non-global zone without additional arguments

zonename inst#_1 inst#_2 inst#_3 ...
: Dynamically sets IBM tape devices not greater than 7 in a non-global zone

zonename all
: Dynamically sets all IBM tape devices on the system in a non-global zone

-d
: Removes the IBM tape device from a non-global zone

-h
: Displays help information

The **ztapelist** command displays all of the IBM tape devices in the global zone and the tape devices in the non-global zones as shown in the following example. An option of **ztapelist -z zonename** is also provided to show all of the assigned tape devices in a particular zone.

```
# /opt/IBMtape/ztapelist -l
Running in global zone ...
Inst#    Special File   Device        Serial No         TGT/LUN     Ucode     World Wide NN     World Wide PN
------- ------------- ---------- ----------------- ------------ ---------- -----------------  -----------------
193   /dev/rmt/27st   03592E05(e/e)    000001365066       2/0       1A38   500507630019F016   500507630059F016
194   /dev/rmt/28smc  03584L22         0000000T003904E5   2/1       805r            N/A               N/A
200   /dev/rmt/29st   ULT3580-TD4(e)   1300000044         4/0       82F0   500507630019F009   500507630059F009
201   /dev/rmt/30smc  03584L32         0000000T00390401   4/1       805r                  N/A              N/A
206   /dev/rmt/31st   ULT3580-TD3      1210003557         7/0       73P5    500507630019F007   500507630059F007
38    /dev/rmt/32smc  03584L32         0000000T00390401   7/1       805r                  N/A              N/A


Running in non-global zone camshaft ...
Inst#    Special File  Device           Serial No         TGT/LUN    Ucode     World Wide NN   World Wide PN
------- ------------- ---------------- ------------------ -------- ---------- -------------    -------------
200    /dev/rmt/29st   ULT3580-TD4(e)  1300000044          4/0       82F0   500507630019F009  500507630059F009
201    /dev/rmt/30smc  03584L32        0000000T00390401    4/1       805r         N/A               N/A

Running in non-global zone softail ...
Inst#    Special File    Device         Serial No       TGT/LUN   Ucode     World Wide NN       World Wide PN
------- ------------ -------------    ------------     ---------- -------- ---------------  ------------------
193   /dev/rmt/27st      03592E05(e/e) 000001365066        2/0      1A38   500507630019F016    500507630059F016
194   /dev/rmt/28smc     03584L22      0000000T003904E5    2/1      805r          N/A                N/A
```

To add the tape devices in non-global zones, run **# /opt/IBMtape/ztapelist -a**, as shown in the following example:

```
# ztapelist -a
Issuing this function will assign the tape devices in non-global zone
Do you wish to continue? [y/n]: y
Inst#   Special File       Device        Serial No          TGT/LUN      Ucode    World Wide NN      World Wide PN
------- ---------------- ------------   --------------    ------------ ---------- ----------------   --------------
193    /dev/rmt/27st      03592E05(e/e)   000001365066        2/0       1A38    500507630019F016  500507630059F016
194    /dev/rmt/28smc     03584L22        0000000T003904E5    2/1       805r          N/A                  N/A
200    /dev/rmt/29st      ULT3580-TD4(e)  1300000044          4/0       82F0    500507630019F009  500507630059F009
201    /dev/rmt/30smc     03584L32        0000000T00390401    4/1       805r          N/A                  N/A
206    /dev/rmt/31st      ULT3580-TD3     1210003557          7/0       73P5    500507630019F007  500507630059F007
38     /dev/rmt/32smc     03584L32        0000000T00390401    7/1       805r          N/A                  N/A

Enter Instance Number (Inst #) of a device to be added: 200

  ID NAME             STATUS         PATH
   0 global           running         /
   1 camshaft         running        /zones/zone1
   2 softail          running        /zones/zone2
Enter the zonename  where the device will be added: camshaft
Inst#    Special File      Device           Serial No       TGT/LUN  Ucode    World Wide NN   World Wide PN
------- ---------------  ---------------- ----------------- --------- ------- ---------------  ---------------
200    /dev/rmt/29st     ULT3580-TD4(e)      1300000044       4/0     82F0    500507630019F009 500507630059F009

Do you wish to continue to add the devices? [y/n]: y
Enter Instance Number (Inst #) of a device to be added: 193
Enter the zonename  where the device will be added: softail
Inst#    Special File      Device           Serial No       TGT/LUN  Ucode    World Wide NN   World Wide PN
------- ---------------  ---------------- ----------------- --------- ------- ---------------  ---------------
193    /dev/rmt/27st     03592E05(e/e)      000001365066     2/0      1A38    500507630019F016 500507630059F016

Do you wish to continue to add the devices? [y/n]: n
#
```

The **ztapelist** command also allows the user to remove all or some assigned tape devices from the non-global zone, as shown in the following example:

```
# /opt/IBMtape/ztapelist -d
Issuing this function will remove the tape devices from non-global zone
Do you wish to continue? [y/n]: y
Do you want to remove the tape devices from all of non-global zones? [y/n]: n
  ID NAME            STATUS        PATH
   0 global          running         /
   1 camshaft        running         /zones/zone1
   2 softail         running         /zones/zone2

Enter the zonename where the devices will be removed: camshaft
Do you want to remove all of the tape devices from this zone? [y/n]: n
Inst#    Special File      Device          Serial No       TGT/LUN    Ucode     World Wide NN      World Wide PN
------- ---------------- -------------  ----------------- ----------- -------- ------------------- ------------------
200     /dev/rmt/29st    ULT3580-TD4(e)   1300000044        4/0       82F0   500507630019F009     500507630059F009
201    /dev/rmt/30smc    03584L32         0000000T00390401   4/1       805r         N/A               N/A

Enter Instance Number (Inst #) of a device to be removed: 201
Removing this tape device /dev/rmt/30smc for this zone camshaft ...
Inst#   Special File       Device          Serial No       TGT/LUN   Ucode    World Wide NN       World Wide PN
------- ------------- ------------------ ---------------- ---------- -------- ------------------- ------------------
200    /dev/rmt/29st     ULT3580-TD4(e)     1300000044       4/0      82F0    500507630019F009   500507630059F009

Do you wish to continue to remove the devices from this zone? [y/n]: n
Do you wish to continue to remove the devices from other zone? [y/n]: y
  ID NAME            STATUS        PATH
   0 global          running        /
   1 camshaft        running        /zones/zone1
   2 softail         running        /zones/zone2

Enter the zonename where the devices will be removed: softail
Do you want to remove all of the tape devices from this zone? [y/n]: y
Removing all of tape devices for this zone softail ...
Inst#     Special File         Device       Serial No         TGT/LUN  Ucode   World Wide NN    World Wide PN
------- ------------------ -------------- ----------------- --------- -------- -------------  -------------------
Do you wish to continue to remove the devices from other zone? [y/n]: n
```

Configuration Parameters

When using devices controlled by IBMtape, certain device characteristics, such as the default *block size*, can be controlled through the device driver configuration file. The IBMtape configuration file is named *IBMtape.conf*. The working copy of this file is located in the */usr/kernel/drv* directory.

During installation of IBMtape, the working copy of *IBMtape.conf* is preserved, if it exists. During removal of IBMtape, the working copy of *IBMtape.conf* is not deleted. These conventions allow configuration settings to remain across updates of IBMtape. A reference copy of *IBMtape.conf* with factory default settings is always installed in the */opt/IBMtape* directory.

Note: IBM requires that the Solaris native SCSI tape device driver *st* be configured so that it does not attempt to support SCSI targets that are controlled by IBMtape. Refer to “Preventing Conflicts with Other Device Drivers” on page 149 for more information about multiple driver access to a device.

Attention: Failure to prevent more than one device driver from operating the same tape drive may cause system panics or data loss on the tape drive.

Configuration settings are applied only at start time, or when IBMtape is unloaded manually from, then reloaded into, memory. If you change configuration settings in *IBMtape.conf*, you can make the changes effective by restarting the system. As an alternative to restarting, ensure that no IBMtape-owned devices are in use, then issue the following:

```
% /opt/IBMtape/tmd -s   for IBMtape.4.0.9.2 and later
% /usr/sbin/rem_drv IBMtape
% /usr/sbin/add_drv -m '* 0666 bin bin' IBMtape
% /opt/IBMtape/tmd      for IBMtape.4.0.9.2 and later
```

Default settings in *IBMtape.conf* can be overridden for a particular device (and only while the device is kept open) using the *ioctl* application programming interface (API) of the device driver. The parameter settings made through the API revert back to the default values in *IBMtape.conf* the next time the device is opened. Refer to the *IBM TotalStorage and System Storage Tape Device Drivers: Programming Reference* for more information about changing configuration parameters under program control.

IBMtape.conf contains one stanza for each SCSI target address/LUN pair owned by IBMtape. The reference *IBMtape.conf* file supplied with the package contains a stanza for every possible SCSI target and LUN combination supported by IBM tape systems.

The following example shows the stanza for target 0, LUN 0, with IBMtape's default configuration parameter values. The parameter immediate is disabled, which means that SCSI commands Write FM, Locate, Load-Unload, Erase, and Rewind complete before returning status:

```
name="IBMtape" class="scsi"
target=0 lun=0
block_size=0
buffering=1
immediate=0
trailer=0
sili=0;
```

The following example shows the stanza for target 0, LUN 0, with IBMtape's default configuration parameter values and the rewind immediate mode set on, which causes the SCSI rewind command to return control to the application program before the command actually completes on the tape drive:

```
name="IBMtape" class="scsi"
target=0 lun=0
block_size=0
buffering=1
rew_immediate=1
trailer=0
sili=0;
```

If immediate is set to 1 and rew_immediate is set to 0, the setting of rew_immediate is ignored.

The **name** variable identifies IBMtape as the device driver, and **class** identifies the type of device supported as SCSI.

The **target** and the **lun** variables determine the target address and LUN of IBM devices that are controlled by that stanza. On systems with multiple SCSI adapters, a single target/LUN stanza controls the configuration settings for all devices addressed with that target address and LUN. Thus, two or more supported IBM devices on the system that have the same target and LUN settings but are attached to different SCSI buses are all affected by the configuration parameters of the single stanza having that target address and LUN.

After installation of the IBMtape package is complete, you may eliminate unnecessary probing for devices by commenting out unused target/LUN pairs. In this example, the stanzas for target 0, LUN 0 and target 0, LUN 1 have been commented out. Those address/LUN combinations will not be probed, which saves time during a restart or manual reload of IBMtape. However, if an IBM device is addressed at target 0, LUN 0 or target 0, LUN 1, it will not be detected.

```
#name="IBMtape" class="scsi"
#target=0 lun=0
#block_size=0
#buffering=1
#immediate=0
#trailer=0
#sili=0;

#name="IBMtape"  class="scsi"
#target=0 lun=1
#block_size=0
#buffering=1
#immediate=0
#trailer=0
#sili=0;
```

The remaining five configuration parameters specifically affect the behavior of the IBM device or devices associated with that stanza (target and LUN). All of these parameters are specific to tape drive device operation only and have no effect on medium changer device behavior. The default configuration parameters are adequate for most purposes. However, the values in the configuration file can be modified to suit the specific requirements of the application or the user.

Remember that modifying a value in the configuration file determines the value of the parameter at device open time. When the device is open, the value of a parameter can be altered using an *ioctl* function call, but the change is effective only while the device remains open. Working configuration parameters revert back to the default values (established by the configuration file) when the device is closed and reopened. Refer to the *IBM Tape Device Drivers: Programming Reference* for more information about changing configuration parameters through program control.

Table 27 lists and describes the set of configuration parameters recognized by the IBMtape device driver.

Table 27. Configuration Parameters Recognized by IBMtape

Parameter	Values	Description
block_size	(0=variable length)	This option specifies the device block size that is established with the SCSI Mode Select command during an open function call. Until this value is changed, it is the working block size. Variable block size is established using a value of zero. Any other positive value represents a fixed block size. The maximum supported block size varies for each tape device. Refer to the appropriate hardware reference manual for additional information. **Note:** IBMtape does not allow odd-byte-count fixed block reads or writes. For instance, a fixed block size of 4096 or 4098 is allowed, but 4097 is not. If you attempt to read or write using an odd-byte-count fixed block size, the read or write will return -1, with errno set to 22, invalid argument. If you must read or write odd-byte-count blocks, set block size to 0 (variable block size), then transfer one block's worth of data per read or write.

Table 27. Configuration Parameters Recognized by IBMtape (continued)

Parameter	Values	Description
buffering	(0=Off, 1=On)	When a write command is processed, the data is either directly stored on the physical tape or buffered in device hardware. Buffering can be turned On and Off with this option. If buffering is disabled, the effective performance of the device may be degraded seriously, because the tape devices cannot take advantage of their buffering optimization. Buffer flushing (or committing data to the tape) can be controlled by the application through the STIOC_SYNC_BUFFER *ioctl* function
immediate	(0=Off, 1=On)	If immediate is set to 0, the SCSI commands Write FM, Locate, Load-Unload, Erase, and Rewind return with status when the command actually completes on the tape drive. If immediate is set to 1, these commands return with status before the command actually completes.
rew_immediate	(0=Off, 1=On)	If rew_immediate is set to 0, the SCSI Rewind command returns with status when the command actually completes on the tape drive. If it is set to set to 1, the Rewind command returns with status before the command actually completes. If immediate is set to 1, the setting of rew_immediate is ignored.
trailer	(0=Off, 1=On)	If a tape drive encounters logical end-of-tape (EOT) during a write operation, it returns a check condition status. The driver returns 0 bytes written to notify the application of this EOT situation. A check condition will also be returned by the tape drive for every subsequent write operation when past EOT. If trailer is enabled, writing records past EOT is allowed by the device driver. Following the first time the write operation notifies the application of EOT, all subsequent EOT notifications are suppressed by the driver, and the actual number of bytes written is returned. When physical end of media is reached, all write operations fail with a return code of -1, regardless of the trailer setting. When trailer is enabled, managing the media past EOT is the application's responsibility.
sili	(0=Off, 1=On)	Normally, during a read operation, if a larger block of data is requested than is actually read from the tape, the tape device raises a check condition. The IBMtape device driver must perform error handling procedures, which add overhead to the read operation. The IBMtape driver does not surface this as an error condition to the application and ultimately returns the actual number of bytes read. However, this driver error processing results in less than optimum read performance in some scenarios. When SILI mode is enabled, the tape device is forced to Suppress Illegal Length Indication during read operations. This eliminates the error processing performed by the driver and results in improved read performance for some scenarios. The actual number of bytes read is still returned to the application in SILI mode.

Removing IBMtape

All active processes using IBM devices supported by the IBM SCSI Tape and Medium Changer Device Driver for Solaris must be stopped in order for the removal procedure to complete successfully.

Use the **pkgrm** command to remove the IBMtape package from the system.

```
% /usr/sbin/pkgrm IBMtape
```

Adding or Removing Devices

To add support for a new IBM tape system or to remove support for a previously attached IBM tape system, perform the following steps:

1. Edit the *IBMtape.conf* file in the */usr/kernel/drv* directory to reflect the change in IBM device support. Either add a new stanza to provide support for a device that is to be added, or remove (comment out) a stanza for a device that is no longer supported.
2. When adding support for a new device, ensure that the target and LUN values in the configuration file stanza match the target and LUN settings of the IBM device. Refer to "Configuration Parameters" on page 160 for more information about the *IBMtape.conf* configuration file.
3. Shut down and power Off the host system.
4. Plug the new device into the SCSI bus or unplug the existing device from the bus. Pay particular attention to proper SCSI cabling and termination.
5. Power On and start the host system.

   Notes:

   a. It is possible to reinitialize the IBMtape device driver without restarting the system. This is done by first unloading the device driver, then reloading the device driver into kernel memory.
   b. For the version of IBMtape.4.0.9.2 and later, the TMD daemon has to be stopped by running the */opt/IBMtape/tmd -s* command to unload the IBMtape driver from the kernel. Running the */opt/IBMtape/tmd* command restarts the daemon afterwards to reload the device driver.

   The commands to unload the device driver are:

```
 % /opt/IBMtape/tmd -s        for IBMtape.4.0.9.2 and later
% /usr/sbin/rem_drv IBMtape
```

   The commands to reload the device driver are:

```
% /usr/sbin/add_drv -m '* 0666 bin bin' IBMtape
% /opt/IBMtape/tmd            for IBMtape.4.0.9.2 and later
```

   When the IBMtape device driver is reloaded, it will read the *IBMtape.conf* file and acknowledge changes made in the file. This method can be used to modify configuration parameters.

   Note: It is strongly suggested that the host system and all devices attached to the SCSI bus be powered Off before adding or removing devices from the SCSI bus. Hot plugging SCSI devices can cause hardware damage and disruption of reliable system operation.

Tapelist Utility Program

A Tapelist Utility Program called *tapelist* is installed in the `/opt/IBMtape` directory as part of the IBMtape package. The tapelist utility provides the user a listing of tape, medium changer and SAN data gateway devices configured with the IBMtape driver. It also displays the information of HBA with IBM tape drive attachment and the current status of load balancing. The following is an example of a Tapelist Utility Program output.

```
# tapelist
Instance : 697
Special File : /dev/rmt/6st
Device : 03592E05(e/e)
Serial Number : 000001300168
TGT/LUN : 7/0
Ucode : 04C4
World Wide NN : 5005076302000127
World Wide PN : 5005076302400127
Dev Phy Path : /devices/pci@1f,2000/QLGC,qla@1/IBMtape@7,0:st
Path Type : N/A

# tapelist -t
hba_index  hba_inst  hba_driver   reg_count   usage_count       HBA Path
---------  --------  ----------- ----------- ----------- --------------------------
 0         0         qla2300     4           1          /devices/pci@4,2000/fibre-channel@1
 1         2         lpfc        4           2          /devices/pci@6,2000/pci@1/fibre-channel@4
 2         3         lpfc        1           0          /devices/pci@6,2000/pci@1/fibre-channel@5
```

A new feature is added in tapelist to display the drive information in the library that is running `/opt/IBMtape/tapelist -L`.

```
# tapelist -L
Addr  Inst#  Special File   Device      Serial No        TGT/LUN Ucode  World Wide NN    World Wide PN
----- ----- ------------- ------------- ---------------- ------- ----- ---------------  ---------------
Library (/dev/rmt/5smc) Info:
      2894  /dev/rmt/5smc 03584L22       0000000T003904E5  3/1    806c   N/A              N/A
274   2893  /dev/rmt/13st 03592E05(e/e)  000001365066      3/0    1D10   500507630019F016 500507630059F016
276   2914  /dev/rmt/14st 03592E06(e/e)  000001326803      14/0   2444   500507630019F019 500507630059F019

Library (/dev/rmt/7smc) Info:
      2899  /dev/rmt/7smc 03584L32       0000000T00390401  6/1    806c   N/A              N/A
265   2898  /dev/rmt/6st  ULT3580-TD4(e) 1300000044        6/0    82F0   500507630019F009 500507630059F009

Library 3494 Info:
      322   /dev/rmt/10st 03592J1A       000001300147      1/0    0464   5005076300000000 5005076300400000
```

The following is a definition of the fields and headers that appear in the screens above:

```
Inst #         The instance number of the particular device.
Special File   The device special file used to access this device.
Device         A string indicating the device model and encryption information
               (e/e: encryption capable/encryption enable).
Serial No      The serial number of the device.
TGT/LUN        The SCSI target and LUN of the device.
Ucode level    The current microcode (firmware) loaded on the device.
World Wide NN  A number indicating Fibre Channel World Wide Node Name of the device.
World Wide PN  A number indicating Fibre Channel World Wide Port Name of the device.
Dev Phy Path   A string indicating the device path in the device tree.
Path Type      A primary or alternate path used for failover.
hba_index      The index number of the particular HBA in the HBA list.
hba_inst       The instance number of the particular HBA assigned by the Solaris system.
hba_driver     The HBA driver name with IBM tape drive attachment.
reg_count      The number of IBM tape drives attached on the HBA.
usage_count    The number of IBM tape drives currently using the HBA.
HBA Path       A string indicating the HBA device path in the device tree.
Addr           The element address where the drive is located in the library.
e/e            The first and second instances of "e" stand for encryption
               capable and encryption enable.
```

The usage of the tapelist program is as follows:

```
-l  Print for all of the configured devices with the  column headers in long list
-L  Display the tape drives information in the tape library
-c  Don't print column headers in long list for all of the configured devices
-t  Display HBA information and current load balancing status
-f  Print the list for a particular file only
-h  Help menu
```

Running tapelist without any options displays the device information line by line for all of the configured devices.

Special Files

After the IBMtape driver is installed, a set of special files is available for performing input/output (I/O) operations to each supported device. The device special file names created by the IBMtape device driver are similar to the SCSI tape special files generally used on Solaris systems.

Each tape instance has a set of minor numbers that provides access to the same physical device, but each minor number provides a different function or behavior for the tape subsystem. These minor numbers are accessed through variations of the special file name for that device. The special files are created in the */dev/rmt* directory. These special files are actually symbolic links to files created within the */devices* subdirectory hierarchy.

Issuing the *ls -la /dev/rmt* command gives useful information about these device special files. The following example shows entries returned by this command for a single IBM tape subsystem. This listing is system dependent, so entries vary slightly in format, depending on the operating system/platform and SCSI adapter support. There may also be entries included for other devices that are not supported by the IBMtape device driver.

```
lrwxrwxrwx root other 79 Aug 26 18:54 0smc   ->
 /devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,
 10000/IBMtape@2,0:smc
lrwxrwxrwx root other 78 Aug 26 18:54 0st    ->
 /devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,
 10000/IBMtape@2,0:st
lrwxrwxrwx root other 79 Aug 26 18:54 0stb   ->
 /devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,
 10000/IBMtape@2,0:stb
lrwxrwxrwx root other 80 Aug 26 18:54 0stbn  ->
 /devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,
 10000/IBMtape@2,0:stbn
lrwxrwxrwx root other 79 Aug 26 18:54 0stc   ->
 /devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,
 10000/IBMtape@2,0:stc
lrwxrwxrwx root other 80 Aug 26 18:54 0stcb  ->
 /devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,
 10000/IBMtape@2,0:stcb
lrwxrwxrwx root other 81 Aug 26 18:54 0stcbn ->
 /devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,
 10000/IBMtape@2,0:stcbn
lrwxrwxrwx root other 80 Aug 26 18:54 0stcn  ->
 /devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,
 10000/IBMtape@2,0:stcn
lrwxrwxrwx root other 79 Aug 26 18:54 0stn   ->
 /devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,
 10000/IBMtape@2,0:stn
```

These entries show the device hierarchy established to support I/O for an IBM SCSI tape system. The attachment path of the device special files spans from the system board, through the S-bus, to the Sun F/W SCSI adapter (supported by the

QLGC, isp SCSI adapter device driver), to the IBM device at SCSI target 2 and LUN 0 (supported by the IBMtape device driver). All nine of these special files are associated with the **same** IBM device (device number 0).

Device Behaviors

Certain device behaviors are determined by which special file in the set is opened for device access. The *smc* special file controls only the medium changer portion of the device and accepts only medium changer operations by way of the *ioctl* entry point. The *smc* special file does not support the read and write entry points. Only one *st* type special file for a particular device may be opened at any one time. The *smc* special file may be opened concurrently with any one of the *st* special files.

The IBMtape device driver decides which types of special files to create during installation, based on the IBM device type being configured. For the IBM 3490E Magnetic Tape Subsystem, only the eight *st* type special files are created. For other IBM tape drives, all nine special files shown previously are created. For IBM tape libraries and autoloaders, only a single *smc* special file is created.

With the information from the previous command, issuing the *ls -la /devices/iommu@f,e0000000/sbus@f,e0001000/QLGC,isp@3,10000* command presents further information about the same special files, as shown in the following example. Again, the actual path information specified in the command will vary from system to system.

```
crw-rw-rw- 1 bin bin 109,1696 Aug 26 18:54 IBMtape@2,0:smc
crw-rw-rw- 1 bin bin 109,1664 Aug 26 18:56 IBMtape@2,0:st
crw-rw-rw- 1 bin bin 109,1728 Aug 26 18:54 IBMtape@2,0:stb
crw-rw-rw- 1 bin bin 109,1732 Aug 26 18:54 IBMtape@2,0:stbn
crw-rw-rw- 1 bin bin 109,1688 Aug 26 18:54 IBMtape@2,0:stc
crw-rw-rw- 1 bin bin 109,1752 Aug 26 18:54 IBMtape@2,0:stcb
crw-rw-rw- 1 bin bin 109,1756 Aug 26 18:54 IBMtape@2,0:stcbn
crw-rw-rw- 1 bin bin 109,1692 Aug 26 18:54 IBMtape@2,0:stcn
crw-rw-rw- 1 bin bin 109,1668 Aug 26 18:54 IBMtape@2,0:stn
```

These entries show the *major* and *minor* numbers associated with each special file. Here, the major number is 109 and identifies to the system that the IBMtape device driver is in support of these special files. Major numbers are assigned by the system at the time the driver is installed and vary from system to system. The nine different minor numbers are specific to the special file names and are used by the device driver to determine which special file was used to access the device and control the device behavior accordingly. For example, the minor number 1696 indicates to the driver that the device was opened by way of the *smc* special file. For more information on device special files and major and minor numbers, consult the Solaris *mtio* man pages.

File Naming Conventions

Table 28 shows the special file naming convention and the associated device attributes recognized by the IBMtape device driver.

Table 28. IBM SCSI Tape/Medium Changer Special Files for Solaris

Special File Name	BSD Compatibility[1]	Rewind on Close[2]	Compression[3]
/dev/rmt/[0–255]smc[4]	N/A	N/A	N/A
/dev/rmt/[0–255]stn[5]	No	No	No
/dev/rmt/[0–255]stcn[5]	No	No	Yes
/dev/rmt/[0–255]st[5]	No	Yes	No
/dev/rmt/[0–255]stc[5]	No	Yes	Yes
/dev/rmt/[0–255]stbn[5]	Yes	No	No
/dev/rmt/[0–255]stcbn[5]	Yes	No	Yes
/dev/rmt/[0–255]stb[5]	Yes	Yes	No
/dev/rmt/[0–255]stcb[5]	Yes	Yes	Yes

Notes:

1. The **BSD (b)** device special file modifies *close* behavior for non-rewind devices. If the device is opened for **no rewind on close** in **non-BSD** mode, and if the last command before closing the device was a read, then the tape is positioned after the filemark immediately following the last block read. If the device is opened for **no rewind on close** in **BSD** mode, and if the last command before closing the device was a read, the tape is left positioned exactly where it was following the last block read. If the device is opened for **rewind on close** the **BSD** mode is not relevant.
2. The **no rewind on close (n)** device special file does not rewind the tape during a *close* operation. Otherwise, the tape is rewound when the device is closed. If the last operation before closing the device was a write or write filemark, then a sufficient number of filemarks is written so that two filemarks will follow the data.

   For the non-rewind special files, the tapes are positioned between the trailing filemarks before closing. If the device is then reopened and more data is written, it is separated by a single filemark from the previous data.
3. The **compression (c)** device special file determines whether the tape device will use built-in hardware compression while storing data on the tape. The compression mode of the device can also be set to the desired state programmatically through the STIOC_SET_PARM *ioctl*, regardless of the default compression mode established by the special file originally used to open the device.
4. The **smc** special file is created only for IBM tape systems that provide medium changer capability. For IBM tape libraries and autoloaders, the *smc* special file is the only file created because the IBMtape device driver supports only the medium changer portion and does not support the tape drive portion of these devices. For the IBM 3490E Magnetic Tape System, there is no *smc* special file created.
5. Only one *st* special file may be opened at one time. The *smc* special file may be opened by itself or in conjunction with one of the *st* type files. The *smc* special file accepts only medium changer commands. Tape drive commands issued to the medium changer fail, with *errno* set to 22, invalid argument.

   Aside from the normal configuration with the medium changer answering as a distinct target/LUN pair, some supported devices can be configured with a nonstandard integrated medium changer reporting at the same target and LUN as the tape drive. In such a case, both *st* and *smc* special files will accept a limited subset of medium changer commands. If you want to use this nonstandard mode, consult the appropriate hardware reference to determine whether the drive supports such a configuration.

Persistent Naming Support

The device special file names are created by the IBMtape driver in the order that the tape devices are presented by the Solaris system. Each device special file name is maintained with the same logical name across reboots, even when an existing device is powered off or not connected.

However, the logical names of devices may be changed due to the swapping of connecting cables, HBA mapping changes, tape device driver updates, or other reasons.

The user can rename the logical name by editing the */etc/devlink.tab* system file for the persistent name binding and reloading the IBMtape driver as follows:

1. Before the persistent name binding, make sure that the IBM tape devices are configured at the different target and LUN addresses if the devices are attached on more than one HBA.

   The Ultrium-3 tape drive is connected to two Emulex HBAs with the same address of target 3 and LUN 0 via a switch in the following example. You have to use the HBA utility, follow HBA vendor instructions, or both to persistently bind the tape devices at the different mapped target and LUN.

```
# tapelist -l
Inst#  Special File     Device      Serial No     TGT/LUN    Device Physical Path
------ -------------- ------------- ------------  --------  -------------------------------------
454   /dev/rmt/2st    ULT3580-TD3   1210003557    3/0       /devices/pci@6,2000/pci@1/fibre-channel@5/IBMtape@3,0
582   /dev/rmt/8st    ULT3580-TD3   1210003557    3/0      /devices/pci@1f,2000/pci@1/fibre-channel@5/IBMtape@3,0
```

   The tape drive is mapped at target 3, LUN 0 on HBA 1 and target 24, LUN 0 on HBA 2 after device persistent binding.

```
 # tapelist -l
Inst#  Special File     Device      Serial No     TGT/LUN    Device Physical Path
------ --------------- -----------  ---------   ----------   ----------------------------------------
454   /dev/rmt/4st    ULT3580-TD3   1210003557     3/0       /devices/pci@6,2000/pci@1/fibre-channel@5/IBMtape@3,0
1136 /dev/rmt/7st     ULT3580-TD3   1210003557    24/0       /devices/pci@1f,2000/pci@1/fibre-channel@5/IBMtape@18,0
```

   Note: Device persistent binding is not provided on Sun HBAs, so this persistent name approach cannot be used with the same physical drive that is attached to multiple Sun HBA ports.

2. Start persistent name binding. In this example, the user renames `4st` and `7st` to `10st` and `11st`.

   a. Create the entry for persistent naming. Determine the target address from the Device Physical Path in the output of **tapelist -l** and add the planned device special file name in the entry. Here, 4st and 7st drives are located at 3,0 (target 3, LUN 0) and 18,0 (target 24 (0x18), LUN 0) at the device physical paths of `/devices/pci@6,2000/pci@1/fibre-channel@5/IBMtape@3,0` and `/devices/pci@1f,2000/pci@1/fibre-channel@5/IBMtape@18,0`. Add the address and device file name into the entries:

```
type=ddi_byte:tape;addr=3,0;    rmt/10\M0
type=ddi_byte:tape;addr=18,0;  rmt/11\M0
```

      Notes:

      1) A tab is entered between addr=3,0; and rmt/10\M0.

      2) The 0 in the entry is the zero in M0.

3) To avoid conflicts with the current device special files assigned by the system automatically, be sure to assign a higher number for the persistent name.
4) The address is w500507630059f007,0 for the tape drive on the Sun HBA with the path of /devices/pci@1,0/pci1022,7450@1/pci1077,141@1/fp@0,0/tape@w500507630059f007,0.

b. Add the above entry into the /etc/devlink.tab system file.

c. Remove existing links created by the IBMtape driver from /dev/rmt by running the **# rm** command.

d. Run the **# devfsadm** command without any options to enable IBMtape to create the new device special file name as defined in the entries in /etc/devlink.tab. A system reboot is also required if the tape device is attached on Sun HBA.

e. Run **tapelist** to list the device special files.

```
# tapelist -l
Inst#  Special File     Device      Serial No     TGT/LUN    Device Physical Path
------ ------------ ------------ ------------- ----------  -------------------------
454   /dev/rmt/10st  ULT3580-TD3   1210003557     3/0      /devices/pci@6,2000/pci@1/fibre-channel@5/IBMtape@3,0
1136 /dev/rmt/11st   ULT3580-TD3   1210003557    24/0      /devices/pci@1f,2000/pci@1/fibre-channel@5/IBMtape@18,0
```

Control Path Failover Support for Libraries

Configuring and Deconfiguring Path Failover Support

Control path failover (CPF) support is enabled automatically by default when the IBMtape device driver is installed on Solaris system. The Solaris IBMtape device driver provides a driver configuration parameter failover for you to enable or disable the library control path failover support. To enable the CPF support for all of the paths, no any action is required. To disable the CPF support for all of the paths or a particular path, use the following steps:

1. To disable CPF support for all the paths, add and set the failover parameter to off at the beginning of IBMtape.conf file in the directory of /usr/kernel/drv.
2. To disable a particular path, add and set the failover parameter to off in the path entry in IBMtape.conf file. For example, name="IBMtape" class="scsi" target=3 lun=1 failover=0;
3. Stop the TMD (tape monitor daemon) running on the system and unload the IBMtape driver module from the current kernel:

   ```
   # /opt/IBMtape/tmd -s
   # /usr/sbin/rem_drv IBMtape
   ```
4. Reload the IBMtape driver module in the kernel and start the daemon:

   ```
   # /usr/sbin/add_drv -m ' 0666 bin bin' IBMtape
   # /opt/IBMtape/tmd
   ```

Primary and Alternate Paths

When the device driver configures a logical device with path failover support enabled, the first device configured always becomes the primary path. When a second or more logical device is configured with path failover support enabled for the same physical device, it configures as an alternate path. The device driver supports up to 16 physical paths for single a device.

The primary and alternate path information can be obtained in the field of "Path Type" running the /opt/IBMtape/tapelist command output will be similar to the example in Figure 11.

```
#tapelist -1
Inst# Special File    Device    Serial No    TGT/LUN  Ucode   WWNN     WWPN
       Device Physical Path                  Path Type
-----------------------------------------------------------------------------------------------
-------------------------------------------------------
686   /dev/rmt/12smc 03584L32  0000000T0039    1/1    402j    N/A      N/A
 /devices/pci@If,2000/QLGC,qla@1/IBMtape@1,1 Primary
688   /dev/rmt/14smc 03584L32  0000000T0039    2/1    402j    N/A      N/A
 /devices/pci@If,2000/QLGC,qla@1/IBMtape@2,1 Alt_path_1
694   /dev/rmt/26smc 03584L32  0000000T0039    5/1    402j    N/A      N/A
 /devices/pci@If,2000/QLGC,qla@1/IBMtape@5,1 Alt_path_2
```

Figure 11. Example of Control Path Failover Support Command Output

The labeling of a logical device as either a primary or alternate path is for information only; it is used to:

1. Identify the actual number of physical devices configured on the system and a specific logical device associated with them. There will be only one logical device labeled the primary path for each physical device. However, there can be multiple logical devices labeled as an alternate path for the same devices.
2. Provide information about which logical devices configured on the system have path failover enabled.

Querying Primary and Alternate Path Configuration

You can display the primary and alternate path configuration for all devices with the tapelist utility.

You can also display the primary and alternate path configuration for any device running the tapeutil utility and selecting option 54 **Query Device Paths**. For example,

```
Path list in the device:
      Path 0  Path 1  Path 2

Current active path: Path 0

Enter path number or <enter> for all of the paths:

Instance Number ............. 201
SCSI Target ................. 4
SCSI LUN .................... 1
Serial Number ............... 000001301084
Ucode Level ................. 227B
Path World Wide NN .......... N/A
Path World Wide PN .......... N/A
Path Type ....................Primary
Path .........................Enable
Device Physical Path ........ /devices/pci@1f,2000/scsi@1/IBMtape@4,1

Instance Number ............. 231
SCSI Target ................. 3
SCSI LUN .................... 1
Serial Number ............... 000001301084
Ucode Level ................. 227B
Path World Wide NN .......... N/A
Path World Wide PN .......... N/A
Path Type ................... Alt_path_1
Path ........................ Enable
Device Physical Path ........ /devices/pci@4,2000/pci@1/fibre-channel@4/IBMtape@3,1
```

```
Instance Number ............. 293
SCSI Target ................. 2
SCSI LUN .................... 1
Serial Number ............... 000001301084
Ucode Level ................. 227B
Path World Wide NN .......... N/A
Path World Wide PN .......... N/A
Path Type ................... Alt_path_2
Path ........................ Enable
Device Physical Path ........ /devices/pci@6,4000/lpfc@4/IBMtape@2,1
```

This option shows specific information for each path, such as the instance number for the logical device, device serial number, the target id, the logical unit number, the current enabled status, path type and so on. It also indicates how many paths are configured for the device, and which path is active currently.

Disable and Enable Primary and Alternate Paths

Once you install the IBMtape device driver, by default, all the available paths for a physical device are enabled. If for some maintenance reason you need to disable a path and do not want to fail over to this path, you can run the following steps:

1. Turn the `failover` parameter to off in the path entry in the *IBMtape.conf* file. For example,

   ```
   name="IBMtape" class="scsi" target=3 lun=1 failover=0;
   ```

2. Stop the TMD (tape monitor daemon) running on the system and unload the IBMtape driver module from the current kernel:

   ```
   # /opt/IBMtape/tmd -s
   # /usr/sbin/rem_drv IBMtape
   ```

3. Reload the IBMtape driver module in the kernel and start the daemon:

   ```
   # /usr/sbin/add_drv -m ' 0666 bin bin' IBMtape
   # /opt/IBMtape/tmd
   ```

To enable a path from a disabled state, you can run the following steps:

1. Remove the parameter of `failover=0` in the path entry in *IBMtape.conf* file. For example,

   ```
   name="IBMtape" class="scsi" target=3 lun=1"
   ```

2. Stop the TMD (tape monitor daemon) running on the system and unload the IBMtape driver module from the current kernel:

   ```
   # /opt/IBMtape/tmd -s
   # /usr/sbin/rem_drv IBMtape
   ```

3. Reload the IBMtape driver module in the kernel and start the daemon:

   ```
   # /usr/sbin/add_drv -m ' 0666 bin bin' IBMtape
   # /opt/IBMtape/tmd
   ```

Data Path Failover and Load Balancing Support for Tape Drives

Note: The tape drive failover feature code must be installed prior to enabling the DPF for IBM Ultrium tape drive in the Solaris IBMtape device driver. Refer to “Supported Devices and Feature Codes” on page 14 to determine which feature code is required for your machine type.

Configuring and Deconfiguring Path Failover Support

Path failover support for tape drives is not enabled automatically when the device driver is installed. It must be configured initially on each logical device after installation. When path failover support is enabled for a logical device, it remains set until the device is deleted or the support is deconfigured. The path failover setting is retained even if the system is rebooted. Path failover support can be enabled on all configured devices at one time, or it can be enabled or disabled selectively by logical device. It may be desirable at times to configure some, but not all, logical paths to a device with the support enabled. Follow the above steps to enable the DPF support:

1. To enable the support globally on all currently configured devices, add an entry of `dpf_support=1` at the beginning of the *IBMtape.conf* file, such as
```
dpf_support=1;
```
2. Or, to enable a particular path, add the parameter `dpf_support` and turn it on in the path entry in the *IBMtape.conf* file. For example,
```
name="IBMtape" class="scsi" target=3 lun=0 dpf_support=1;
```
3. For the IBM Ultrium tape drive, you need to enter the DPF feature key(s) in the parameter `dpf_keys` at the beginning of the *IBMtape.conf* file in the directory /usr/kernel/drv. For example:
```
dpf_keys="A729E60F7B119411, C7A0B9ef2c1a4360, a729e60f7b118460";
```
Notes:
   a. The parameter dpf_keys is in the format "*key1*, *key2*, *key3*, ... ". Each key is 16 characters long with a comma "," and a space " ". The IBMtape driver supports up to 36 dpf keys.
   b. DPF keys do not need to be added in *IBMtape.conf* if you are running the latest drive code on Ultrium-3 and Ultrium-4 drives.
4. Stop the TMD (tape monitor daemon) running on the system and unload the IBMtape driver module from the current kernel
```
# /opt/IBMtape/tmd -s
# /usr/sbin/rem_drv IBMtape
```
5. Reload the IBMtape driver module in the kernel and start the daemon:
```
# /usr/sbin/add_drv -m ' 0666 bin bin' IBMtape
#  /opt/IBMtape/tmd
```

This will deconfigure all devices to remove or comment out the parameter `dpf_support` in the *IBMtape.conf* file, and reboot the system or deconfigure and reconfigure all devices. For example,

1. To disable the support globally on all currently configured devices, remove or comment out the entry `dpf_support=1` at the beginning of the *IBMtape.conf* file.
```
# dpf_support=1;     * comment out the parameter
```
2. Stop the TMD (tape monitor daemon) running on the system and unload the IBMtape driver module from the current kernel:
```
# /opt/IBMtape/tmd -s
# /usr/sbin/rem_drv IBMtape
```
3. Reload the IBMtape driver module in the kernel and start the daemon:
```
# /usr/sbin/add_drv -m ' 0666 bin bin' IBMtape
#  /opt/IBMtape/tmd
```

To disable the support on a single logical device, following these steps:

1. To enable the support globally on all currently configured devices, add an entry `dpf_support=1` at the beginning of the *IBMtape.conf* file, such as
```
dpf_support=1;
```
2. To disable a particular path, add the parameter `dpf_support` and turn it off in the path entry in the *IBMtape.conf* file. For example,
```
 name="IBMtape" class="scsi" target=3 lun=0 dpf_support=0;
```

3. Stop the TMD (tape monitor daemon) running on the system and unload the IBMtape driver module from the current kernel:

```
# /opt/IBMtape/tmd -s
# /usr/sbin/rem_drv IBMtape
```

4. Reload the IBMtape driver module in the kernel and start the daemon:

```
# /usr/sbin/add_drv -m ' 0666 bin bin' IBMtape
#  /opt/IBMtape/tmd
```

Primary and Alternate Paths

When the device driver configures a logical device with path failover support enabled, the first device configured always becomes the primary path. When a second logical device is configured with path failover support enabled for the same physical device, it configures as an alternate path. A third logical device is configured as the next alternate path, and so on. The device driver supports up to 16 physical paths for a single device.

For example, if 0st (port 0 of 3592) is configured first, then 5st (port 1), 18st (port 1) and 21st (port 0) to the two HBAs through a switch (here, WWPN 5005076302400127 from port 0 and 5005076302800127 from port 1), the /opt/IBMtape/tapelist command output will be similar to the example in Figure 12.

```
#tapelist -1
Inst# Special File   Device      Serial No   TGT/LUN  Ucode      WWNN             WWPN
        Device Physical Path                   Path Type
-----------------------------------------------------------------------------------------------
----------------------------------------------------------
685   /dev/rmt/0st  03592J1A    000001300168   1/0    04CE   5005076302000127 5005076302400127
 /devices/pci@1f,2000/QLGC,qla@1/IBMtape@1,0   Primary
697   /dev/rmt/5st  03592J1A    000001300168   7/0    04CE   5005076302000127 5005076302800127
 /devices/pci@1f,2000/QLGC,qla@1/IBMtape@7,0   Alt_path_1
666   /dev/rmt/18st 03592J1A    000001300168   1/0    04CE   5005076302000127 5005076302800127
 /devices/pci@1f,4000/JNI,FCR@2/IBMtape@1,0    Alt_path_2
670   /dev/rmt/21st 03592J1A    000001300168   3/0    04CE   5005076302000127 5005076302400127
 /devices/pci@1f,4000/JNI,FCR@2/IBMtape@3,0    Alt_path_3
```

Figure 12. Example of Data Path Failover Support Command Output

The labeling of a logical device as either a primary or alternate path is for information only, in order to:

1. Be able to identify the actual number of physical devices configured on the system and a specific logical device associated with them. There will be only one logical device labeled the primary path for each physical device. However, there may be many (multiple) logical devices labeled as an alternate path for the same devices.
2. Provide information about which logical devices configured on the system have path failover support enabled.

Querying Primary and Alternate Path Configuration

You can display the primary and alternate path configuration for all devices with the tapelist utility.

You can also display the primary and alternate path configuration for any device running the tapeutil utility and selecting option 54 **Query Device Paths**. For example,

```
Path list in the device:
  Path 0  Path 1  Path 2  Path 3
Current active path: Path 0

Enter path number or <enter> for all of the paths:

 Instance Number ................. 685
 SCSI Target ..................... 1
 SCSI LUN ........................ 0
 Serial Number ................... 000001300168
 Ucode Level ..................... 04CE
 Path World Wide NN ............. 5005076302000127
 Path World Wide PN ............. 5005076302400127
 Path Type ....................... Primary
 Path ............................ Enable
 Device Physical Path ............ /devices/pci@1f,2000/QLGC,qla@1/IBMtape@1,0

 Instance Number ................. 697
 SCSI Target ..................... 7
 SCSI LUN ........................ 0
 Serial Number ................... 000001300168
 Ucode Level ..................... 04CE
 Path World Wide NN ............. 5005076302000127
 Path World Wide PN ............. 5005076302800127
 Path Type ....................... Alt_path_1
 Path ............................ Enable
 Device Physical Path ............ /devices/pci@1f,2000/QLGC,qla@1/IBMtape@7,0

 Instance Number ................. 666
 SCSI Target ..................... 1
 SCSI LUN ........................ 0
 Serial Number ................... 000001300168
 Ucode Level ..................... 04CE
 Path World Wide NN ............. 5005076302000127
 Path World Wide PN ............. 5005076302800127
 Path Type ....................... Alt_path_2
 Path ............................ Enable
 Device Physical Path ............ /devices/pci@1f,4000/JNI,FCR@2/IBMtape@1,0

 Instance Number ................. 670
 SCSI Target ..................... 3
 SCSI LUN ........................ 0
 Serial Number ................... 000001300168
 Ucode Level ..................... 04CE
 Path World Wide NN ............. 5005076302000127
 Path World Wide PN ............. 5005076302400127
 Path Type ....................... Alt_path_3
 Path ............................ Enable
 Device Physical Path ............ /devices/pci@1f,4000/JNI,FCR@2/IBMtape@3,0
```

This option shows specific information for each path, such as the instance number for the logical device, device serial number, the target id, the logical unit number, the current enabled status, path type and so on. It also indicates how many paths are configured for the device, and which path is the current active path.

Disable and Enable Primary and Alternate Paths

Once you enter the parameter `dpf_support` in the *IBMtape.conf* file and install the IBMtape device driver, all the available paths for a physical device are enabled. If for some maintenance reason you need to disable a path and do not want to fail over to this path, you may run the following steps:

1. Add the parameter of ″dpf_support″ and turn it off in the path entry in IBMtape.conf file. For example,

   ```
   name="IBMtape" class="scsi" target=3 lun=0 dpf_support=0;
   ```

2. Stop the TMD (tape monitor daemon) running on the system and unload the IBMtape driver module from the current kernel:

```
# /opt/IBMtape/tmd -s
# /usr/sbin/rem_drv IBMtape
```

3. Reload the IBMtape driver module in the kernel and start the daemon:

```
# /usr/sbin/add_drv -m ' 0666 bin bin' IBMtape
#  /opt/IBMtape/tmd
```

To enable a path from a disabled state, you may run the following steps:

1. Remove the parameter `dpf_support=0` from the path entry in *IBMtape.conf* file. For example,

```
name="IBMtape" class="scsi" target=3 lun=0;
```

2. Stop the TMD (tape monitor daemon) running on the system and unload the IBMtape driver module from the current kernel:

```
# /opt/IBMtape/tmd -s
# /usr/sbin/rem_drv IBMtape
```

3. Reload the IBMtape driver module in the kernel and start the daemon:

```
# /usr/sbin/add_drv -m ' 0666 bin bin' IBMtape
#  /opt/IBMtape/tmd
```

System-Managed Encryption

Device Driver Configuration

System-Managed Encryption can be set on global or a specific tape drive in IBMtape.conf in /usr/kernel/drv. There are 2 new configuration parameters added for encryption:

sys_encryption_proxy "ON/OFF" Use System Encryption FCP Proxy Manager

sys_encryption_write "OFF/ON/CUSTOM" System Encryption for Write Commands at BOP

The sys_encryption_proxy parameter enables device driver system-managed encryption for a tape drive by setting the value to ON (default set).

The sys_encryption_write parameter controls if the device driver will set the tape drive to encryption enabled for write commands. When set to OFF, the tape drive will use encryption for read operations and write operations will not use encryption. When set to ON, the tape drive will use encryption for both read/write operations. When set to CUSTOM, the device driver will not modify current tape drive setting. The custom setting is intended for applications using system-managed encryption to control write encryption without device driver intervention. The parameter is set to "CUSTOM" by default.

Note: If sys_encryption_write is set to ON, an application can not open a tape drive using the append mode.

To make a global setting to enable SME in IBMtape.conf:

sys_encryption_write=1; # System Encryption for Write Commands at BOP

To enable SME for a particular target:

name="IBMtape"

class="scsi"

target=0

lun=0
block_size=0
buffering=1
immediate=0
trailer=0
sili=0
sys_encryption_write=1;

To disable SME in a particular target:

name="IBMtape"
class="scsi"
target=0
lun=0
block_size=0
buffering=1
immediate=0
trailer=0
sili=0
sys_encryption_proxy=0;

Querying Tape Drive Configuration

There are is a new tapeutil command added for encryption to query tape drive encryption settings.

The current tape drive encryption settings can be queried using the tapeutil menu with the option “Get Drive Encryption Settings” or the command line “tapeutil –f /dev/rmt/device_special_file encryption”.

Following is an example of a correct tape drive configuration with sys_encryption_write set to “on”. If the sys_encryption_write parameter is set to “no”, the tape drive “Encryption State” will be Off.

```
# tapeutil -f /dev/rmt/38stn encryption
```

GET_ENCRYPTION_STATE command succeeded.

Encryption settings:

Encryption Capable ... Yes
Encryption Method .... System
Encryption State........ ON

Testing Data Encryption Configuration and Connectivity

There is a new tapeutil command added to validate the `ibmekm.conf` file server entries and test tape drive to server connectivity operations. This test can be run using the tapeutil menu in the option “EKM Test” or the command line “tapeutil –f/dev/rmt/device_special_file ekmtest”.

The first test checks the server configuration defined in the `ibmekm.conf` file and then communication to the configured servers. This test reports back the number of servers available. The second test runs a basic diagnostic that checks the tape drive to server communication and reports success or fail. The third test runs an enhanced diagnostic that checks a key operation between the tape drive and server then reports success or fail.

Following is an example of a successful test:

```
# tapeutil –f /dev/rmt/38stn ekmtest
Testing server configuration and connections...
Test complete, servers available 2
Running basic drive to server encryption test...
Test complete, completion code 0
Running full drive to server encryption test...
Test complete, completion code 0
```

If the first server test fails with one of the following errors, then the remaining tests will not be run. Perform the recommended problem determination in the order below to resolve the problem.

1. “Invalid argument or Operation not supported on device” - System encryption is not configured for the tape drive in the device driver. Set sys_encryption_write to “ON” in IBMtape.conf for the tape drive. If sys_encryption_write can not be set then the tape drive is not encryption capable or encryption support disable.
2. “Can’t assign requested address” - Either the `ibmekm.conf` file is missing or is invalid. Check and correct.
3. “Network is down” – All configured servers in the `ibmekm.conf` file are not available. Check all servers configured in the `ibmekm.conf` file are currently running on the server IP address and configured correctly.
4. “Permission denied” - If the full drive to server encryption test fails with a permission error , unload the tape and run the command again. If the problem is persistent, check if either the EKM server isn’t available recently or the drive isn’t claimed in EKM server.

Field Support Information

When encryption failures require field support and/or development analysis, run /opt/IBMtape/diags_info script to generate a file of diags.out. Tape drive dumps and EKM server logs may be needed in addition to this information.

Tape Utility Program (tapeutil)

A SCSI Tape and Medium Changer Utility Program called *tapeutil* is provided with the IBM SCSI Tape and Medium Changer Device Driver for Solaris and is installed in the */opt/IBMtape* directory as part of the IBMtape package. The program provides the following service aids for IBM tape subsystems.

- Download Device Microcode
- Force Device Diagnostic Dump
- Query Device Microcode Level
- Query Device Serial Number
- Query Device Type/Verify Device Attachment
- Store Device Diagnostic Dump

The source code for the *tapeutil* program is provided for example purposes and is installed in the */opt/IBMtape* directory during the IBMtape package installation. This source code is commented and demonstrates calls to all of the supported device driver entry points and *ioctl* commands, thus giving the application developer a starting point for interfacing to the IBMtape device driver.

The *tapeutil* program provides both an interactive mode and a command-line mode. If the *tapeutil* program is called with no command-line parameters, the interactive mode is started. In the interactive mode, the device to be operated on should first be opened using option 1. Other options may then be selected. The user is prompted for additional information if required for the specific options selected. The results of a command are displayed after it is executed. If an error occurs for the command, error information and device sense data is displayed. The device can be closed using option 2, or it is closed automatically when the Quit option is selected. The menu is displayed automatically one time when the program is first called. To prevent unnecessary scrolling of the screen, the menu is not displayed automatically again after each command but instead is refreshed only after the M (menu refresh) command is entered.

Interactive Mode

The program provides a menu driven test tool for exercising or testing IBM tape and medium changer devices with a full suite of supported operations.

- Cartridge Inventory
- Mounting/Demounting Cartridges
- Reading/Writing Data
- Setting/Displaying Device Information/Status
- Tape Motion Commands

Command-Line Mode

In addition to the menu driven front end, the *tapeutil* program provides a command-line mode that is convenient for use in shell scripts.

Note: When using the command-line calls to the *tapeutil* program, the tape device is opened and closed for each invocation. Configuration parameters that are changed in one call to the *tapeutil* program are returned to default values when the device is closed.

If command-line parameters are provided when the program is called, the command-line mode is started. For each command-line execution of the program, the device is first opened, the specific command is issued, and the device is then closed. The program can be driven from within a shell script if desired. Results of the operation are displayed only when executed in verbose mode. No information is displayed when not in verbose mode. This is particularly useful for quiet shell script implementations. A completion code, as defined in */usr/include/sys/errno.h*, for the operation requested is always returned from the program upon exit (in both verbose and quiet mode).

The usage of the *tapeutil* program in command-line mode is as follows:

`tapeutil -f` **`device`** `-o` **`operation`** `[`**`options`**`]`

where **device** is the name of the tape device special file (for example: */dev/rmt/1st*) and **operation** is one of the following values. The **device** special file and the **operation** are required. The specific **options** associated with a particular operation are indicated. Parameters enclosed in square brackets are optional. All others are required.

Service Commands

Query Serial Number	`tapeutil -f f -o qsn [-w w] [-v]`
Query Microcode Level	`tapeutil -f f -o qmc [-w w] [-v]`

Force Dump	`tapeutil -f` **f** `-o fdp [-w` **w**`] [-v]`
Store Dump	`tapeutil -f` **f** `-o sdp [-w` **w**`] [-v] -z` **z**
Download Microcode	`tapeutil -f` **f** `-o dmc [-w` **w**`] [-v] -z` **z**
Format Cartridge	`tapeutil -f` **f** `-o fmt [-w` **w**`] [-v]`
Query Device Type	`tapeutil -f` **f** `-o chk [-w` **w**`] [-v]`

Basic SCSI Commands

Test Unit Ready	`tapeutil -f` **f** `-o tur [-w` **w**`] [-v]`
Inquiry	`tapeutil -f` **f** `-o inq [-w` **w**`] [-v]`
Request Sense	`tapeutil -f` **f** `-o req [-w` **w**`] [-v]`
Reserve	`tapeutil -f` **f** `-o res [-w` **w**`] [-v]`
Release	`tapeutil -f` **f** `-o rel [-w` **w**`] [-v]`

Medium Changer Commands

Move Medium	`tapeutil -f` **f** `-o mov [-w` **w**`] [-v] -s` **s** `-d` **d**
Position To Element	`tapeutil -f` **f** `-o pos [-w` **w**`] [-v] -s` **s**
Element Information	`tapeutil -f` **f** `-o ele [-w` **w**`] [-v]`
Inventory	`tapeutil -f` **f** `-o inv [-w` **w**`] [-v]`
Audit	`tapeutil -f` **f** `-o aud [-w` **w**`] [-v]`
Lock/Unlock Door	`tapeutil -f` **f** `-o lck [-w` **w**`] [-v] -x` **x**

Tape Drive Commands

Read	`tapeutil -f` **f** `-o rea [-w` **w**`] [-v] -b` **b** `-n` **n** `-m` **m**
Write	`tapeutil -f` **f** `-o wri [-w` **w**`] [-v] -b` **b** `-n` **n** `-m` **m** `[-r` **r**`] [-z` **z**`]`
Write File Mark	`tapeutil -f` **f** `-o eof [-w` **w**`] [-v] -c` **c**
Erase Tape	`tapeutil -f` **f** `-o era [-w` **w**`] [-v]`
Rewind	`tapeutil -f` **f** `-o rew [-w` **w**`] [-v]`
Retension	`tapeutil -f` **f** `-o ret [-w` **w**`] [-v]`
Offline	`tapeutil -f` **f** `-o off [-w` **w**`] [-v]`
Load/Unload Tape	`tapeutil -f` **f** `-o lod [-w` **w**`] [-v] -t` **t**
Forward Space File	`tapeutil -f` **f** `-o fsf [-w` **w**`] [-v] -c` **c**
Backward Space File	`tapeutil -f` **f** `-o bsf [-w` **w**`] [-v] -c` **c**
Forward Space Record	`tapeutil -f` **f** `-o fsr [-w` **w**`] [-v] -c` **c**
Backward Space Record	`tapeutil -f` **f** `-o bsr [-w` **w**`] [-v] -c` **c**
Locate End of Data	`tapeutil -f` **f** `-o eod [-w` **w**`] [-v]`
Get Record Size	`tapeutil -f` **f** `-o grs [-w` **w**`] [-v]`
Set Record Size	`tapeutil -f` **f** `-o srs [-w` **w**`] [-v] -x` **x**
Get Device Status	`tapeutil -f` **f** `-o gds [-w` **w**`] [-v]`
Get Device Information	`tapeutil -f` **f** `-o gdi [-w` **w**`] [-v]`
Get Media Information	`tapeutil -f` **f** `-o gmi [-w` **w**`] [-v]`

Get Position	`tapeutil -f` **f** `-o gpo [-w` **w**`] [-v] -t` **t**
Set Position	`tapeutil -f` **f** `-o spo [-w` **w**`] [-v] -t` **t** `-x` **x**
Get Parameter	`tapeutil -f` **f** `-o gpa [-w` **w**`] [-v] -t` **t**
Set Parameter	`tapeutil -f` **f** `-o spa [-w` **w**`] [-v] -t` **t** `-x` **x**
Sync Buffer	`tapeutil -f` **f** `-o syn [-w` **w**`] [-v]`
Display Message	`tapeutil -f` **f** `-o msg [-w` **w**`] [-v] -t` **t** `-y` **y1,y1**

Note: Invoking the *tapeutil* program with the **-h** flag (for example, **tapeutil -h**) or or **-?** flag (for example, **tapeutil -?**) will display the usage help information.

Flag Description

The supported flags, their meanings, their associated operations, and their acceptable ranges are as follows:

-? Usage Help (stand alone flag) {no value required}

-b Block Size (rea, wri)

{0 < (block size x blocking factor) < 2097152}

-c Operation Count (eof, fsf, fsr, bsf, bsr) {0–65535}

-d Destination Address (mov)

{device specific, determine range from Element Info}

-f Device Special File Name (always required)

{*/dev/rmt/1st* or similar}

-h Usage Help (stand alone flag) {no value required}

-m Multiples to Read or Write (rea, wri) {0–2097152}

-n Blocking Factor (rea, wri)

{0 > (block size x blocking factor) < 2097152}

-o Operation (always required) {refer to the previous list}

-r Random Seed (wri) {0–65535}

-s Source Address (mov, pos)

{device specific, determine range from Element Info}

-t Type of Parameter Value

- (gpo) {1=logical block, 2=physical block}
- (spo) {1=logical block, 2=physical block}
- (gpa) {1=block size, 2=compression, 3=buffering, 4=immediate, 5=trailer, 6=write protect, 7=acf mode, 8=capacity, 9=sili}
- (spa) {1=block size, 2=compression, 3=buffering, 4=immediate, 5=trailer, 6=write protect, 8=capacity, 9=sili}
- (msg) {1=display msg0, 2=display msg1, 3=flash msg0, 4=flash msg1, 5=alternate msg1/msg2}
- (lod) {1=load, 2=unload}

-v Verbose Mode (optional for all commands, stand alone flag)

{no value required, absence of flag means quiet mode}

-w Open Mode (optional for all commands)

{1=read/write, 2=read only (default), 3=write only, 4=append}

-x Parameter Value
- (lck) {1=lock, 2=unlock}
- (srs) {0–65536}
- (spo) {0–65535}
- (spa) {0–65535}

-y Messages (msg) {message1,message2}

-z Input/Output File Name
- (sdp) {path and name of the file in which to store dump}
- (dmc) {path and name of the microcode image file}
- (wri) {path and name of the file containing write data pattern}

Notes:

1. For read and write operations, the size of one buffer of data transferred during a single SCSI read or write command is determined by the product of the *Block Size* value and the *Blocking Factor* value. The number of these buffers transferred is determined by the *Multiplier* value. The actual total number of bytes transferred is then Block Size x Blocking Factor x Multiplier. If the device is set to fixed block mode (block size not equal to zero), the product of *Block Size* and *Blocking Factor* must be a multiple of the device block size setting.
2. For further information about the Get Parameter (*gpa*) and Set Parameter (*spa*) operations, refer to the STIOC_GET_PARM and STIOC_SET_PARM *ioctl* commands described in the *IBM Tape Device Drivers: Programming Reference*, GA32-0566.

   The following examples should help demonstrate and clarify the command-line usage of the *tapeutil* program. For all examples, substitute the actual value of the special file associated with the target device.
   - To query the serial number of the device:
     ```
     /opt/IBMtape/tapeutil -f /dev/rmt/0st -o qsn -v
     ```
   - To request inquiry data from the device:
     ```
     /opt/IBMtape/tapeutil -f /dev/rmt/0st -o inq -v
     ```
   - To move a cartridge from cell 32 to the tape drive (16):
     ```
     /opt/IBMtape/tapeutil -f /dev/rmt/0smc -o mov -s 32 -d 16 -v
     ```
   - To set the block size of the device to 64K:
     ```
     /opt/IBMtape/tapeutil -f /dev/rmt/0st -o spa -t 1 -x 65535 -v
     ```
   - To write one hundred 64K blocks of data to the tape device:
     ```
     /opt/IBMtape/tapeutil -f /dev/rmt/0stn -w 1 -o wri -b 65535
     -n 1 -m 100 -v
     ```
   - To write two filemarks to the tape device:
     ```
     /opt/IBMtape/tapeutil -f /dev/rmt/0stn -w 1 -o eof -c 2 -v
     ```
   - To rewind the tape device:
     ```
     /opt/IBMtape/tapeutil -f /dev/rmt/0stn -o rew -v
     ```
   - To read one hundred 64K blocks of data from the tape device:
     ```
     /opt/IBMtape/tapeutil -f /dev/rmt/0stn -o rea -b 65535 -n 1 -m 100 -v
     ```
3. For 3592 Media types, capacity scaling is only supported for the standard 300 GB rewritable data cartridge.

4. Attempting to set capacity scaling that is not supported by a device or the current media loaded will always return 100% and cannot be changed. For example, the 60GB (Economy Data) cartridge for the IBM 3592 cannot be capacity scaled and will always be 100%.

Downloading Device Microcode

The IBM tape drives support downloading a new microcode image from the host system by way of the SCSI bus.

To download microcode to these devices, perform the following steps:

- Verify that the IBM tape system is powered On and online.
- Verify that the tape drive does not have a tape cartridge currently loaded.
- Determine the current version of microcode present on the device by entering the following command, substituting **n** with the actual value from the device special file associated with the target device:

  `/opt/IBMtape/tapeutil -f /dev/rmt/nst -o qmc -v`
- Determine the source of the new microcode image file (that is, diskette, such as */vol/dev/aliases/floppy0* or host file, such as */tmp/ucode_image.file*).
- Enter the following command, substituting **file** with the actual path and filename of the microcode image file and substituting **n** with the actual value from the device special file associated with the destination device:

  `/opt/IBMtape/tapeutil -f /dev/rmt/nst -o dmc -z file -v`

Notes:

1. The microcode download procedure may also be performed using the interactive mode of the *tapeutil* program. To call the *tapeutil* program in this format, enter the following command, then click option 1 to open the device, followed by option 4 to query the current device microcode level, and option 7 to download new microcode:

   `/opt/IBMtape/tapeutil`
2. If the current microcode on a tape drive is FIPS (Federal Information Processing Standard) code, then a window will display `Warning: The drive is currently using FIPS code.  Do you want to continue with downloading new drive code? [y/n]:`

   Enter `n` to exit without downloading new code, otherwise enter `y` to continue with the download code procedure.

Forcing and Storing Device Diagnostic Dump

The IBM tape drives support forcing a diagnostic dump and storing that dump to a mounted tape cartridge or to a host system file through the SCSI bus.

To store a diagnostic dump on these devices, perform the following steps:

1. Verify that the IBM tape system is powered On and online.
2. Enter the following to force the dump, substituting *n* with the actual value from the device special file associated with the target device:

   `/opt/IBMtape/tapeutil -f /dev/rmt/nst/-o sdp -z file -v`
3. Determine the location for the dump to be stored (that is, diskette, such as */vol/dev/aliases/floppy0* or host file, such as */tmp/diag_dump.file*).
4. Enter the following command, substituting **file** with the actual path and filename of the dump destination file and substituting **n** with the actual value from the device special file associated with the dump device:

   `/opt/IBMtape/tapeutil -f /dev/rmt/nst -o sdp -z file -v`

5. The dump may also be written to a tape cartridge mounted in the dump device using *tapedrive* for **file**.

Note: The diagnostic dump procedure may also be performed using the interactive mode of the *tapeutil* program. To call the *tapeutil* program in this format, enter the following command, then click option 1 to open the device and option 6 to store the dump:

```
/opt/IBMtape/tapeutil
```

Problem Determination

The following sections describe the service and diagnostic aids that are part of the IBM SCSI Tape and Medium Changer Device Driver for Solaris package. Procedures for verifying correct installation of the device, basic problem determination guidelines, and outlines of the utility program included with the IBMtape package are included.

Functional Verification

If you wish to verify that the installation of the IBM SCSI Tape and Medium Changer Device Driver for Solaris package was successful, follow these steps:

1. Enter this command to verify installation was successful

   ```
   /usr/bin/pkginfo IBMtape
   ```

   The following information should be displayed:

   ```
   system  IBMtape  IBM SCSI Tape & Medium Changer Device Driver x.x.x.x
   ```

   where **x.x.x.x** is the version of the device driver.
2. To verify that device driver support for a specific IBM tape system attached to the system is functioning correctly, enter the following command:

   ```
   /opt/IBMtape/tapeutil -f /dev/rmt/nst -o chk -v
   ```

   substituting for **n** the number associated with the device special file assigned to the IBM tape system that you want to check. Listing the contents of the */dev/rmt* directory (using the *ls* command) can be helpful in determining the proper special file name. For medium changer devices, the special file name */dev/rmt/***n***smc* should be used.

   The following information should be displayed:

   ```
   IBM xxxxxxxx configured at /dev/rmt/nst.
   ```

   where **xxxxxxxx** is the model number of the IBM tape system and **n** is the same number specified in the verify command.
3. To verify that the IBMtape device driver is loaded in kernel memory, enter the following command:

   ```
   /usr/sbin/modinfo | /usr/bin/grep IBMtape
   ```

   The following information should be displayed:

   ```
   165 f5f10000  15c0s 109   1  IBMtape (IBM SCSI Tape/Medium Changer DD)
   ```

The first five fields shown will probably not match your specific output. The fields indicate the ID, load address, size, major number, and revision for the IBMtape device driver and vary from machine to machine

Installation Problems

If you are experiencing problems with installation of the IBM SCSI Tape and Medium Changer Device Driver for Solaris package, the following information may be of assistance. If you cannot solve the problems after checking the following, contact the appropriate IBM service representative.

- If you receive the following message during installation:

  ```
  drvconfig: System call 'modctl_modconfig' failed:
    No such device or address.
  Warning: Driver (IBMtape) configuration failed.
    System could not install driver.
  ```

  it indicates that the IBMtape device driver was not loaded because it did not detect the presence of any supported IBM devices on the SCSI bus. Verify that SCSI adapter device driver support is installed and configured correctly. Verify that the IBM tape subsystem is connected properly to the SCSI bus, powered On, and online. It is not necessary for the tape drive to have a cartridge loaded to be recognized by the IBMtape device driver.
- If you cannot open an IBM device, verify that you are using the correct special file. The IBM tape special files are of the form **st** in the */dev/rmt* directory. The IBM medium changer special files are of the form **smc* in the */dev/rmt* directory. Ensure that the Sun native tape device driver (*st*) is not contending for the same IBM device by consulting the *st.conf* file in the */kernel/drv* directory and commenting out conflicting stanzas.

Tape Monitor Daemon (tmd)

The Tape Monitor Daemon is introduced in the version of IBMtape.4.0.9.2 or later. It is designed to run concurrently with the IBMtape driver and to automatically retrieve and store the IBM tape drive diagnostic information (drive dump) into the /var/opt/IBMtape directory. The daemon is automatically started when the driver is installed, even no any tape device is attached on the system. An entry ″ name=″IBMtape″ parent=″pseudo″ instance=16383; ″ is also entered into the configuration file of /usr/kernel/drv/IBMtape.conf automatically for the daemon during the IBMtape driver installation.

The following options can be used to configure the tape monitor daemon running it on the command line. Most options can also be specified in the /etc/tmd.conf configuration file. However, the command line options will override any configuration file options.

The following options can be used to configure the tape monitor daemon running it on the command line. Most options can also be specified in the /etc/tmd.conf configuration file. However, the command line options will override any configuration file options.

```
-s   Stop any currently running instance of the tape monitor daemon.
-r   Restart the tape monitor daemon and reload all configuration settings.
-d   Turn on drive error diagnostic retrieval and storage.
     This option is enabled by default.
-D   Turn off drive error diagnostic retrieval and storage.
-p   <directory>  Specify an alternate directory for the storage of
     drive diagnostic information. Default directory is /var/opt/IBMtape
-l   <filename>  Specify a file for writing daemon related log messages.
     By default, the tmd only writes status information to the syslog file of
     /var/adm/messages.
-y   Turns off writing log messages to syslog.
-z   Turn off compression.  By default, the tmd will use a form of file compression
     to reduce the size of stored diagnostic information.
```

The file name of dump presents some useful information. An example of the dump file is:

```
IBMtape.000001300148.2004-04-09-14:54:14.dump.gz
```

Here, 000001300148 represents the serial number of the tape device, 2004-04-09-14:54:14 is the time stamp for the dump retrieval.

A message is also logged in the syslog file of /var/adm/messages after a drive dump is retrieved by tmd. For example,

```
Apr  9 14:54:21 Java tmd[3279]: Drive dump saved to /var/opt/IBMtape
 IBMtape.000001300148.2004-04-09-14:54:14.dump.
```

Tracing Facility

IBMtape incorporates a tracing facility that is useful for performing problem determination. The tracing facility logs diagnostic information to */var/adm/messages* based on the control variable IBM_trace. Refer to "Setting the IBM_trace Level for Static Tracing" on page 188 for instructions on how to set the trace value.

IBM_trace values range from 0–13 and result in posted messages as shown in Table 29. Postings are cumulative, so trace level 3 also posts items for levels 2, 1, and 0. A trace value of 2 or 3 is suitable for most normal production environments, with little or no degradation of throughput. IBM_trace values of 4 and higher increasingly degrade performance and should generally be used only when directed by IBM support personnel.

Table 29. Tracing Facility

Trace Level	Items Traced
0	Severe error conditions only. For installations with extremely small */var* file systems, this setting can prevent filling the file system unexpectedly. However, this may be at the cost of not recording messages related to serious device or system environment errors.
1	Device sense data. Sense data can help in diagnosing the source of unexpected error conditions.
3	Device opens and closes. Decoded SCSI command, sense key, ASC and ASCQ for sense data.
4–13	Increasingly verbose tracing information. These tracing levels are generally useful only to IBMtape developers.
Note: IBMtape earlier than Version 4.0.2.7 had only IBM_trace values 0–4. Message content and selection differed significantly from current IBMtape versions.	

By default, system error messages, including IBMtape trace messages, are placed in */var/adm/messages*. If your installation has modified */etc/syslog.conf* to redirect system error messages, IBMtape tracing is handled as other kernel messages. Refer to the *syslog.conf* man page and the comments in *syslog.conf* for information about the system logging operation. Changes made to *syslog.conf* take effect after the next system restart.

The following shows trace level 2 output, with system date and time stamps removed. Device instance 390 is opened on the first line. The device minor number 12450 is decoded and shows that the SCSI medium changer (*smc*) special file was opened.

The second line decodes selected fields from the sense data that follows it. The decoded information shows that sense data was generated during a Move Medium command. Looking up the decoded Sense Key /ASC/ASCQ combination in the 3590 hardware reference, we find that the command failed because the move *from* location was empty. The actual sense data follows the decoded fields.

Note: Solaris, rather than printing multiple sixteen byte lines of hex zeroes, prints only the first such line, followed by a repeat count.

```
IBMtape(390) _open: 374 Inst 390, Minor 12450 (smc), Flags 0x5,
                TL 2/0/0, 4.0.2.8
IBMtape(390) check_sense: cmd 0xa5(move_medium), key/asc/ascq 0x5/3b/e,
              defer 0, retry 0, rc 22
IBMtape(390) 03590B11 SENSE DATA:
IBMtape(390)  70  0  5  0  0  0  0 58  0  0  0  0 3b  e ff  2
IBMtape(390)   0 20  1 40  a  9  1  0  0  0  0  0  0  0 a5  0
IBMtape(390)   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
last message repeated 1 time
IBMtape(390)   0  0  0  0  0  0  0  0 36 33 39 20 20 20 20  0
IBMtape(390)   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
IBMtape(390) _close: Inst 390, Minor 12450 (smc), Flags 0x5, exit(0)
```

In the next example, the device open line shows that a tape drive (*drv*) device special file was opened. The sense data for device instance 292 was generated during a space operation. The Sense Key/ASC/ASCQ shows that a filemark was encountered during the space.

```
IBMtape(292) _open: 554 Inst 292, Minor 9412 (drv), Flags 0x5,
                TL 2/0/0, 4.0.2.8
IBMtape(292) check_sense: cmd 0x11(space), key/asc/ascq 0x0/0/1,
                defer 0, retry 0, rc 5
IBMtape(292) 03570B02 SENSE DATA:
IBMtape(292)  f0  0 80  0  0  0  1 48  0  0  0  0  0  1 ff  a
IBMtape(292)  c4 b1  0 20  0  5  1 91  0 34  0  0  0  0 11  0
IBMtape(292)   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  6
IBMtape(292)  6f 28  0 ad 73 32  0  0  0  0  0  0  0  0  0  0
IBMtape(292)   0  0  0  0  0  0 20  0 31 42 41 20 20 20 20  0
IBMtape(292) _close: Inst 292, Minor 9412 (drv), Flags 0x5, exit(0)
```

Finally, the sense data for device instance 230, a tape drive, occurred during a test unit ready and indicates that no tape is loaded in the drive.

```
IBMtape(230) _open: 728 Inst 230, Minor 7366 (drv), Flags 0x5,
                TL 2/0/0, 4.0.2.8
IBMtape(230) check_sense: cmd 0x0(test_unit_ready),
                key/asc/ascq 0x2/3a/0, defer 0, retry 0, rc 5
IBMtape(230) 03570B02 SENSE DATA:
IBMtape(230)  70  0  2  0  0  0  0 48  0  0  0  0 3a  0 ff  2
IBMtape(230)  c4  8  0 30  0  6  1 40  0  0  0  0  0  0  0  0
IBMtape(230)   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
last message repeated 1 time
IBMtape(230)   0  0  0  0  0  0  0  0 31 42 41 20 20 20 20  0
IBMtape(230) _close: Inst 230, Minor 7366 (drv), Flags 0x5, exit(0)
```

You can match an instance number with its corresponding device special file in two steps.

1. Find the instance number in */etc/path_to_inst*:

```
$ grep 292 /etc/path_to_inst
"/pci@6,4000/scsi@2,1/IBMtape@2,0" 292 "IBMtape"
```

2. List *long* the contents of */dev/rmt* and search for the path name you found in the previous step:

```
$ ls -l /dev/rmt | grep "/pci@6,4000/scsi@2,1/IBMtape@2,0"
lrwxrwxrwx   1 root     other        48 Aug  26 11:49 8st ->
   ../../devices/pci@6,4000/scsi@2,1/IBMtape@2,0:st
lrwxrwxrwx   1 root     other        49 Aug  26 11:49 8stb ->
   ../../devices/pci@6,4000/scsi@2,1/IBMtape@2,0:stb
```

In this example, */dev/rmt/8st*, */dev/rmt/8stb*, and so on, are symbolic links to the device special files that are associated with device instance 292.

Dynamic Tracing Utility

A dynamic tracing utility named tapedtrc is introduced in the IBMtape.4.1.6.0 or later driver to dynamically set, reset, start, stop and query IBMtape tracing at any time for debugging use. The program is located in the */opt/IBMtape* directory, with the tracing level set to 0 by default.

Use the tapedtrc program from the command line as follows:

```
/opt/IBMtape/tapedtrc  [option]
    options:
        [set]             - Set IBMtape trace level and/or start the tracing
        [set] level       - Set trace to a particular trace level
        [get]             - Query the current IBMtape trace level
        [start]           - Start IBMtape tracing
        [stop]            - Stop IBMtape tracing without the trace level reset
        [clean]           - Stop the IBMtape tracing and reset IBMtape trace
                            level to 0
        [help]            - IBM tapedtrc help menu
```

Setting the IBM_trace Level for Static Tracing

The user can still enable or disable static IBMtape tracing and set the IBM trace level in */etc/system* or by running the **adb** system command. The host is required to reboot to enable or disable the tracing when the trace level is set in */etc/system*. The IBMtape driver must be loaded in the kernel. The tracing will start or stop at the next device open, if the tracing is enabled or disabled using the **adb** command.

The default value for IBM_trace is zero (0). You can define another IBM_trace value by placing an entry in */etc/system*, so that IBM_trace is set at each restart. For example, this entry in */etc/system* sets IBM_trace to 2 at each restart:

```
set IBMtape:IBM_trace = 2
```

When IBM_trace is set in */etc/system*, it affects tracing during driver loading, initialization, and operation.

You may also set or modify the IBM_trace value manually in an *adb* session. Because the driver must already be loaded and initialized before using this method, the trace value that is set is active only during driver operation.

In this sample session, *ksh>* is a shell prompt, and *adb>* is the *adb* session prompt. Commands that you enter are in **boldface**. Explanatory comments follow pound signs (#) or exclamation and pound sign pairs (!#). Text lines without a prefix are *adb* session responses to commands.

```
#
#  Start adb session and set session prompt.
ksh> adb -P "adb> " -k -w /dev/ksyms /dev/mem
physmem 7c5e
!#
!#  Set default for input values to base 10.
adb> a$d
```

```
radix=10 base ten
!#
!#  Display current IBM_tape value as unsigned decimal integer.
adb> IBM_trace/u
IBM_trace:
IBM_trace:      0
!#
!#  Set new IBM_trace value.
!#  adb will confirm the old and new values.
adb> IBM_trace/w 2
IBM_trace:      0       =       2
!#
!#  Quit session.
adb> $q
#
#  Back to the shell.
ksh>
```

Running Diags_info Script

Run the diags_info script located in the /opt/IBMtape directory. This script detects the problems on the configuration files, gathers important system HBAs and configuration information. The script should be run as root. If not run as root, the information should be labeled as such, but the value of the information is degraded when run as a non-root user.

To facilitate capture of data, the script places information in a file called diags.out in the directory locating the script. Send the output file to the location identified by your IBM service representative.

iostat Command

IBMtape driver supports the iostat system command, which reports I/O statistics for the supported tape drives in IBMtape.4.1.2.7 and later versions. Refer to man (1M) iostat for the command usage.

Chapter 7. Windows Tape and Medium Changer Device Driver

This chapter describes the hardware requirements, software requirements, and installation notes for the Microsoft Windows device drivers for IBM TotalStorage and System Storage tape devices.

Purpose

The Windows tape and medium changer device drier is designed specifically to take advantage of the features provided by the IBM tape drives and medium changer devices. The goal is to give applications access to the functions required for basic tape operations (such as backup and restore) and medium changer operations (such as mount and demount the cartridges), as well as to the advanced functions needed by full tape management systems. Whenever possible, the driver is designed to take advantage of the device features transparent to the application

Data Flow

The software described in this chapter covers the Windows device driver and the interface between the application and the tape device.

Figure 13 illustrates a typical data flow process.

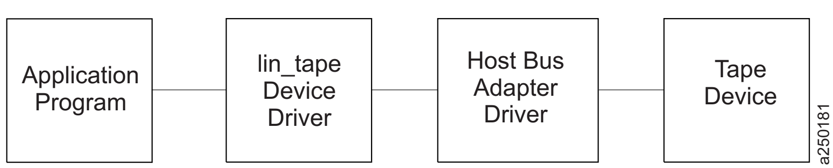


Figure 13. Data Flow for Windows Device Driver (IBMmag)

Product Requirements

The most current information on supported hardware and software configurations for the Windows tape and medium changer device driver can be found in the README files on the IBM Device Drivers FTP site, which is located at ftp://ftp.software.ibm.com/storage/devdrvr/Windows.

Hardware Requirements

One of more of the following processors is required by the IBMtape device driver:

- For the Windows NT® operating system, Intel compatible processors with a minimum processor level of Intel 486DX or Pentium®.
- For the Windows 2000, Windows Server 2003, and Windows Server 2008 operating systems, a 32-bit Intel compatible processor.
- For the Windows Server 2003 and the Windows Server 2008 operating systems, a 64-bit Itanium processor.
- For the Windows Server 2003 and the Windows Server 2008 operating systems, a processor that supports extended 64-bit architectures (Intel EM64T and AMD64).

One or more of the following SCSI host adapters:

- Adaptec 2940U2W or IBM P/N 33L5000 for IBM System x (also known as xSeries®) (LVD)
- Adaptec 2944UW (HVD)
- Adaptec SCSI Adapter 29160 or IBM P/N 19K4646 for IBM System x (also known as xSeries) - Single Port LVD
- Adaptec SCSI Adapter 39160 - Dual Port LVD
- Adaptec SCSI Card 39320-R (LVD) and 39320D-R (dual port LVD)
- Adaptec SCSI Card 29320ALP or IBM P/N 13N2249 (LVD)
- Symbios SYM22910 64-bit PCI-to-Ultra-2 SCSI Dual Channel Host Adapter (LVD) from LSI Logic Corporation
- IBM Ultra320 SCSI Controller 2 P/N 13N2249 (SCSI LVD)
- IBM Ultra320 SCSI Controller 2 P/N 39R8743 (SCSI LVD)

One or more of the following SAS host bus adapters:

- LSI Logic SAS3800X SAS Adapters
- IBM SAS Host Bus Adapter Controller Part Number 25R8060

NTD: Can we combine the new Fibre Channel host bus adapter items with the already existing FC-AL ones (see immediately below)?

One or more of the following Fibre Channel host bus adapters:

- IBM xSeries and Netfinity® FAStT Host Adapter P/N 19K1246
- IBM xSeries and Netfinity FAStT Host Adapter P/N 24P0960
- DS4000™ FC 4Gbps PCI-X Single Port HBA P/N 39M5894
- DS4000 FC 4Gbps PCI-X Dual Port HBA P/N 39M5895
- QLogic 4Gb FC Single-Port PCIe HBA P/N 39R6525
- QLogic 4Gb FC Dual-Port PCIe HBA P/N 39R6527
- Qlogic QLA2310
- Qlogic QLA2340
- Qlogic QLA2342

- Qlogic QLA2460
- Qlogic QLA2462
- Qlogic QLE2460 (not supported for Windows 2008 SPI for IA64 servers)
- Qlogic QLE2462 (not supported for Windows 2008 SPI for IA64 servers)

One or more of the following FC-AL host bus adapters:
- Qlogic QLE2460/2 4Gb PCI-E FC HBA, QLA2310FL, QLA2340, QLA2340L, QLA2342, QLA2342L, QLA2342L, QLA2460, QLA2462 Fibre Channel Adapters
- Emulex LPe11000/2 4Gb PCI-E FC HBA, LP1150, LP8000 and LP9002L, LP402DC, LP952L, LP9802, LP9802DC, LP10000, LP11000, LP11002 Fibre Channel Adapters
- IBM Netfinity P/N 00N6881, 19K1246, 24P0960, 39M5894, 39M5895

Attention: Using a single Fibre Channel host bus adapter (HBA) for concurrent tape and disk operations is not recommended. Tape and disk devices require incompatible HBA settings for reliable operation and optimal performance characteristics. Under stress conditions (high I/O rates for tape, disk, or both) where disk and tape subsystems share a common HBA, stability problems have been observed. These issues are resolved by separating disk and tape I/O streams onto separate HBAs and using SAN zoning to minimize contention. IBM is focused on assuring server and storage configuration interoperability. It strongly recommends that your implementation plan includes provisions for separating disk and tape workloads.

Software Requirements

The software requirements are supported:
- SCSi or FC adapter device driver
- Microsoft Windows NT Version 4.0 with Service Pack 6 or later
- Microsoft Windows 2000 Build 2195 or later, or
- Microsoft Windows Server 2003 Build 3790 or later
- Microsoft Windows Server 2008 SP1 Build 6001 or later

Installation and Configuration Instructions

This section includes instructions for installing and configuring the Windows tape and medium changer device driver on Windows NT and Windows 2000/Windows Server 2003/Windows Server 2008.

Windows NT Instructions

This section describes how to install, remove, start, and stop the Windows tape and medium changer device driver on Windows NT.

Installation Overview

The installation process consists of the following steps:
1. Verify that the prerequisites have been satisfied.
2. Install the SCSI or FC adapter.
3. Install the SCSI or FC adapter device driver.
4. Connect the TotalStorage/Magstar or Ultrium device to the adapter.
5. Power On the TotalStorage/Magstar or Ultrium device.
6. Reboot the system.

7. Create an emergency repair disk (optional) using the following procedure:
   a. Select **Start**, **Help**, and **Find** from the Windows NT desktop.
   b. Type **Emergency** in box 1.
   c. Select **Using the Repair Disk utility to make an emergency repair disk**.
   d. Click **Display**, then follow the instructions to make an emergency repair disk (ERD).

      Note: Refer to “Software Requirements” on page 193.

      This step is highly recommended.
8. Install the TotalStorage/Magstar or Ultrium tape device driver.

Installation Procedure

To install the device drivers, follow this procedure:

1. Log on as Administrator.
2. Download the appropriate driver from the IBM Device Drivers FTP site, which is located at ftp://ftp.software.ibm.com/storage/devdrvr/Windows/WinNT.
   - For TotalStorage or Magstar tape devices, select **IBMMag.WinNT.exe**
   - For Ultrium tape devices, select **IBMUltrium.WinNT.exe**
3. Go to the directory in which you saved the .exe file and double click it.
4. Follow the *InstallShield* directions to install the package.
   a. If you select **compact** installation, the program copies the system files of the latest IBM TotalStorage/Magstar or Ultrium Device Driver (**Device Driver System Files** component) to the system directories. This provides the latest support necessary for TotalStorage or Magstar devices.

      Note: This option overwrites previous versions of the TotalStorage/Magstar or Ultrium device driver installed on your system.
   b. If you select **typical** installation, all support included in the **compact** installation is done. Also, the **Device Driver Depot** component is installed, which includes copying all available versions included in the InstallShield package to a user defined directory.

      Note: This option overwrites any previous versions of the TotalStorage/Magstar or Ultrium device driver installed on your system.
   c. If you select **custom** installation, the program allows you to select the components to install. Refer toFigure 14 on page 195.

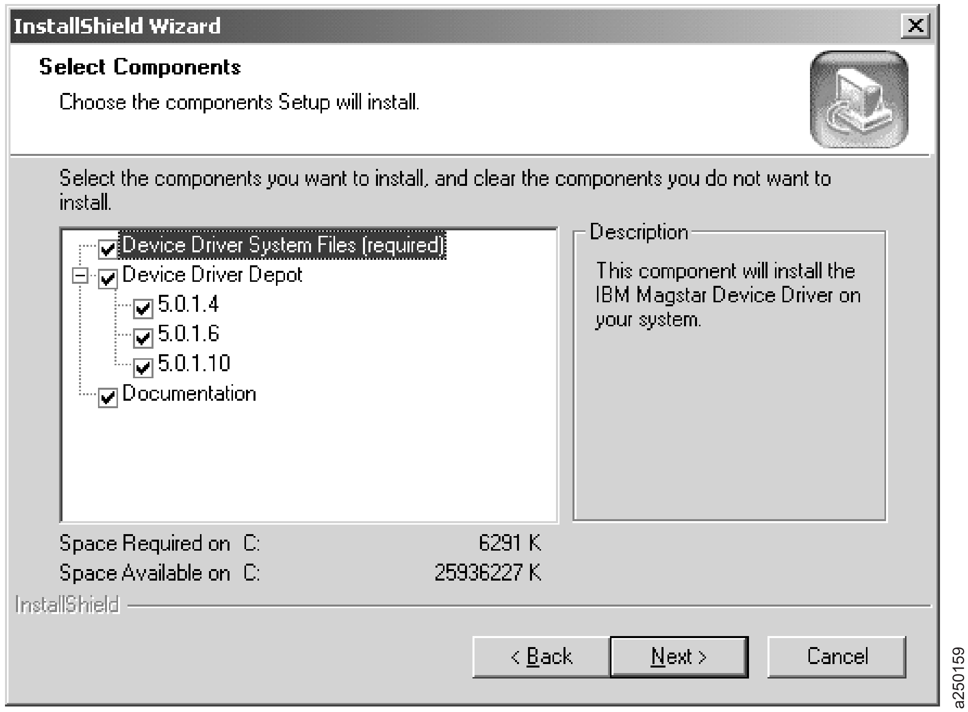


Figure 14. Select Components Menu (using IBMUltrium.WinNT.exe)

The **Device Driver Depot** component includes the most recent versions of the device drivers. If you install this component, you are able to select which versions of the device drivers to install.

The **Documentation** component copies the PDF version of the *Installation and User's Guide* and the *Programming Reference* to your hard drive.

The **Device Driver System Files** component copies all the files needed for device driver support to the system directory.

5. After you have completed installing the device driver, enable the driver. Click **Start**, move to **Settings**, then click **Control Panel**. Refer to Figure 15 on page 196.

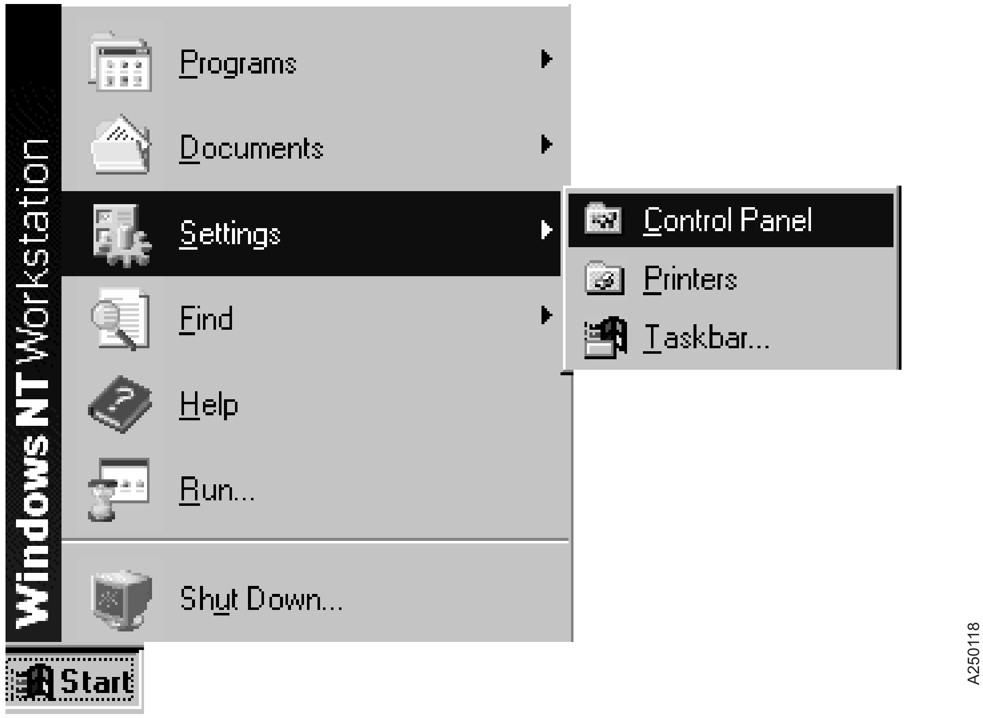


Figure 15. Start Menu

6. Double-click **Tape Devices**. If the TotalStorage/Magstar or Ultrium tape or changer device was already powered On and attached to the system during boot up, the devices should appear in the box, and Windows NT should start to generate the driver list. Refer to Figure 16.

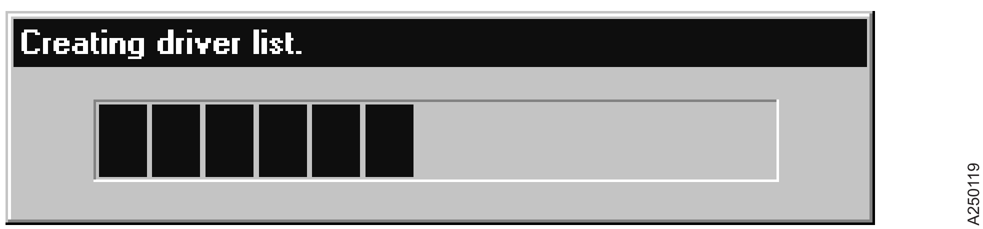


Figure 16. Creating Driver List

If Microsoft Windows NT did not detect the attached TotalStorage or Magstar device, click **Detect** to select the device, and Microsoft Windows NT rescans the bus. Refer to Figure 17.

Figure 17. Rescanning for Tape Device

If you still cannot see the device, ensure that the cable is attached properly. Also, ensure that the device is terminated properly, the device is powered On, and the adapter driver is enabled.

If you have more than one IBM TotalStorage/Magstar or Ultrium device attached on the same host system, the operating system will prompt you multiple times for the same driver. **Cancel** the multiple driver installation requests.

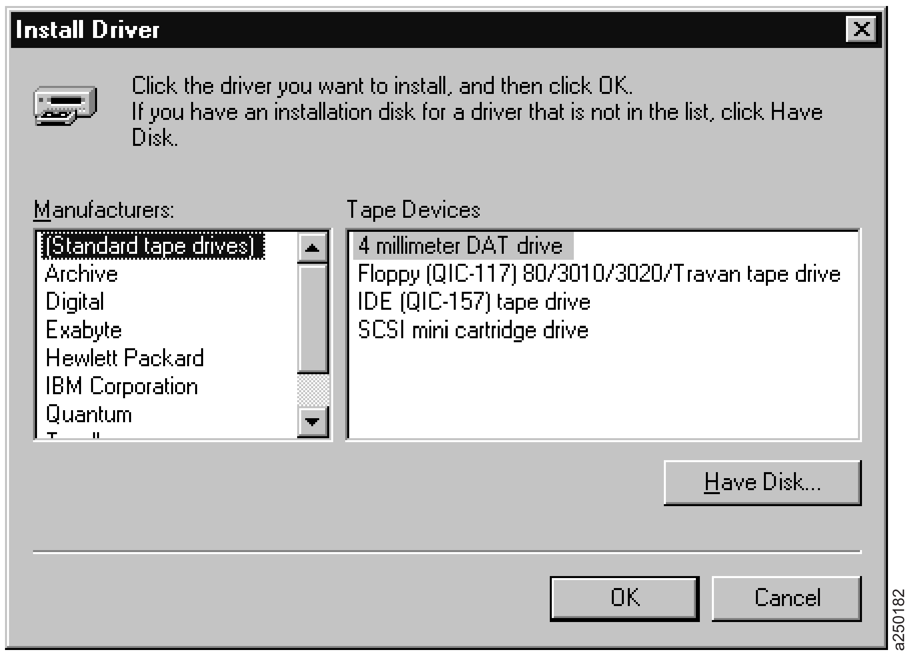


Figure 18. Install Driver Menu - Select **Cancel**.

Click **Drivers**, then click **Add...**. Refer to Figure 19 on page 198.

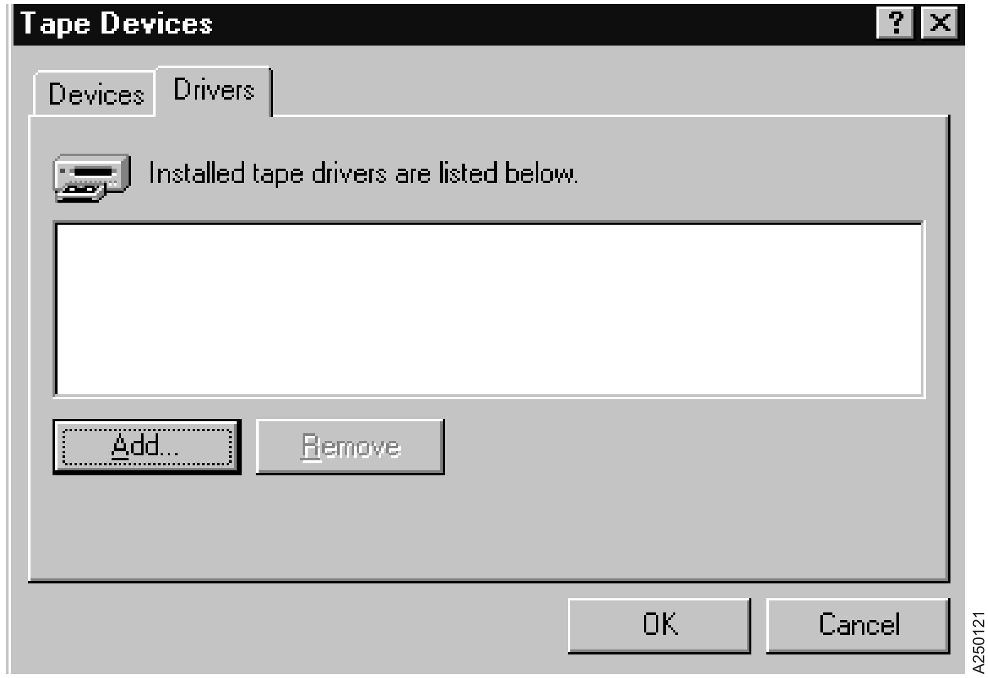


Figure 19. Tape Devices Menu

7. If you installed the **Device Driver System Files** component in step 4 on page 194, skip to step 8 on page 199. If you did not install **Device Driver System Files** component, click **Have Disk**, then enter the directory where your device driver setup file (*IBMUltrium.inf* or *IBMMagstar.inf*) is located in the box labeled **Copy manufacturer's files from:**. You can also **Browse...** to select the directory. Refer to Figure 20 on page 199.

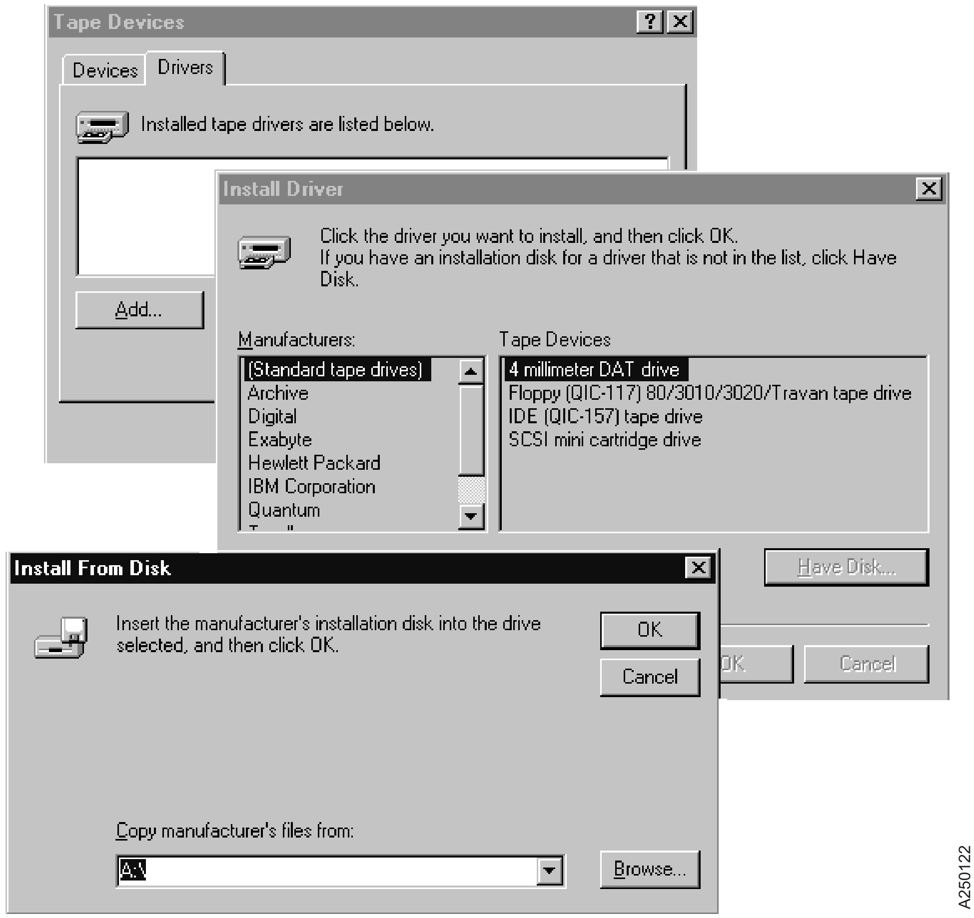


Figure 20. Have Disk Menu

8. Click **IBM Corporation**, select the device driver that matches your device, then click **OK**. Refer to Figure 21 on page 200.

   Note: If you are using both a tape drive and a medium changer (as you would find in a tape library, for example), select one of the **IBM Tape Libraries**. This will install drivers for both the medium changer and the tape drives.

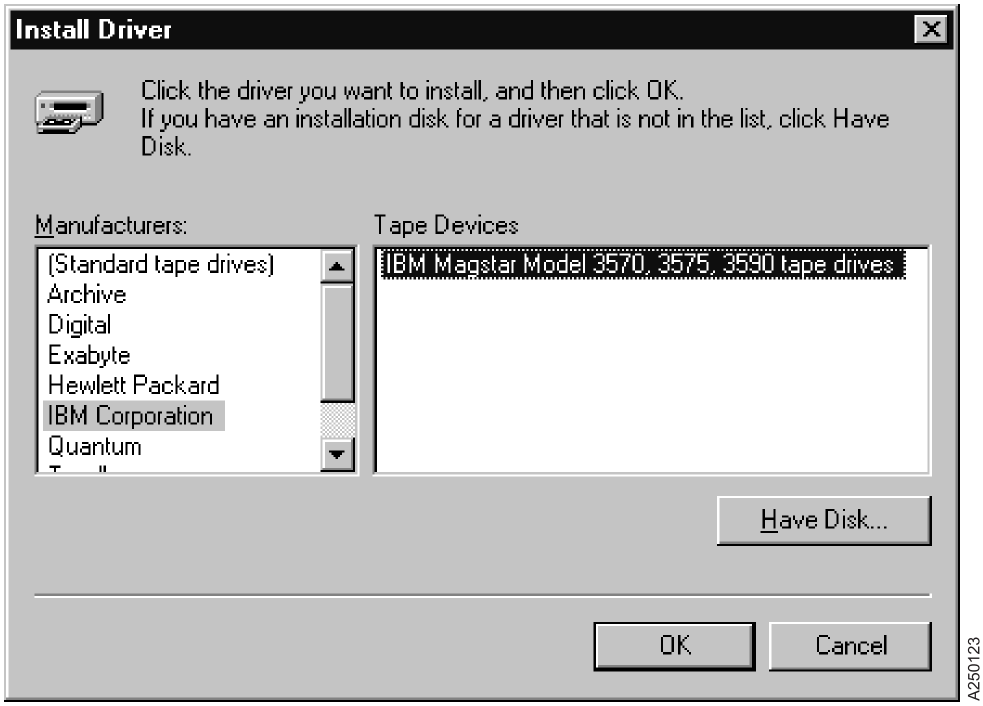


Figure 21. Install Driver Menu

9. If you installed the **Device Driver System Files** component in step 4 on page 194, Windows NT might ask the question shown in Figure 22. Click **Yes**, then skip to step 11. If your system prompts you for *ibmtape.sys* or *Magstar.sys*, browse to select the drivers directory (*c:\winnt\system32\drivers*). The system may also prompt you for *ntutil.exe*. Browse to the system32 directory (*c:\winnt\system32*).

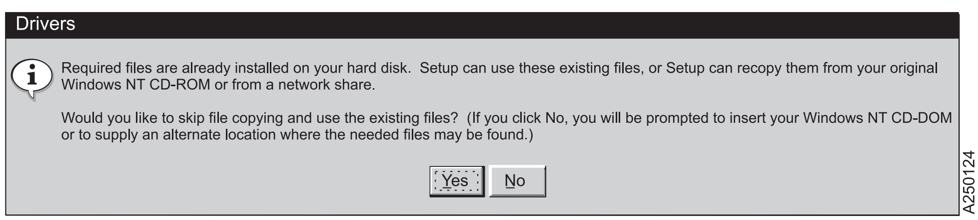


Figure 22. Windows NT Statement

10. If you **did not** install the **Device Driver System Files** component in step 4 on page 194, click **No** to the question in step 9. The operating system prompts you for the location of the driver files.
11. Reboot the operating system to start the drivers.

Removal Procedure

The Windows NT tape and medium changer device drivers manage all TotalStorage/Magstar or Ultrium devices and changers exclusively. If you use applications that have their own device drivers that access TotalStorage/Magstar, or Ultrium devices and changers, you must remove the Windows NT tape and medium changer device drivers before installing and configuring those other applications.

1. Log on as Administrator.
2. Click **Start**, move to **Settings**, then click **Control Panel**.

3. Double-click **Tape Devices**.
4. Click the **Drivers** tab.
5. Highlight the IBM tape device, then click **Remove**. When you see the message *Are you sure you want to remove this driver?*, click **Yes**. Refer to Figure 23.

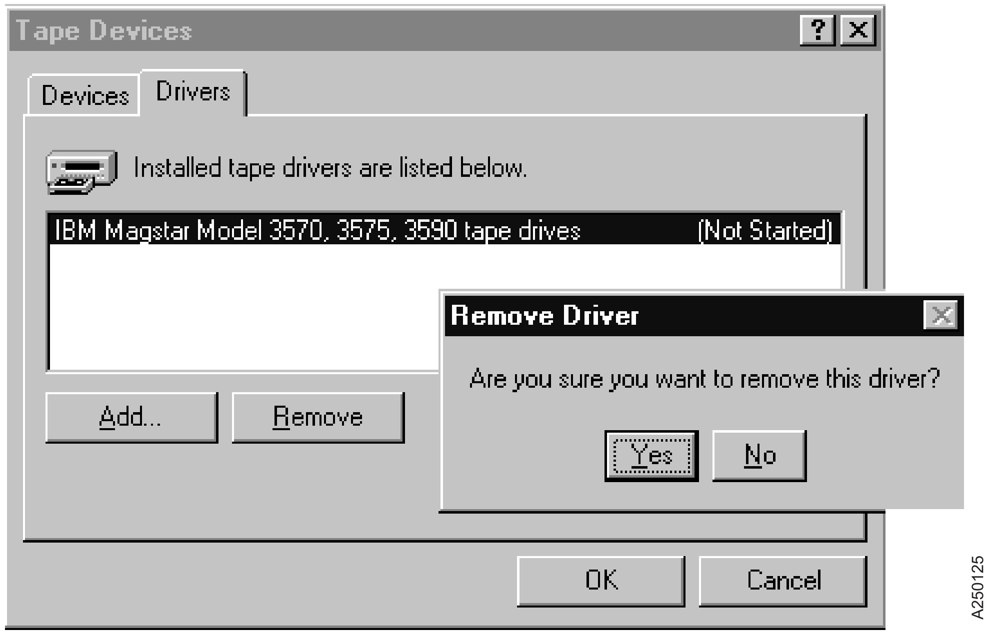


Figure 23. Tape Devices and Remove Driver Menu

6. Click **Add/Remove Programs** in the **Control Panel** window. Refer to Figure 24.

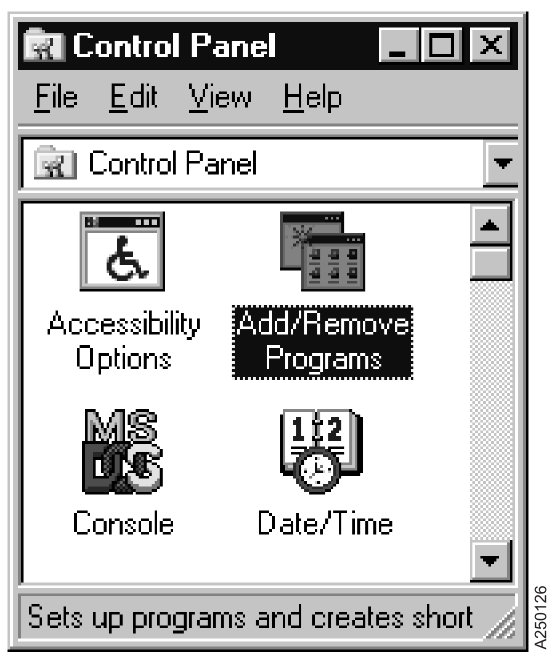


Figure 24. Control Panel Selection

7. Highlight the IBM tape device driver, then click **Add/Remove...**. Refer to Figure 25.

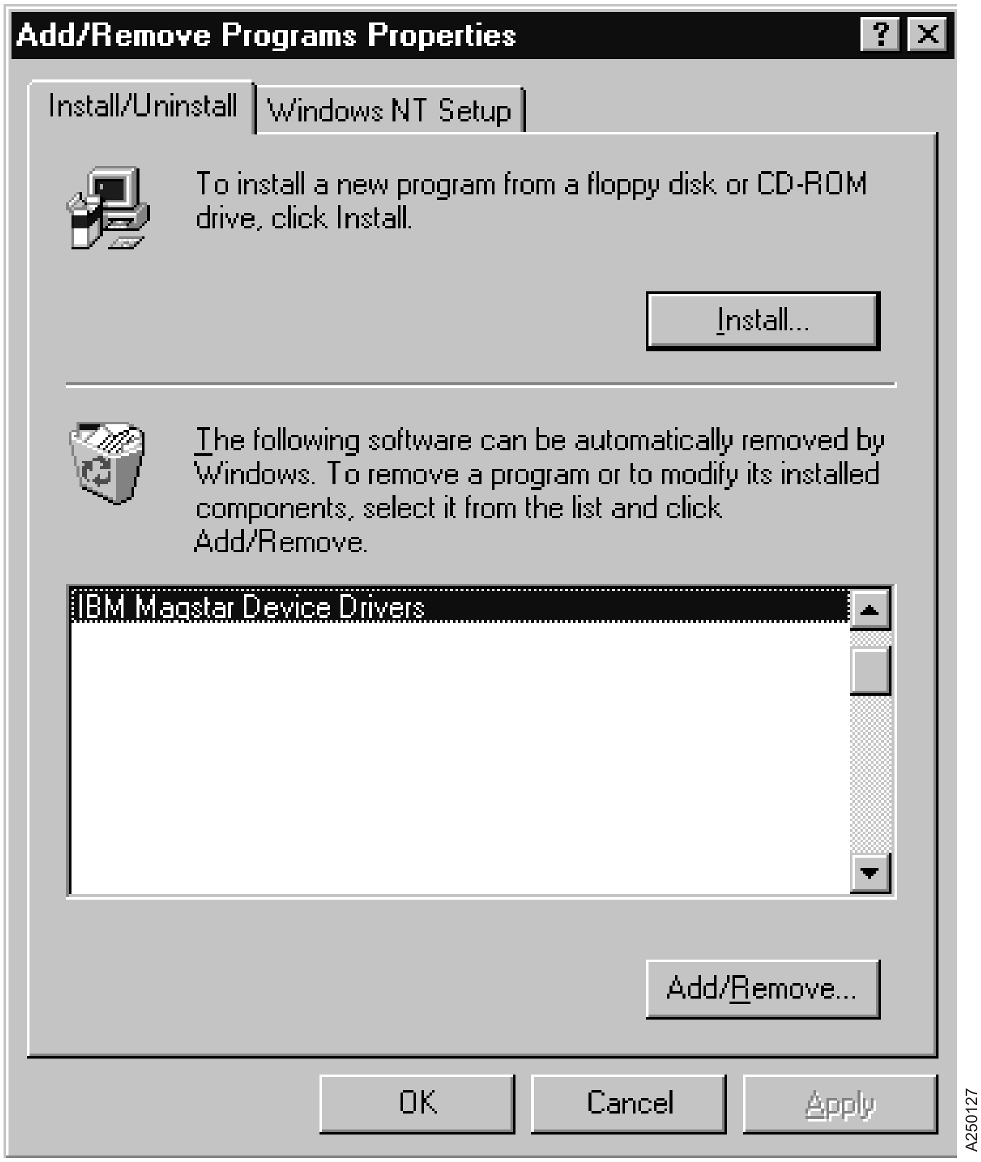
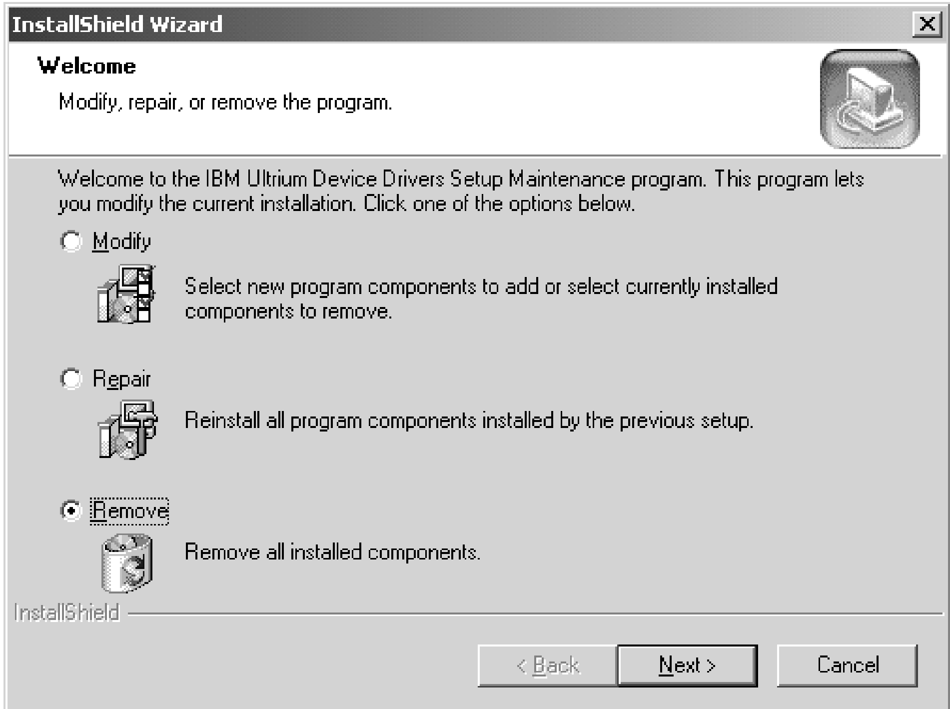


Figure 25. Add/Remove Programs Properties

8. Select the **Remove** option, then follow the *InstallShield Wizard* to uninstall the drivers.



a250128

Figure 26. Drive Removal Menu.

Note: Uninstalling the drivers removes all TotalStorage/Magstar or Ultrium device driver files, registry settings, and other components, such as the *Installation and User's Guide* and Device Driver Depot.

9. Click **Next >**.
10. Shut down and reboot the system.

Manual Starting and Stopping Procedures

The TotalStorage/Magstar or Ultrium device and medium changer drivers support being stopped and started without a reboot. If the TotalStorage/Magstar or Ultrium device and medium changer drivers are used in conjunction with other drivers that support being stopped and started without a reboot, you are able to switch between device drivers without rebooting the system.

To control manually when the TotalStorage/Magstar or Ultrium device drivers start or stop, set the startup mode to **Manual**, then start the device driver manually when required.

To set the startup mode to **Manual**:

1. Log on as Administrator.
2. Click **Start**, move to **Settings**, then click **Control Panel**.
3. Double-click **Device**.
4. Scroll down to find the IBM tape device and medium changer drivers.
5. Select the driver, click **Startup...**, select **Manual**, then click **OK**. Refer to Figure 27 on page 204.

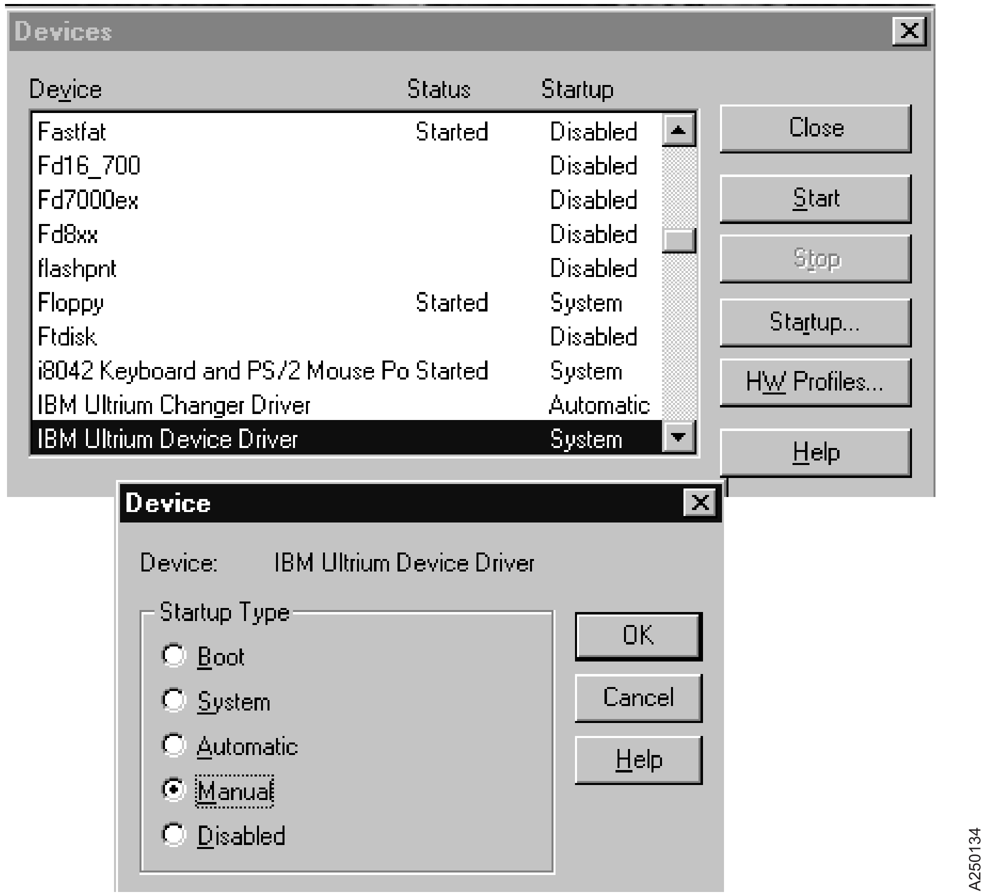


Figure 27. Manual Starting and Stopping Menu

6. If required for other drivers, repeat steps 4 on page 203 and 5 on page 203 of this procedure for each driver.

To start a driver manually:

1. Ensure that the other drivers that may conflict are not started.
2. Perform steps 1 through 4 of the procedure outlined above.
3. Select the driver, then click **Start**.
4. Repeat for the other driver, if necessary.

To stop a driver manually:

1. Perform steps 1 through 4 of the procedure outlined above.
2. Select the driver, then click **Stop**.

Windows 2000, Windows Server 2003, and Windows Server 2008 Instructions

This section describes how to install, remove, and uninstall the Windows tape and medium changer device drivers on Windows 2000, Windows Server 2003, and Windows Server 2008.

Note: The last driver level to include support for Windows 2000 is V6.1.4.8. Subsequent levels include support for Windows Server 2003. Windows Server 2008 is supported beginning at V6.1.9.5.

Installation Overview

The installation process consists of the following steps:

1. Verify that the hardware and software requirements have been met.
2. Install the host bus adapters and drivers.
3. Shut down the system.
4. Connect the tape and medium changer devices to the host bus adapters.
5. Power on the tape and medium changer devices.
6. Set the tape and medium changer device addresses.
7. Reboot the system.
8. Log on as Administrator.
9. Install and configure the devices and device drivers using the installation application.

All drives accessible from a medium changer must be on the same physical SCSI bus as the changer.

Installation Procedures

These procedures make the following assumptions:

- No other driver is installed that claims the tape and medium changer devices.
- If you are updating the device driver from a Microsoft certified version to an uncertified version, it is recommended that you first uninstall the certified driver by referring to the uninstall procedures in this documentation.
- The host bus adapter is installed, configured properly, and is running supported microcode and driver levels.
- Drivers are identified by the following conventions, where *nnnn* refers to a version of the driver. If there is more than one version, use the latest.
  - Windows 2000 and Windows Server 2003, 32-bit

    `IBMTape.x86_nnnn.zip`
  - Windows Server 2008, 32-bit

    `IBMTape.x86_w08_nnnn.zip`
  - Windows Server 2003 for Itanium (IA64) 64-bit architecture

    `IBMTape.i64_nnnn.zip`
  - Windows Server 2008 for Itanium (IA64) 64-bit architecture

    `IBMTape.i64_w08_nnnn.zip`
  - Windows Server 2003 for extended 64-bit architectures (Intel EM64T and AMD64)

    `IBMTape.x64_nnnn.zip`
  - Windows Server 2008 for extended 64-bit architectures (Intel EM64T and AMD64)

    `IBMTape.x64_w08_nnnn.zip`

To install the device drivers, follow this procedure:

1. Log on as Administrator.
2. Download the appropriate driver from a subdirectory of:

   `ftp://ftp.software.ibm.com/storage/devdrvr/Windows`

   Drivers for Windows 2000 are in the Win2000 subdirectory. Drivers for Windows Server 2003 are in the Win2003 subdirectory. Drivers for Windows

Server 2008 are in the Win2008 subdirectory. When you download the files, ensure that you use FTP ″binary″ mode.

3. Unzip the driver package to a hard drive directory of your choice, other than the root directory.
4. Ensure that the tape and medium changer devices are connected to your host bus adapter and configured properly by locating the devices in Device Manager.
5. For driver packages prior to v6.1.8.9, double-click `install.exe` in the driver package.

   For all subsequent levels, double-click either `install_exclusive.exe` or `install_nonexclusive.exe`.

   - With `install_exclusive.exe`, the driver will issue automatic reserves on open and also prevent multiple open handles from the host to a drive from existing at the same time, as is required by applications such as Tivoli Storage Manager.
   - With `install_nonexclusive.exe`, the driver will permit open handles from the host to a drive to exist at the same time, as is required by applications such as Microsoft Removable Storage Manager (RSM).

   The necessary .sys files for correct driver operation will be installed, and all IBM devices will be associated with the driver. Refer to Figure 28.

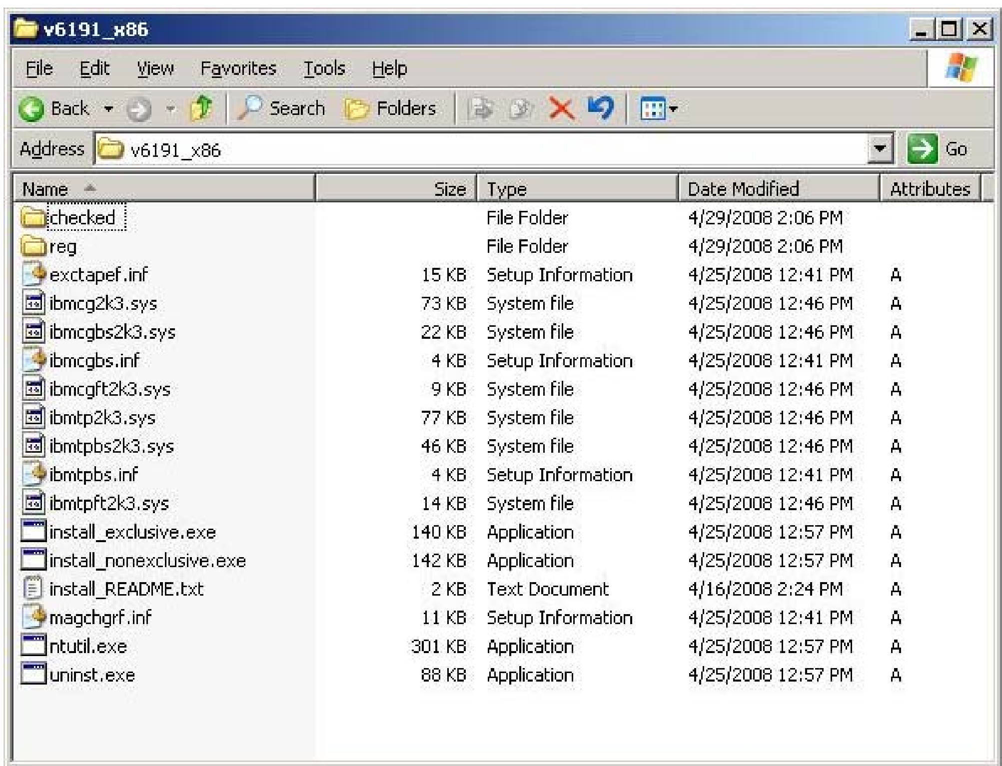


Figure 28. Installation Application in Windows Explorer

Notes:

a. There are additional installation features available through the command line interface (CLI), which include the following:
   - Installing only the tape or medium changer device drivers (-t or -c)

- Running in debug mode, which will create the file *debug.txt* in the driver package directory (-d)
- Running in silent mode, which will suppress pop-up messages requiring user intervention, but only with Microsoft-certified IBM drivers (-s)
- Disabling the Microsoft RSM service (-r), available in driver packages v6.1.8.6 and later

To install the device drivers using any of these features, instead of double clicking the install executable file, open a command-line window and cd to the driver package directory. For the usage information, type *install_exclusive.exe -h* or *install_nonexclusive.exe -h* at the prompt.

b. If the Windows "Found New Hardware" Wizard begins during installation, cancel the wizard. The install application will perform the necessary steps.

6. If you are installing a driver that has not been certified by the Microsoft Windows Hardware Quality Laboratories (WHQL), you will be presented with a warning screen. Refer to Figure 29. If you want to continue installing the driver, select **Continue Anyway**.

Note: All drivers released by IBM have been through a complete test to ensure that they are stable and conform to specified requirements.

Figure 29. Windows Logo Testing screen

7. If you are installing a Windows Server 2008 driver that has not been certified by the Microsoft Windows Hardware Quality Laboratories (WHQL), it will likely have a Verisign digital signature. During installation, you may be presented with a prompt to install the software. Mark the ″Always trust software from IBM Corporation″ check box and click **Install**. You should only see this screen the first time that you install the drivers, provided you click the Always trust software box.
8. To verify that the tape and medium changer devices and drivers are installed correctly, follow the instructions in Appendix B, “Verifying Proper Attachment of Your Devices,” on page 307.

Device Removal or Disable Procedure

If you need to remove a device, or if you are altering the hardware configuration, you should uninstall or disable the device first.

1. Right-click **My Computer**, select Manage to open the Computer Management Console, and click **Device Manager**.
2. Right-click the TotalStorage or Magstar device you wish to uninstall and select **Uninstall ...**. If you wish to disable the device without uninstalling it, you may select **Disable**.
3. You will be prompted to confirm the uninstallation. Click **OK**.
4. In Device Manager, under **System devices**, right click **Changer Bus Enumerator** and select **Uninstall**.
5. In Device Manager, under **System devices**, right click **Tape Bus Enumerator** and select **Uninstall**.

Note: This removal procedure will remove the device from the device tree, but it will not uninstall the device driver files from your hard disk.

Uninstalling the Device Drivers

To uninstall the device drivers from the system, which includes deleting the system files and deallocating other system resources, complete the following:

1. Complete the steps under Device Removal or Disable Procedure to remove the tape and medium changer devices.
2. Double-click *uninst.exe* in the driver package.

   Note: This removes all the files in the system directories that were created during the installation of the device driver. It does not delete the .zip file or the files that were extracted from the .zip file. If you desire to remove these files, you will need to delete them manually.
3. Reboot the system.

Persistent Naming Support on Windows Server 2003 and Windows Server 2008

The Windows tape driver has an option for enabling device object names that will persist across reboots of the operating system. For example, if your tape drive has the name `\.\tape4801101` and the persistent naming option is used, then `\\.\tape4801101` will be reserved for use by that device after an operating system reboot.

Perform the following steps to enable this feature:

1. Add a DWORD value to the registry called PersistentNaming and assign it a value 1 at:

   ```
   HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\ibmtp2kx
   ```
2. Reboot your system. On reboot, the system writes information to the registry to associate the World-Wide Node Name from Inquiry p. 0x83 with the persistent name used by the operating system.
   - If the World-Wide Node Name is unavailable, or if the drive is a virtual (that is, emulated) drive, then the device serial number will be used rather than the World-Wide Node Name.
   - If the PersistentNaming option is not specified in the registry, then there is no guarantee that your devices will claim the same device name after reboot or driver initialization.

You can find registry subkeys with persistent naming information at

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\ibmtpbs2kx
```

Alternately, you can use the Windows Device Manager to examine the device number in order to determine that persistent naming is enabled on your host. Persistent names will contain tape device numbers based at 4801101 (which is the decimal equivalent of hexadecimal 0x49424D and ASCII "IBM").

If two physical paths exist to a drive and different Windows device names are required (which happens, for example, when two different HBAs are connected to the drive and Data Path Failover is disabled) the first path discovered will claim the persistent device name, and any subsequent paths that connect to the same device will receive names according to the order in which they are discovered by the Windows Device Manager.

Control Path Failover Support for Tape Libraries

To take advantage of Windows Control Path Failover (CPF) support, the appropriate feature code must be installed. Refer to “Supported Devices and Feature Codes” on page 14 for what feature code may be required for your machine type.

Configuring and Unconfiguring Control Path Failover Support

Control Path Failover support is enabled automatically when the device driver is installed. It may be disabled or reenabled for the entire set of attached medium changers by modifying the registry.

1. Open the **reg** folder of the driver package.
2. Double click *DisableCPF.reg* or *EnableCPF.reg*.
3. Reboot the system. This is necessary for any registry modification to take effect.

Querying Primary and Alternate Path Configuration

To check if the control path failover has been enabled in the device driver and display the primary and alternate paths, you may use the Windows utility program *ntutil*. Open a tape library from *ntutil* library mode, choose option “get changer bus info”. If you have multiple paths configured for Changer0, you will see it lists multiple Changer0s. Each one is associated with a different bus address. One of them is marked as “Primary”. The other Changer0s that are not marked as Primary are alternate paths for Changer0

The following is an example of the output of “get changer bus info” option from ntutil. Changer0 and Changer1 have failover enabled, and Changer2 does not have failover enabled. There are one path for Changer1 and two paths for Changer0. For Changer0, the path with SCSI address 03.01.00.01 is the primary path, and the path with SCSI address 03.01.03.01 is an alternate path.

```
  DEVICE NAME       SERIAL       FAILOVER  SCSI ADDRESS  ACTIVE  PRIMARY
-------------------------------------------------------------------------
   Changer1    0000000T00390401    YES     03.01.04.01     YES   PRIMARY
   Changer0    0000000T003904E5    YES     03.01.00.01     YES   PRIMARY
   Changer0    0000000T003904E5    YES     03.01.03.01     YES
   Changer2    0000000T00361234     NO     03.01.02.01     YES   PRIMARY
```

Checking Disablement of Control Path Failover Setting

If you have disabled the control path failover in device driver’s setting by double clicking the *DisableCPF.reg* file and reboot your system, you may go into the registry by issuing the Windows *regedit* command to confirm that CPF has been disabled. Look for a line like the following if your system is Windows Server 2003 or Windows Server 2008:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ibmcg2kx]
"FailoverDisabled"=dword:00000001
```

This indicates that CPF has been disabled in the driver. This setting only takes effect after your system is rebooted.

Data Path Failover Support for Tape Drives

To take advantage of Windows Data Path Failover (DPF) support, the appropriate feature code must be installed. Refer to “Supported Devices and Feature Codes” on page 14 for what feature code may be required for your machine type.

Configuring and Unconfiguring Data Path Failover Support

Data Path Failover support is enabled automatically when the device driver is installed. It may be disabled or reenabled for the entire set of attached drives or medium changers by modifying the registry.

1. Open the **reg** folder of the driver package.
2. Double click *DisableDPF.reg* or *EnableDPF.reg*.
3. Reboot the system. This is necessary for any registry modification to take effect.

For LTO generation 3 or lower, for tape drives that require a data path license key on the host side to enable DPF, the device driver looks for a file called %system_root%:\IBM_DPF.txt for the key, where %system_root% is the drive letter where Windows has been installed, typically c, (for example, C:\IBM_DPF.txt). The file should contain the key on a single line, with no spaces and no other text on the line. If multiple keys are required, place each key in the file on its own line. The driver looks for this file at initialization, and if the file contains a valid DPF license key, the DPF feature is enabled and any eligible devices will have multi-path support.

Note: For LTO generation 3 running the latest drive microcode, there is no longer a requirement to provide a DPF license key in IBM_DPF.txt. The microcode now handles the enablement of the DPF feature.

Querying Primary and Alternate Path Configuration

To check if the data path failover has been enabled in the device driver and display the primary and alternate paths, you may use the Windows utility program *ntutil*. Open a tape library from *ntutil* base mode, choose option “get tape bus info”. If you have multiple paths configured for Tape0, you will see it lists multiple Tape0s. Each one is associated with a different bus address. One of them is marked as “Primary”. The other Tape0s that are not marked as Primary are alternate paths for Tape0.

The following is an example of the output of this option from *ntutil*. Tape0, 1, and 3 have failover enabled. Tape2 does not have failover enabled. Tape3 has two paths configured. The path with the SCSI address 03.01.05.00 is the primary path, and the path with SCSI address 03.01.07.00 is an alternate path.

```
  DEVICE NAME       SERIAL       FAILOVER  SCSI ADDRESS  ACTIVE  PRIMARY
-------------------------------------------------------------------------

   Tape1          000001300486    YES     03.01.01.00     YES   PRIMARY
```

```
Tape2            1200019801     NO     03.01.08.00     YES   PRIMARY
Tape3          000001350051    YES     03.01.05.00     YES   PRIMARY
Tape3          000001350051    YES     03.01.07.00     YES
Tape0          000001365066    YES     03.01.00.00     YES   PRIMARY
```

Checking Disablement of Data Path Failover Setting

If you have disabled the data path failover in device driver's setting by double clicking the *DisableDPF.reg* file and reboot your system, you may go into the registry by issuing the Windows *regedit* command to confirm that DPF has been disabled. Look for a line like the following if your system is Windows Server 2003 or Windows Server 2008:

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\ibmtp2kx]
"FailoverDisabled"=dword:00000001
```

This indicates that DPF has been disabled in the driver. This setting only takes effect after your system is rebooted.

System-Managed Encryption

Device Driver Configuration

System-managed encryption parameters on Windows are placed in the registry under the key for the device driver. The parameters are populated in user-created subkey containing the serial number of the device. The registry keys (**sys_encryption_proxy** and **sys_encryption_write**) are used to determine SME enablement and invocation of the EKM proxy on write, respectively.

Note: Leading zeros in the serial number should be excluded. For example, if the serial number of the encryption-capable tape drive were 0123456789, the user would create the following registry key:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\ibmtp2k3\123456789
```

Under this key, the user would create DWORD values called **sys_encryption_proxy** and/or **sys_encryption_write**, and assign them values corresponding with the desired behavior.

The device driver SME settings can be set for all drives at once by placing the ″**sys_encryption_proxy**″ and ″**sys_encryption_write**″ registry options under the device driver key, found at:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\ibmtp2k3
```

When this option is chosen, the settings established for all drives are overridden by the serial-number specific settings described the previous paragraph.

If no options are specified in the registry, the driver uses the default values for the parameters.

- The default value for **sys_encryption_proxy** is 1.

  This value causes the device driver to handle encryption key requests, if the drive is set up for system-managed encryption. This value should not need to be changed. A value of 0 causes the device driver to ignore encryption key requests for system-managed encryption drives, and is not desirable.
- The default value for **sys_encryption_write** is 2.

  This value causes the device driver to leave the encryption write-from-BOP settings alone. It does not turn on or turn off encryption writing, but instead uses the settings that are already in the drive. If encryption has not been set up previously, then the drive writes unencrypted data. A value of 0 causes the device driver to write unencrypted data. A value of 1 causes the device driver to write encrypted data.

Changes to the registry require a reboot before the settings are able to be viewed; however, during new installations of the driver, if the old driver is not uninstalled, the old settings remain in place and no reboot is required.

Configuration File

The file *%system_root%:\IBMEKM.conf* is used to store the IP address of the EKM server and other network-related parameters. The phrase *%system_root%* refers to the drive letter where the Windows installation is located, typically C (for example C:\IBMEKM.conf).

The format for the EKM server parameters is:

Server<tab>Timout<tab>IPAddress:Port

For example, if the Windows installation is installed at c:\, the ekm server is named ekmtest, the desired timeout is 10 seconds, and the IP address is 127.0.0.1 at port 4242. Then create a line in c:\IBMEKM.conf as follows:

ekmtest<tab>10<tab>127.0.0.1:4242

Querying Tape Drive Configuration

There is an *ntutil* command to query the encryption settings of a tape drive. The *ntutil* command is menu option 59, “get encryption state”.

The following is an example of the output when the drive is configured for system-managed encryption, with encryption turned on:

```
Encryption capable: True
Encryption method: System Managed (2)
Encryption state: On (1)
```

Windows Utility Program (ntutil)

The Windows Utility Program (*ntutil*) allows easy operation of your TotalStorage or Magstar devices and changers. It is supported only with the IBM-supplied device drivers.

The *ntutil* program can be used for the following purposes:
- Help determine if there are hardware or connection problems
- Determine which devices are recognized by the device and changer drivers
- Force a drive dump
- Load new microcode on your drive or changer device
- Send SCSI commands to drives and changers
- Receive the status of SCSI commands
- Obtain sense data for SCSI commands that encounter errors

The *ntutil* program can be run in interactive mode or batch mode. In interactive mode, a menu is presented to issue one command at a time. The results of that command are then presented. In batch mode, you use an editor to create a file that contains *ntutil* commands, which are presented to *ntutil* one by one.

Calling ntutil

The *ntutil* program can be called as a command from the command line or from within a shell script:

```
ntutil <-f input-file> <-o output-file> <-t tape-path-special-file-name>
       <-c changer-path-special-file-name> <-l >
       <-d >
```

The options are:

```
-f input-file   Specifies the input file for batch mode. If a file is specified,
                is specified, ntutil will execute in batch mode and read input
                from this file. The default for this file is ntutil.in.

-o output-file  Specifies the output file for batch mode.
                The default for this file is ntutil.OUT.

-t tape-special-file-name
                Specifies the tape device special file value (for example, tape0)
                to substitute when executing an open (for both batch and
                interactive mode).

-c changer-special-file-name
                Specifies the changer device special file value (for example,
                lb0.1.2.3 for Windows NT, Changer0 for Windows 200x) to substitute
                when executing an open (for both batch and interactive mode). The
                special value def_lun_1 specifies that an open uses the default
                lun 1 associated with the tape-special-file-name.

-l              Specifies that an open will open both the tape path special file and
                the changer path special file (for both batch and interactive mode).

-d              Turns on internal tracing printouts in the output. Used only for
                tool debugging.

-b              Begins ntutil in batch mode (same as -f, but the default input file
                ntutil.in is used.
```

Note: If no parameters are specified, *ntutil* operates in interactive mode.

Interactive Mode

When *ntutil* is called without the *-f* flag, it defaults to interactive (or manual) mode. This mode allows a developer to interactively issue the operations to be performed. When in interactive mode, *ntutil* displays a menu of functions that can be used.

There are two modes available in interactive mode:

- Base mode (LUN0) commands, such as *open, close, read, write*
- Library mode (LUN0 and LUN1) supports *open, close, read*, and *write*, plus medium changer commands such as *read element status* and *move media*.

Note: On Microsoft Windows 200x platforms, library functions are available if the Removable Storage Management component of Microsoft Windows 200x is stopped.

The other TotalStorage or Magstar devices must be in Random mode both when started and to operate *ntutil* in Library mode.

Base mode (for example, only LUN 0 of the device specified by the tape-path-special-file-name tape0 is accessed) is shown in Figure 30 on page 214.

```
Test tool version x.x.x.x
Variable settings
==================== BASE MODE =======================================
tape-special-file-name: tape0
gp->fd0=-1  gp->fd1=-1 block size=1024  block count=1
hex block id = 0000000000000000
return_error_when_fail 1  exit_on_unexpected_result 0  trace_flag 0

manual test menu:
=======================================================================
  1: set device special file            2: display symbols
  3: set block size R/W (now !0 fixed)
  5: set return error when fail         6: set/reset trace
  7: set exit on unexpected result      8: Library Mode
=======================================================================
 20: open                              21: close
 22: read                              23: write
 24: read and display block            25: flush (buffer->media)
 26: read block id                     27: erase
 28: locate block id                   29: display block data
=======================================================================
 30: write filemark(s)                 31: rewind
 32: forward space filemark(s)         33: unload
 34: reverse space filemark(s)         35: load
 36: forward space record(s)           37: return error
 38: reverse space record(s)           39: test unit ready
 43: set media parms (block size)      44: set dev parms (compression)
 46: get device information            47: restore data
 48: get medium information            49: inquiry
 50: poll registered devices
 53: space EOD                         54: display message
=======================================================================
 70: system command
=======================================================================
 80: Force Dump                        81: Read Dump
 82: Update MicroCode                  83: Log Sense
 84: Get Last Sense                    85: Get Version
 86: Associative/Persistent WProtect   87: Read/Write Test
 88: List registered devices           89: Get MTDevice info
=======================================================================
 99: return to main menu
=======================================================================
enter selection:
```

Figure 30. Example of Base Mode

Library mode (LUN 0 specified by the tape-path-special-file-name, for example, tape0, and LUN 1 specified by the changer-path-special-file-name, for example, lb0.1.2.3, are accessed) is shown in Figure 31 on page 215.

```
Test tool version x.x.x.x
Variable settings
==================== LIBRARY MODE ====================================
tape-special-file-name: tape0, changer-special-file-name: lb0.1.2.3
gp->fd0=-1  gp->fd1=-1 block size=1024  block count=1
hex block id = 0000000000000000
return_error_when_fail 1  exit_on_unexpected_result 0  trace_flag 0

manual test menu:
=======================================================================
  1: set device special file            2: display symbols
  3: set block size R/W (now !0 fixed)
  5: set return error when fail         6: set/reset trace
  7: set exit on unexpected result      8: Base Mode
=======================================================================
 10: ioctl return library inventory     11: ioctl move medium
 12: ioctl initialize element status    13: ioctl get changer params
=======================================================================
 20: open                               21: close
 22: read                               23: write
 24: read and display block             25: flush (buffer->media)
 26: read block id                      27: erase
 28: locate block id                    29: display block data
=======================================================================
 30: write filemark(s)                  31: rewind
 32: forward space filemark(s)          33: unload
 34: reverse space filemark(s)          35: load (3590 System Mode)
 36: forward space record(s)            37: return error
 38: reverse space record(s)            39: test unit ready
 43: set media parms (block size)       44: set dev parms (compression)
 46: get device information             47: restore data
 48: get medium information             49: inquiry
 50: poll registered devices
 53: space EOD                          54: display message
=======================================================================
 70: system command
=======================================================================
 80: Force Dump                         81: Read Dump
 82: Update MicroCode                   83: Log Sense
 84: Get Last Sense                     85: Get Version
 86: Associative/Persistent WProtect    87: Read/Write Test
 88: List registered devices            89: Get MTDevice info
=======================================================================
 99: return to main menu
=======================================================================
enter selection:
```

Figure 31. Example of Library Mode

To issue SCSI commands, the device must be open. Open a SCSI device by issuing option 20 (*Open*) from library mode as shown above. The device names can be obtained using command 88 (*List registered devices*) also shown above. Tape device names have the format *tapen*, where *n* is a digit, such as 0, 1, and so on. If the device driver is stopped, then restarted without a reboot, the name will not be the same as it was previously. Rather, the next unused name in the operating system will be used. For example, if there is one tape device defined on the Windows system, that device is named *tape0* when the device driver is started the first time. If the device driver is stopped, then restarted, the name is *tape1*. This behavior continues until the system is restarted.

Batch Mode

Batch input files can contain these kinds of statements:

- Command
- Comment
- Delay
- Exit

- Pause
- Set
- Symbols
- System
- Type

Each type of statement is described in the following section.

Command

Device driver function is exercised by command entries in the input file. Command statements must be on a single line of the input file. The command and command text is case sensitive, but leading or embedded blanks are ignored.

```
command  command-text <result-text>
```

Comment

This statement is used to execute a tape command and to test the command completion status for an expected result.

Lines that start with a pound sign (#) or a space and blank lines are comments and are ignored.

Supported Command Text Fields

The command text specifies the tape operation to perform. The following sections describe the possible values for these fields.

close:

```
SYNTAX: close

        for example, close

FUNCTION Tested:
```

This command calls the device driver CloseHandle entry point and closes the tape device special file opened previously. The RC is always 1.

device_info:

```
SYNTAX: device_info

        for example, device_info

FUNCTION Tested:
```

This command calls the device driver GetTapeParameters entry point with an operation of GET_TAPE_DRIVE_INFORMATION. If the command is successful, the information returned is in a TAPE_GET_DRIVE_PARAMETERS structure, described in the Microsoft Software Development Kit (SDK). The information is formatted and printed. This includes information such as ECC, compression, default block size, and features.

display_block (data):

```
SYNTAX: display_block

        for example, display_block

FUNCTION Tested:
```

This command displays the data buffer contents for the last *read* or *write* transfer.

display_message:

```
SYNTAX: display_message

        for example, display_message

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an operation of input/output control IOCTL_SCSI_PASS_THROUGH and a CDB[0] of 0xC0 (Display Message). If the operation is successful, the message is displayed on the operator panel. The message alternates between **TESTMSG** and **T E S T** unless the Manual mode display message was used previously, in which case that data is displayed.

dump_blocks (read and display blocks):

```
SYNTAX: dump_blocks
        dump_blocks records = n

        for example, dump_blocks records = 2

FUNCTION Tested:
```

This command calls the device driver ReadFile entry point. It attempts to read from the tape device special file opened previously. The amount of data to be read depends on the current settings of the *block_size* and *block_count* variables and on the number of records specified. The first 16 bytes of each record are printed.

This command transfers **n** records. A record has a length of *block_size*block_count* for Fixed-Block mode, and *block_count* for Variable mode. For example, if block_size=1024, block_count=2 and records=3, then three 2048 transfers are done. If the operation succeeds, the RC shows the total number of bytes transferred.

erase:

```
SYNTAX: erase

        for example, erase

FUNCTION Tested:
```

This command calls the device driver EraseTape entry point with an operation of TAPE_ERASE_LONG. It attempts to erase, then leave at load point, the tape device special file opened previously.

find_devices:

```
SYNTAX: find_devices

        for example, find_devices

FUNCTION Tested:
```

This command (which corresponds to the *List registered devices* command on the interactive menu) searches the following registry key and looks for tape and medium changer device identifiers:

```
"HARDWARE\\DEVICEMAP\\Scsi\\Scsi Port W\\Scsi Bus X\\Target Id Y\\Logical Unit Id Z"
```

It then prints a list of those devices supported by the Windows tape and medium changer device drivers.

flush (buffer to media):

```
SYNTAX: flush

        for example, flush

FUNCTION Tested:
```

This command calls the device driver WriteTapemark entry point with an operation of TAPE_FILEMARKS and a count of 0. This writes the data buffer contents to the tape media of the tape device special file opened previously.

force_dump:

```
SYNTAX: force_dump

        for example, force_dump

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an operation of IOCTL_SCSI_PASS_THROUGH and a CDB[0] of SCSIOP_SEND_DIAGNOSTIC. This forces a microcode dump.

forward_filemark:

```
SYNTAX: forward_filemark n

        for example, forward_filemark 2

FUNCTION Tested:
```

This command calls the device driver SetTapePosition entry point with an operation of TAPE_SPACE_FILEMARKS and a count of **n**. It attempts to forward space **n** files on the cartridge in the tape device special file opened previously.

forward_record:

```
SYNTAX: forward_record n

        for example, forward_record 2

FUNCTION Tested:
```

This command calls the device driver SetTapePosition entry point with an operation of TAPE_SPACE_RELATIVE_BLOCKS and a count of **n**. It attempts to forward space **n** records on the cartridge in the tape device special file opened previously.

get_last_sense:

```
SYNTAX: get_last_sense

        for example, get_last_sense

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an operation of OBTAIN_SENSE. This displays the last sense data returned by a sense command.

get_mtdevice:

```
SYNTAX: get_mtdevice

        for example, get_mtdevice

This command is valid for 3590 devices in an automated library only.

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an operation of OBTAIN_MTDEVICE. This displays the library device number (in hexadecimal).

get_version:

```
SYNTAX: get_version

        for example, get_version

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an operation of OBTAIN_VERSION. This displays the Windows Device Drivers Version ID string.

init_element_status:

```
SYNTAX: init_element_status

        for example, init_element_status

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an opcode of LIBRARY_AUDIT. This issues an *Initialize Element Status* command to the device.

inquiry:

```
SYNTAX: inquiry n

        for example, inquiry 0

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an operation of IOCTL_SCSI_PASS_THROUGH and a CDB of SCSIOP_INQUIRY. If the command is successful, the information returned is displayed. Specify *n=0* to obtain inquiry information from the drive. Specify *n=1* to obtain inquiry information from the changer. Only Inquiry page 0 is supported.

load (3590 System Mode and Ultrium devices):

```
SYNTAX: load

        for example, load

FUNCTION Tested:
```

This command calls the device driver PrepareTape entry point with an operation of TAPE_LOAD. It attempts to load the tape media into the drive of the tape device special file opened previously.

locate_block_id:

```
SYNTAX: locate_block_id

        for example, locate_block_id

FUNCTION Tested:
```

This command calls the device driver SetTapePosition entry point with an operation of TAPE_LOGICAL_BLOCK. It attempts to position to the last *read_block_id* value for the cartridge in the tape device special file opened previously or the block ID set by the *set block_id* function.

log_sense (page_code):

```
SYNTAX: log_sense page_code = xx

        for example, log_sense page_code = 00

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an operation of IOCTL_SCSI_PASS_THROUGH and a CDB[0] of SCSIOP_LOG_SENSE. This reads the log sense page specified by the page code and displays the data.

The *page_code (xx)* is two hex characters that specify the log sense page to be read.

medium_info:

```
SYNTAX: medium_info

        for example, medium_info

FUNCTION Tested:
```

This command calls the device driver GetTapeParameters entry point with an operation of GET_TAPE_MEDIA_INFORMATION. If the command is successful, the information returned is in the TAPE_GET_MEDIA_PARAMETERS structure described in the Microsoft SDK. The information is formatted and printed. The information returned is *block_size*, *partition count*, and *write protect*.

move_medium:

```
SYNTAX - Windows NT: move_medium saddr = n daddr = n

         where saddr = decimal source address (moving from)
               daddr = decimal destination address (moving to)

         for example, move_medium saddr = 11 daddr = 13

SYNTAX - Windows 2000 and Windows Server 2003 and Windows Server 2008:
         move_medium stype = n saddr = n dtype = n daddr = n
         where
         stype = decimal source address type
              2=SE(ChangerSlot), 3=IEE (ChangerIEPort), 4=DTE (ChangerDrive)
         saddr = decimal source address (moving from)
         dtype = decimal target address type (see stype for supported values)
         daddr = decimal destination address (moving to)

         for example, move_medium stype = 2 saddr = 1 dtype = 4 daddr = 0
```

In Windows 200x, the addresses are remapped such that the first element of each type is mapped to 0, the second element is mapped to 1, and so on. Issue a *return_lib_inventory_all* command to obtain the mapping.

```
FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an opcode of MOVE_MEDIUM. This attempts to move a data cartridge from a source element location to a destination element location inside the library. If this command is successful, the information returned will be displayed. In Windows NT, the source and destination addresses are the decimal equivalents of the hex Element Addresses described in the appropriate hardware manuals. In Windows 2000, Windows Server 2003, and Windows Server 2008, the addresses are remapped such that the first element of each type is mapped to 0, the second element is mapped to 1, and so on. Issue a *return_lib_inventory_all* command to obtain the mapping.

open:

```
SYNTAX: open <tape-special-file-name> <changer-special-file-name> RW RO

        where RW means read/write
              RO means read only

        The tape-special-file-name is in the form of tapen, and n is a numeric
        value from 0...x, the value is assigned by the device driver during the
        boot process and can be found using the interactive modes option 50 poll
        registered devices. This value may be specified explicitly in the open
        statement or overridden using the command-line -t option.

        The changer-special-file-name, supported on Windows NT, is in the form of
        lbw.x.y.z, w is the device SCSI ID, x is the changer device LUN (usually 1),
        y is the adapter SCSI bus (usually 0), and z is the port number where the
        SCSI adapter resides.  These values are assigned during the boot process and
        can be found using the interactive modes option 50 poll registered devices.
        On Windows 2000, Windows Server 2003, and Windows Server 2008, it is in the
        form Changerx, where x is a numeric value, usually starting with 0.
        This may also contain the value def_lun_1 in which case the utility will
        open LUN 1 associated with the tape-special-file-name. This value may be
        specified explicitly in the open statement or overridden using the
        command-line -c option. If this value is specified, the tape-special-file-name
        must be specified.

        For example:
              open RW        means open the default tape-special-file-name and
                             changer-special-file-name if in library mode or those
                             special file names specified by the -t and -c options

              open tapex RW   means open tapex and if in library mode the default
                             changer-special-file-name. The -t and -c options will
                             override tapex and lbw.x.y.z. The mode will be RW.

              open tapex lbw.x.y.z RW
                             means open tapex and if in library mode the changer
                             special file lbw.x.y.z. The -t and -c options will
                             override tapex and lbw.x.y.z. The mode will be RW.

FUNCTION Tested:
```

This command calls the device driver CreateFile entry point. It attempts to open a tape device special file for LUN0 and LUN1.

poll_devices:

```
SYNTAX: poll_devices

       for example, poll_devices

FUNCTION Tested:
```

This command searches the following registry key:

HARDWARE\\DEVICEMAP\\Scsi\\Scsi Port W\\Scsi Bus X\\Target Id Y\\ Logical Unit Id Z for devices supported by the Windows tape and medium changer device drivers. This command then attempts to open each of the devices found and issue INQUIRY and TEST UNIT READY SCSI commands to each open device. The status of all the devices will be printed on the screen. Executing this command will close all currently open devices. If another application is using the device, this command will not be able to query the status of that device.

read:

```
SYNTAX: read
        read records = n

        for example, read records = 2

FUNCTION Tested:
```

This command calls the device driver ReadFile entry point. It attempts to read from the tape device special file opened previously. The amount of data to be read depends on the current settings of the *block_size* and *block_count* variables and on the number of records specified.

This command transfers **n** records. A record has have a length of block_size*block_count for Fixed-Block mode, and block_count for Variable mode. For example, if block_size=1024, block_count=2 and records=3, then three 2048 transfers are done. If the operation succeeds, the RC shows the total number of bytes transferred.

read_block_id:

```
SYNTAX: read_block_id

        for example, read_block_id

FUNCTION Tested:
```

This command calls the device driver GetTapePosition entry point with an operation of TAPE_LOCICAL_POSITION. It attempts to read the current *block ID* value for the cartridge in the tape device special file opened previously.

read_dump:

```
SYNTAX: read_dump
        read_dump dump_name=string device=devnum
where string is eight characters or less and devnum=0 (for tape) or 1 (for changer)

        For example, read_dump dump_name = fsc0000 device = 0.

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an operation of IOCTL_SCSI_PASS_THROUGH and a CDB[0] of SCSIOP_READ_DATA_BUFF. This reads the dump data and writes it to a file specified by dump_name or to a *dump0000.dmp* default file. Dump_name must be eight characters or less. *Ntutil* will add the .dmp extension automatically. The dump is written into the directory where *ntutil* was started.

read_write (test):

```
SYNTAX: read_write

        for example, read_write

FUNCTION Tested:
```

This command performs the following test on the cartridge in the tape device special file opened previously:
1. Rewind the tape.
2. Set Media Parameters to fixed block 32K.
3. Write a 32K random data block on tape.
4. Rewind the tape.
5. Read the block and compare the data to the data written in step 3.

restore_data:

```
SYNTAX: restore_data
        restore_data file_name=string

        for example, read_data file_name=fsc0000

FUNCTION Tested:
```

This command calls the device driver ReadFile entry point. It attempts to perform 64 KB variable *block read* operations on the tape device special file until either a filemark or an end of data is detected. All the data read, regardless of error, is saved in the file specified. If there is no data on the tape, the command fails, and no data is saved. The RC should be ignored in most cases, because this operation always ends with an error (either filemark detected or end of data detected).

return_error:

```
SYNTAX: return_error_state

        for example, return_error_state

FUNCTION Tested:
```

This command calls GetLastError. If the command is successful, the information returned is displayed. There is no RC for this function.

return_lib_inventory_all:

```
SYNTAX : return_lib_inventory_all

         for example, return_lib_inventory_all

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an opcode of LIBRARY_INVENTORY. It attempts to return information about all the hardware components in a library. If this command is successful, the information returned is displayed. For Windows NT, refer to the appropriate hardware manuals for a description of the information returned by a read element status command.

For Microsoft Windows 2000, Windows Server 2003, and Windows Server 2008, the RSM must be stopped to issue this command. A remapped inventory is returned where one element is returned per line.

Each line has the following format:

Type: n, Addr: hex value, *Empty* or *Full*, access type on a new line.

- n is 2, 3, or 4, which indicates SE (ChangerSlot), or IEE (ChangerIEPort), or DTE (ChangerDrive), respectively.
- hex value is a hexadecimal value that indicates the mapped address of the item. This mapped address is used for medium operations on Microsoft Windows 2000, Windows Server 2003, and Windows Server 2008, which is different from using the actual hardware element addresses on Windows NT.
- *Empty* or *Full* indicates if the element is occupied. Note that if a cartridge is in the drive, *Empty* is returned.
- access type is set to *Access* if the element is accessible, blank if not.

If tracing is enabled, the flags field for the element is displayed after the element address. Potential values for this field are documented in the ELEMENT_STATUS_abc constants in *ntddchgr.h*.

reverse_filemark:

```
SYNTAX: reverse_filemark n


        for example, reverse_filemark 2

FUNCTION Tested:
```

This command calls the device driver SetTapePosition entry point with an operation of TAPE_SPACE_FILEMARKS and a count of **n**. It attempts to backward space **n** files on the cartridge in the tape device special file opened previously.

reverse_record:

```
SYNTAX: reverse_record n

        for example, reverse_record 2

FUNCTION Tested:
```

This command calls the device driver SetTapePosition entry point with an operation of TAPE_SPACE_RELATIVE_BLOCKS and a count of **n**. It attempts to backward space **n** records on the cartridge in the tape device special file opened previously.

rewind:

```
SYNTAX: rewind

        for example, rewind

FUNCTION Tested:
```

This command calls the device driver SetTapePosition entry point with an operation of TAPE_REWIND. It attempts to rewind the tape device special file opened previously. Status is presented when the rewind is complete.

set_device_parameters (compression):

```
SYNTAX: set_device_parameters compression = n

        for example, set_device_parameters compression = 0

FUNCTION Tested:
```

This command calls the device driver SetTapeParameters entry point with an operation of SET_TAPE_DRIVE_INFORMATION. The compression value is a required parameter with the command.

compression - device compression On or Off (0 for Off, any other value On).

set_media_parameters (block size):

```
SYNTAX: set_media_parameters block_size = n

        for example, set_media_parameters block_size = 0

FUNCTION Tested:
```

This command calls the device driver SetTapeParameters entry point with an operation of SET_TAPE_MEDIA_INFORMATION. The *block_size* value is a required parameter with the command.

block_size - device block size setting (0 for variable).

set_reset_write_protect:

```
SYNTAX: set_reset_write_protect wp = x

        for example, set_reset_write_protect wp = 1

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an operation of IOCTL_SCSI_PASS_THROUGH, a CDB[0] of MODE_SELECT (0x55), and a mode page of 0x23. Refer to the appropriate hardware reference manual for details on Associative and Persistent Write Protect. The write protect value on the mode page is updated based on the **wp** parameter, where **wp** is:

```
1 - Set Associative Write Protect
2 - Set Persistent Write Protect
3 - Set Both
4 - Reset Associative
8 - Reset Persistent
0 - Reset Both
```

space_eod:

```
SYNTAX: space_eod

        for example, space_eod

FUNCTION Tested:
```

This command calls the device driver SetTapePosition entry point with an operation of TAPE_SPACE_END_OF_DATA. It attempts to space to the end of data on the cartridge in the tape device special file opened previously.

test_unit_ready:

```
SYNTAX: test_unit_ready n

        for example, test_unit_ready 1

FUNCTION Tested:
```

This command calls the device driver DeviceIoControl entry point with an operation of IOCTL_SCSI_PASS_THROUGH and a CDB[0] of

SCSIOP_TEST_UNIT_READY. If the operation is unsuccessful, the sense data is displayed. Specify 0 to send the command to the drive. Specify 1 to send the command to the changer.

unload:

```
SYNTAX: unload

        for example, unload


FUNCTION Tested:
```

This command calls the device driver PrepareTape entry point with an operation of TAPE_UNLOAD. It attempts to rewind and unload the tape device special file opened previously.

update_code:

```
SYNTAX: update_code image_name = string

        for example, update_code image_name = d0i9_430

FUNCTION Tested:
```

This command updates the microcode and calls the device driver DeviceIoControl entry point with an operation of IOCTL_SCSI_PASS_THROUGH and a CDB[0] of SCSIOP_WRITE_DATA_BUFF. The filename portion of the image name must be specified and is read from the directory where *ntutil* was started. The file extent must be **.fmr** for the file.

write:

```
SYNTAX:  write
         write records = n
         write records = n data = m

         for example, write records = 3 data = 888

FUNCTION Tested:
```

This command calls the device driver WriteFile entry point. It attempts to write to the tape device special file opened previously. The amount of data to be written depends on the current settings of the *block_size* and *block_count* variables and on the number of records specified.

This command transfers **n** records. A record has a length of block_size*block_count for Fixed-Block mode, and block_count for Variable mode. For example, if block_size=1024, block_count=2 and records=3, then three 2048 transfers are done. If the operation succeeds, the RC shows the total number of bytes transferred.

If the records attribute is not specified, the default is 1.

Each record written has random bytes preceded by an integer identifier. If the data attribute is not specified, the identifier is a unique sequential counter. This can be overridden by specifying **data=**. This value can be checked on a read command by specifying, for example, result data=888. If data is specified, the record count **must** be specified. Data can be a decimal value from 0–4294967295.

write_filemark:

```
SYNTAX: write_filemark n

        for example, write_filemark 2 (write 2 file marks)

FUNCTION Tested:
```

This command calls the device driver WriteTapemark entry point with an operation of TAPE_FILEMARKS and a count of **n**. It attempts to write **n** filemarks on the tape media.

Supported Result Text Fields

The result text on a command statement is used to test completion status for an expected result. Further execution of a test case is affected by the current settings of the *return_error_when_fail* and *exit_on_unexpected_result* variables. These can be set using a *set* statement.

If no result text is specified, the command is treated as successful.

In the case of rc, except as noted in the command section, a failure returns -1. Otherwise, the operation succeeded.

The syntax for result text is:

```
     < rc <  n> <err <  n> <cc <  n>
         >          >         >
         <=         <=        <=
         >=         >=        >=
         ==         ==        ==
         !=         !=        !=

where:

rc   = return code from the last command.
err  = Operating System GetLastError () value for the last command.
data = expected record identifier for a read command.

For example, consider this statement in the input file:

    command write 1 result rc == -1 err == WRITE_PROTECT

This means we will try to write one record to tape and we expect this to fail
with return code -1, err WRITE_PROTECT.

Note: Symbolic values or integers can be used for err. See the
      appendix on symbolic values for a list of those recognized.
```

set: This statement allows setting of variables that affect how tests are executed. The syntax for a *set* statement is:

```
set variable = value

for example, set return_error_when_fail = 1
```

Here are the variables that can be set:

return_error_when_fail
Can be set to 1 (true) or 0 (false). A setting of true means *ntutil* ends by returning an error at the end of the test if any command does not produce a correct expected result as specified in result text.

exit_on_unexpected_result
Can be set to 1 (true) or 0 (false). If set to true, the first occurrence of an unexpected result terminates the test.

block_id
Can be set to a four byte hex value (for example, 0001aa03). This value is used on a *locate_block_id* command.

block_size
Block size to be used for reading, writing or set_mode. Can be set to 0 to indicate Variable Block mode.

block_count
Number of blocks or bytes (Variable mode) to transfer on an individual data transfer command.

type
Text following the word **type** is typed on the terminal. This can be used to show the progress of the test or to prompt for a subsequent pause statement.

```
type string

for example, type tell operator it's lunch time
```

pause
The test stops until a character is typed at the terminal. This allows for tests that require manual intervention.

delay
The test stops for **n** seconds. This allows for tests that require delay for mechanical completion before continuing. The syntax for the delay statement is:

```
delay value

for example, delay 1
```

system
Text following the word system is passed to the system for execution. The syntax for the system statement is:

```
system operation

for example, system dir
```

symbols
This statement prints the symbol list that may be used in result text fields.

exit
This statement causes immediate termination of the batch test. No further lines are read from the input file.

Symbolic Values

Figure 32 on page 229 shows the values (symbolic error or error number) that can be used for *err*.

Symbolic Error	NT/Win2000 Error #	Symbolic Error	NT/Win2000 Error #
ERROR_SUCCESS	0	DD_CLEANER_INST	316
ERROR_INVALID_FUNCTION	1	DD_MEDIA_NOT_EJECTED	317
ERROR_FILE_NOT_FOUND	2	DD_IOPORT_NOT_CONFIG	318
ERROR_PATH_NOT_FOUND	3	DD_FIRST_DEST_EMPTY	319
ERROR_TOO_MANY_OPEN_FILES	4	DD_END_PHYSICAL_MEDIA	400
ERROR_ACCESS_DENIED	5	DD_MEDIA_BLANK	401
ERROR_INVALID_HANDLE	6	DD_MEDIA_CORRUPTED	402
ERROR_NOT_ENOUGH_MEMORY	8	DD_MEDIA_FAILURE	403
ERROR_BAD_FORMAT	9	DD_MEDIA_INCOMPATIBILITY	404
ERROR_INVALID_BLOCK	10	DD_SECTOR_RELOCATION	405
ERROR_BAD_ENVIRONMENT	11	DD_SECTOR_OUT_OF_RANGE	406
ERROR_INVALID_ACCESS	12	DD_WRITE_PROTECT	407
ERROR_INVALID_DATA	13	DD_CLEAN_MEDIA	408
ERROR_OUTOFMEMORY	14	DD_MEDIA_FAULT	409
ERROR_INVALID_DRIVE	15	DD_CLEANING_COMPLETE	410
ERROR_WRITE_PROTECT	19	DD_LOGICAL_END_OF_MEDIA	411
ERROR_BAD_UNIT	20	DD_MEDIA_NOT_PRESENT	412
ERROR_NOT_READY	21	DD_BEGINNING_OF_MEDIA	413
ERROR_BAD_COMMAND	22	DD_ERASE_FAILURE	414
ERROR_CRC	23	DD_WRITE_TO_WRITTEN_WORM	415
ERROR_HANDLE_EOF	38	DD_WRONG_LENGTH_BLOCK	416
ERROR_NOT_SUPPORTED	50	ERROR_IO_INCOMPLETE	996
ERROR_DEV_NOT_EXIST	55	ERROR_IO_PENDING	997
ERROR_ALREADY_ASSIGNED	85	ERROR_NOACCESS	998
ERROR_INVALID_PARAMETER	87	ERROR_CANTOPEN	1011
ERROR_OPEN_FAILED	110	ERROR_CANTOPEN	1011
ERROR_INSUFFICIENT_BUFFER	122	ERROR_CANTREAD	1012
ERROR_INVALID_NAME	123	ERROR_CANTWRITE	1013
ERROR_BUSY_DRIVE	142	ERROR_END_OF_MEDIA	1100
DD_NO_SENSE	200	ERROR_FILEMARK_DETECTED	1101
DD_DEVICE_DRIVER_FAILURE	201	ERROR_BEGINNING_OF_MEDIA	1102
DD_EEPROM_FAILURE	202	ERROR_SETMARK_DETECTED	1103
DD_MANUAL_INTERVENTION	203	ERROR_NO_DATA_DETECTED	1104
DD_RECOVERED_ERROR	204	ERROR_PARTITION_FAILURE	1105
DD_SCSI_ADAPTER_ERROR	205	ERROR_INVALID_BLOCK_LENGTH	1106
DD_SCSI_ERROR	206	ERROR_DEVICE_NOT_PARTITIONED	1107
DD_SCSI_BUSY	211	ERROR_UNABLE_TO_LOCK_MEDIA	1108
DD_ILLEGAL_REQUEST	207	ERROR_UNABLE_TO_UNLOAD_MEDIA	1109
DD_COMMAND_ABORTED	208	ERROR_MEDIA_CHANGED	1110
DD_HARDWARE_MICROCODE	209	ERROR_BUS_RESET	1111
DD_UNIT_ATTENTION	210	ERROR_NO_MEDIA_IN_DRIVE	1112
ERROR_MORE_DATA	234	ERROR_IO_DEVICE	1117
DD_CARTRIDGE_ENTRY_FAILURE	300	ERROR_TOO_MANY_LINKS	1142
DD_CARTRIDGE_LOAD_FAILURE	301		
DD_CARTRIDGE_IN_FAILED_DRIV	302		
DD_CAROUSEL_NOT_LOADED	303		
DD_CHANGER_FAILURE	304		
DD_DRIVE_FAILURE	305		
DD_DRIVE_OR_MEDIA_FAILURE	306		
DD_ENTRY_EXIT_FAILURE	307		
DD_ENTRY_EXIT_NOT_PRESENT	308		
DD_LIBRARY_AUDIT	309		
DD_LIBRARY_FULL	310		
DD_MEDIA_EXPORT	311		
DD_SLOT_FAILURE	312		
DD_SLOT_OR_MEDIA_FAILURE	313		
DD_SOURCE_EMPTY	314		
DD_DESTINATION_FULL	315		

Figure 32. Symbolic Values

Problem Determination

There is a debug version of the device driver that can be used if you encounter problems. The debug version of the driver issues DbgPrint messages at various places during device driver execution. To capture these messages, you must start a debugger or use a tool like Debug View, available from:

`http://www.sysinternals.com`

Windows NT Instructions

Using the Debug Version

To install and use the debug version of the device driver, perform the following steps after the driver has initially been installed:

1. Quiesce all activity on the tape and medium changer devices.
2. Exit all applications that are using the tape and medium changer devices.
3. Stop the device drive (*MagStar.sys* or *IBMtape.sys*). Refer to "Manual Starting and Stopping Procedures" on page 203 for details on stopping the driver.
4. Locate the *...\checked* directory for the device driver level that you are running. If you chose the default installation location when you installed the drivers, the debug drivers will be found at:
   - *c:\Program files\IBM Corporation\IBM Magstar Device Drivers\n.n.n.n\checked* for TotalStorage or Magstar devices,
   - *c:\Program files\IBM Corporation\IBM Ultrium Device Drivers\n.n.n.n\checked* for Ultrium devices,

   where n.n.n.n are integers that indicate the driver level.

   To determine the driver level, find *c:\winnt\system32\drivers\(ibmtape.sys or magstar.sys)* using Windows Explorer, right-click the file, select Properties, then select the Version tab. If you do not find a *...\checked* directory for your driver, you must install the Device Driver Depot component of the Windows Device Driver Installation Package.
5. Go to the root prompt, for example, c:.
6. Change to the appropriate directory by typing *cd \winnt\system\drivers*.
7. Copy *magstar.sys magstar.orig* or *IBMtape.sys ibmtape.orig*.
8. Copy *magstar.sys* or *IBMtape.sys* from the appropriate *... \checked* directory to *c:\winnt\system32\drivers\(magstar.sys or ibmtape.sys)*. For example, if you installed the Ultrium device drivers at the default installation location and your driver level was 1.1.6.9, you would issue the following command: *copy c:\Program files\IBM Corporation\IBM Ultrium Device Drivers\1.1.6.9\checked\ IBMtape.sys c:\winnt\system32\drivers\ibmtape.sys*.
9. Start the debugger or tool that captures the *DbgPrint* messages.
10. Start the Windows device driver. Refer to "Manual Starting and Stopping Procedures" on page 203 for details on starting the driver.

Registry variable *HKEY_LOCAL_MACHINE/SYSTEM/CurrentControlSet/Services/(Magstar or Ultrium)/MsgLevel* determines how many messages are issued by the debug version of the driver. When the debug version of the driver is started for the first time, it defines the registry variable. The variable is ignored by the non-debug version of the driver.

Using *regedt32*, you can set the value of this variable, to any value from REG_DWORD 0x0 to REG_DWORD 0x5, inclusive. The greater the value, the more messages will be issued.

Restoring the Non-Debug Version

To restore the non-debug version of the driver, perform the following steps:

1. Quiesce all activity on the tape and medium changer devices.
2. Exit all applications that are using the tape and medium changer devices.
3. Stop the device driver (*Magstar.sys* or *IBMtape.sys*). Refer to "Manual Starting and Stopping Procedures" on page 203 for details on stopping the driver.
4. Go to the root prompt, for example, c:.
5. Change to the appropriate directory by entering *cd \winnt\system\drivers*.
6. Copy *magstar.orig magstar.sys* or *ibmtape.orig IBMtape.sys*.
7. Start the device driver. Refer to "Manual Starting and Stopping Procedures" on page 203 for details on starting the driver.

Windows 2000/Windows Server 2003/Windows Server 2008 Instructions

Using the Debug Version

To install and use the debug version of the device driver, perform the following steps after the driver has initially been installed:

1. Quiesce all activity on the tape and medium changer.
2. Exit all applications that are using the tape and medium changer devices.
3. Locate the *\checked* folder for the device driver level that you are running. This folder is in the highest level directory of the driver package, and it contains checked versions of the tape and medium changer device drivers, *ibmtpxxyyy.sys* and *ibmcgxxyyy.sys*, where
   - *xx* = **ft** for the filter driver, **bs** for the bus driver, or **blank** for the base driver, and
   - *yyy* = **2k** for Windows 2000, or **2k3** for Windows Server 2003, or **2k8** for Windows Server 2008. Refer to Figure 33 on page 232.

   Note: The last driver level to include support for Windows 2000 is V6.1.4.8. Subsequent levels include support for Windows Server 2003 and Windows Server 2008 only, and therefore will only contain **a2k3** files.

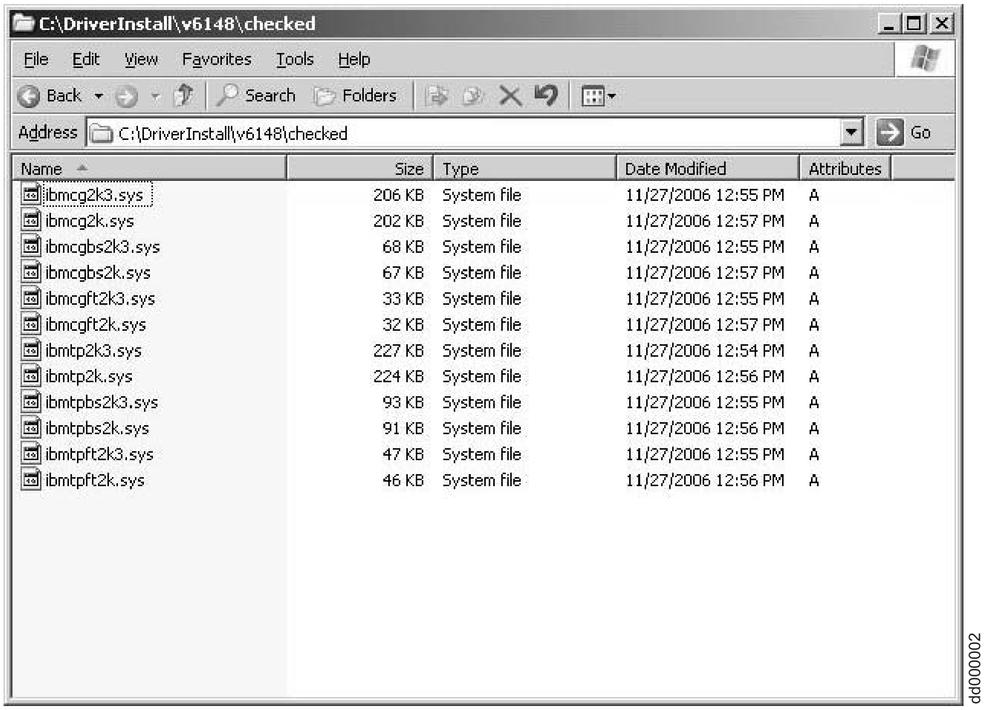


Figure 33. Checked folder.

4. Copy the checked version of *ibmtpxxyyy.sys* or *ibmcgxxyyy.sys* to *\winnt\system32\drivers*, overwriting the version of the file already there.
5. Reboot the system.
6. Start the debugger to capture the *DbgPrint* messages.
7. Issue the commands to the driver. You should see debug statements printed to the debugger window.

Restoring the Non-Debug Version

To restore the non-debug version of the driver, perform the following steps:

1. Quiesce all activity on the tape and medium changer devices.
2. Exit all applications that are using the tape and medium changer devices.
3. In the highest level directory of the driver package, you will find non-debug versions of the tape and medium changer device drivers, *ibmtpxxyyy.sys* and *ibmcgxxyyy.sys*, where
   - *xx* = **ft** for the filter driver, **bs** for the bus driver, or **blank** for the base driver, and
   - *yyy* = **2k** for Windows 2000, or **2k3** for Windows Server 2003, or **2k8** for Windows Server 2008. Refer to Figure 33.

   Note: The last driver level to include support for Windows 2000 is V6.1.4.8. Subsequent levels include support for Windows Server 2003 and Windows Server 2008 only, and therefore will only contain **2k3** files.
4. Copy the non-debug version of *ibmtpxxyyy.sys* or *ibmcgxxyyy.sys* to *\winnt\system32\drivers*, overwriting the version of the file already there.
5. Reboot the system. When the driver has started and commands are issued to it, the driver will no longer produce debug output.

Chapter 8. Tru64 Device Driver

This chapter describes the Tru64 support available for the IBM TotalStorage Enterprise Tape System 3590 and the IBM 3584 UltraScalable Tape Library.

Purpose

The Tru64 Native tape and medium changer device drivers (cam_tape, cam_changer) are designed to take advantage of the features provided by tape drives and medium changer devices. The goal is to give applications access to the functions required for basic tape operations (such as backup and restore) and medium changer operations (such as mount and demount the cartridges).

Product Requirements

The following hardware and software components are supported by IBM.

Hardware Requirements

The following hardware is supported by the Tru64 Native Tape and Medium Changer driver:

- One or more of the following IBM tape and medium changer device:
  - IBM Ultrium LTO 1 tape drive
  - IBM System Storage TS3500, 3581 and 3583 Tape Libraries
  - IBM TotalStorage Enterprise Tape Drive 3590, Model E11 or H11 (Rack Mount with ACF) with FC interface
  - IBM TotalStorage Enterprise Tape Drive 3590, Model E1A or H1A (3494 Tape Library Model) with FC interface
- One or more of the following Fibre Channel Host Bus Adapter:
  - 64 bit PCI to Fibre Channel Host Bus Adapter (Compaq part # DS-KGPSA-CA) Version: 1.32a, F/W Rev: 3.81A4

Software Requirements

Tru64 5.1A Operating System with native device driver and medium changer device driver.

Setting Up the Environment

To set up IBM Tape and Medium changer devices to operate with the Tru64 operating system:

1. Confirm that Tru64 Operating System version 5.1A is properly installed.
2. Confirm that the Host Bus Adapter card and appropriate firmware levels are properly installed on a Compaq Alpha system.

3. Connect the tape and medium changer devices to the host bus adapter card following the instructions for the devices.
4. Power up the tape and medium changer devices, if not already powered up, and wait until they have initialized.
5. Login as root administrator or use the *su* command to gain superuser privileges.
6. At the command line, issue the */sbin/hwmgr -scan scsi* command to detect tapes or medium changers that are connected to the host.
7. To verify the scanned results, issue */sbin/hwmgr -view dev* at the command prompt.
8. Add a new entry to the */etc/ddr.dbase* file for the particular tape or medium changer devices.Figure 34 on page 235 is an example output of a working device configuration for an IBM TotalStorage 3590 tape drive.

   For more information on *ddr_config* utility and the *ddr.dbase* file, refer to the Tru64 documentation and man pages.
9. Issue the */sbin/ddr_config -c /etc/ddr.base* command to compile and link the new entry to the driver module.
10. The native driver will create device special files based on the information given in the *ddr.dbase* file.

```
*SCSIDEVICE
    #
    type = tape
    Name = "IBM" "03590E1A"          #Vendor ID and Product ID

    #

    PARAMETERS:
        TypeSubClass            = rdat
        MaxTransferSize         = 0x0ffffff #(16M-1)
        ReadyTimeSeconds        = 45
        CMD_PreventAllow        = supported
        CMD_ExtReserveRelease   = supported

   DENSITY:
        DensityNumber           = 0
        DensityCode             = Default

   DENSITY:
        DensityNumber           = 1
        DensityCode             = Default
        Blocking                = 0x200         #block size
        Buffered                = 1             #buffered mode

   DENSITY:
        DensityNumber           = 2
        DensityCode             = Default
        CompressionCode         = 0x00          #compression off
        Blocking                = 0x8000        #block size
        Buffered                = 1             #buffered mode

   DENSITY:
        DensityNumber           = 3
        DensityCode             = Default
        CompressionCode         = 0x00          #compression off
        Blocking                = 0x10000       #block size
        Buffered                = 1             #buffered mode

   DENSITY:
        DensityNumber           = 4,5,6,7
        DensityCode             = Default
        CompressionCode         = 0x01          #compression on
        Blocking                = 0x20000       #block size
        Buffered                = 1             #buffered mode
```

Figure 34. Example Output for Tru64 Device Driver and IBM 3590 Tape Drive

RAS Utility Program For Tru64 System (IBMrasutil)

IBMrasutil is a RAS utility program which allows the operator to obtain device dumps from IBM tape and medium changer devices for diagnostic purposes. It is also used to update microcode on the IBM tape and medium changer devices. The following IBM devices are supported:

- IBM Ultrium LTO 1 tape drive inIBM System Storage TS3500, 3581 and 3583 Tape Libraries
- IBM TotalStorage Enterprise 3590 Tape Drive Models E11 or H11 (Rack Mount with ACF)
- IBM TotalStorage Enterprise 3590 Tape Drive Models E1A or H1A (3494 Tape Library Models)

Installation Procedure from the Device Driver FTP Site

If you want to download the RasUtil kit from the FTP site, enter this FTP address:

ftp://ftp.software.ibm.com/storage/devdrvr/Tru64

After you have downloaded the RasUtil kit from the FTP site, untar the kit first and then run the *setld* command to install:

```
tar -xvf RasUtil.x.x.x.kit.tar
setld -l RasUtil.x.x.x.kit
```

After installing the IBM RAS utility program, use *vi* or some other text editor to edit the *.profile* file to add the */usr/opt/RASUTIL/bin* to the search path.

To find more information on how to execute the IBM RAS utility program, issue:

```
IBMrasutil -h
```

Uninstalling

The IBM RAS Utility Program can be uninstalled by using the *setld* command:

```
setld -d IBMRASxxx
```

Update Procedure

To update the RasUtil kit to a newer version, remove the old version first.

```
setld -d IBMRASxxx
```

then follow the above installation steps to install the new version.

Query Procedure

You can determine if the IBM RAS Utility Program kit is installed on the system by using the following command.

```
setld -i|grep IBMRAS
```

To display the RasUtil product kit's fileset and the located directory, enter:

```
setld -i IBMRASxxx
```

Verify Procedure

Use this command to verify the existence of the installed RasUtil program. The *setld -v* command executes any V phase processing included in the subset control program except during installation.

```
setld -v IBMRASxxx
```

Interactive Mode

The interactive mode for the RAS utility program can be invoked from the command line by using the IBMrasutil command. The program will prompt you to enter a device special file name. You must open a device before you can issue any RAS utility subcommands.

Command-Line Mode

The command-line mode for the RAS utility program (*IBMrasutil*) provides the same basic RAS utility commands as the interactive mode. Invoke the *IBMrasutil* command from the Tru64 command line or from within a shell script. If you enter the *IBMrasutil* command without any arguments, the interactive mode will be invoked. The syntax for the command-line mode of the RAS utility program is:

```
IBMrasutil -f Device -Option Filename
```

Note:

1. Device is the name of the device special file (for example, */dev/ntape/tape7*)
2. Filename for input or output operations
3. If only the -f option is issued, the program will query and display the Model Name, Serial Number and the Firmware level of the device.
4. If the -f option is not issued, the interactive mode will be invoked.

Command-Line Options

-D filename

Read device dump information and output it into a file specified by the filename. For example,

```
IBMrasutil -f /dev/ntape/tape7 -D DriveDump.log
```

-M filename

Load the microcode from the specified file to the device. For example,

```
IBMrasutil -f /dev/ntape/tape7 -M 2360.bin
```

Chapter 9. 3494 Enterprise Tape Library Support

Purpose

The IBM 3494 Library device driver is a device driver providing attachment for the IBM TotalStorage 3494 Enterprise Automated Tape Library and IBM TotalStorage Virtual Tape Server (VTS) to a server. The programs described in this chapter support the 3494 Enterprise Automated Tape Library on the following operating systems/platforms:

- AIX on IBM POWER-based servers
- HP-UX
- Linux
- Sun/Solaris
- Microsoft Windows NT
- Microsoft Windows 2000®
- Microsoft Windows 2003
- Microsoft Windows 2008
- Tru64
- SGI/IRIX

MTLIB Program

The *mtlib* program is a command-line interface used to control the 3494 Enterprise Tape Library and is installed, by default, for execution by all users on the system. This program provides the full interface (except for the MTIOCLEW library system call) as specified in the *IBM Tape Device Drivers: Programming Reference*, GA32-0566. Refer to "MTEVENT Program" on page 251.

The *mtlib* program provides an interface to the physical tape drives and volumes within a 3494 Tape Library and also to the virtual tape drives and volumes within an attached IBM TotalStorage Virtual Tape Server (VTS).

To issue commands to the physical tape drives and volumes, specify a device special file name, such as */dev/lmcp0* or the logical name of the library, such as *libmgrc7*.

To issue commands to the virtual tape drives and volumes in an attached VTS library, specify a device special file name or the logical name of the library with the name of the logical VTS library appended with a slash (/). For example, to issue commands to the first attached VTS library, use */dev/lmcp0/vts1* or *libmgrc7/vts1*. The attached VTS libraries are named logically *vts1*, *vts2*, and so on and correspond to the order in which they were configured in the Tape Library. The *-D* flag, in conjunction with the *-E* flags, can be used to display the attached VTS devices and the number of the VTS library.

Syntax and Examples

Figure 35 shows the syntax for the *mtlib* program .

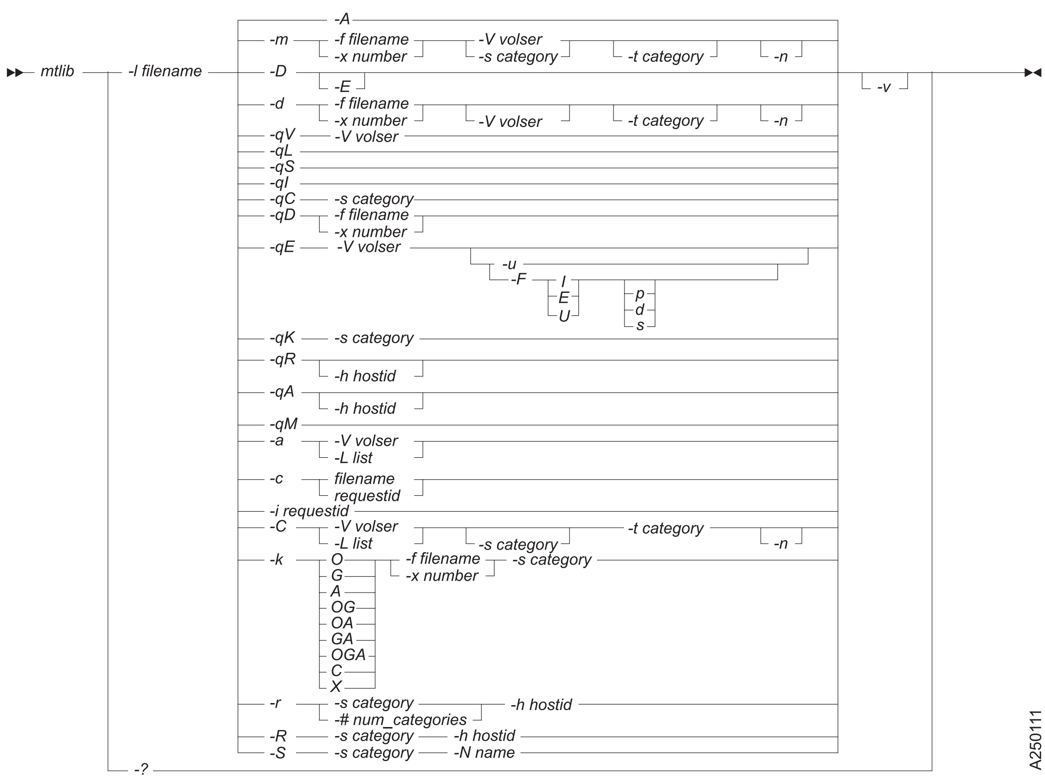


Figure 35. Syntax Diagram for mtlib program

The *mtlib* program has the following flags:

Flag	Description
-f[filename]	Device special file for the drive, for example, */dev/rmt0* (AIX), */dev/rmt/0st* (Sun), */dev/rmt/0m* (HP), */dev/IBMtape0* (Linux), \\.*tape0* (Windows)
-x[number]	Device number of the drive, for example, 518350
-l[filename]	On AIX, library special file name, for example, */dev/lmcp0*, */dev/lmcp0/vts1*. For non-AIX, the logical name of the library, for example, *libmgrc7*.
-q[type]	Query the library information option:

Type	Description
V	Volume data
L	Library data
S	Statistical data
I	Inventory data
C	Category inventory data
D	Device data
E	Expanded volume data
K	Inventory volume count data
R	Reserved category list
A	Category attribute list
M	All mounted volumes

Flag	Description
-D	Return an array of devices configured in the specified library
-E	Return (if used with the -D option) an array of expanded information for all devices configured in the specified library, including the control unit ID, device, and VTS library number
-m	Mount option
-d	Demount option
-c[requestid]	Cancel the pending request option
-n	No wait mode
-i[requestid]	Query the request ID status option
-C	Change the category of a volume
-a	Audit the specified volume
-k[flags]	Assign a category (with one of the following flags) to a device in the library:

Type	Description
O	Enable the category order
C	Clear the cartridge loader
G	Generate the first mount
A	Enable the auto mount

X Remove the device category assignment

Note: Valid combinations are OG, OA, GA, and OGA.

-r Reserve the category

-R Release the category

-S Set the category attribute

Note: The categories must be reserved before using this option.

-s[category] Source or starting category

-t[category] Target category

-V[volser] Volume serial number

-L[list] Filename containing a list of the volume serial numbers. Each Volume Serial should be entered as one per line in the file.

-N[name] Category name to assign to the category (valid characters are uppercase A-Z, 0-9, -, *, or blank).

-h[hostid] Host ID for the reserve or release category or the R/A option for the query command.

-u Include usage date in the expanded volume data (used in conjunction with the -qE option). The default is the ISO format with a period separator. The format can be specified with the -F option.

-F[flags] Format or separator for volume usage and date with the -u option:

Type	Description
I	ISO/Japan yyyy.mm.dd
E	Europe dd.mm.yyyy
U	U.S.A. mm.dd.yyyy
p	Period separator mm.dd.yyyy
d	Dash separator mm-dd-yyyy
s	Slash separator mm/dd/yyyy
-v	Verbose
-#[num_categories]	Number of categories to reserve
-A	Query library addresses and status.
-?	Help text

Note: The *-l* argument is required.

The report in Figure 36 was produced by:

```
mtlib -l /dev/lmcp0 -f /dev/rmt5 -qD (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -f /dev/rmt5 -qD (for AIX VTS library 1)
mtlib -1 libmgrc7 -f /dev/rmt/5st -qD (for Sun)
mtlib -1 libmgrc7 -f /dev/rmt/5m -qD (for HP)
mtlib -1 libmgrc7 -f /dev/IBMtape5 -qD (for Linux)
mtlib -l libmgrc7 -f \\.\tape0 -qD (for Windows)
```

```
Device Data:
   mounted volser............TAF500
   mounted category..........FF00
   device category...........0000
   device state..............Device installed in Library.
                             Device available to Library.
                             Volume is loaded.
                             ACL is installed.
   device class..............3590-B1A
   extended device status....00
```

Figure 36. Device Query

The report in Figure 37 was produced by:

```
mtlib -l /dev/lmcp0 -qV -VCS2000 (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -qV -VCS2000 (for AIX VTS library 1)
mtlib -l libmgrc7 -qV -VCS2000 (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Volume Data:
   volume state.........00
   logical volume.......No
   volume class.........3590 1/2 inch cartridge tape
   volume type..........HPCT 320m nominal length
   volser...............CS2000
   category.............FE00
   subsystem affinity...04 03 05 06 01 02 00 00
                        00 00 00 00 00 00 00 00
                        00 00 00 00 00 00 00 00
                        00 00 00 00 00 00 00 00
```

Figure 37. Volume Query

The report in Figure 38 was produced by:

```
mtlib -l /dev/lmcp0 -qE -VCS2000 (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -qE -VCS2000 (for AIX VTS library 1)
mtlib -l libmgrc7 -qE -VCS2000 (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Expanded Volume Data:
   volume status........00
   logical volume.......No
   volume class.........3590 1/2 inch cartridge tape
   volume type..........HPCT 320m nominal length
   volser...............CS2000
   device category......FF00
```

Figure 38. Expanded Volume Query

The report in Figure 39 was produced by:

```
mtlib -l /dev/lmcp0 -qE -u -VCS2000 (for AIX physical library)
mtlib -l libmgrc7 -qE -u -VCS2000 (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Expanded Volume Data with Usage:
   volume status............00
   logical volume...........No
   volume class.............3590 1/2 inch cartridge tape
   volume type..............HPCT 320m nominal length
   volser...................CS2000
   device category..........FF00
   last used (yyyy.mm.dd)...2001.08.26
```

Figure 39. Expanded Volume Data with Usage

The report in Figure 40 was produced by:

```
mtlib -l /dev/lmcp0 -qK -v (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -qK -v (for AIX VTS library 1)
mtlib -l libmgrc7 -qK -v (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Performing Query Inventory Volume Count Data using /dev/lmcp0
Inventory Volume Count Data:
   sequence number......12345
   number of volumes....207
   category.............0000
```

Figure 40. Inventory Count Data

The report in Figure 41 was produced by:

```
mtlib -l /dev/lmcp0 -D (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -D  (for AIX VTS library 1)
mtlib -l libmgrc7 -D  (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

Refer to also the *tapeutil* subcommand *mtdevice*.

```
0, 00515820 003490C2A00
1, 00515821 003490C2A01
```

Figure 41. Tape Library Device Number

The report in Figure 42 was produced by:

```
mtlib -l /dev/lmcp0 -DE (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -DE (for AIX VTS library 1)
mtlib -1 libmgrc7 -DE (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

Refer to also the *tapeutil* subcommand *mtdevice*.

```
 Type    Mod  Serial #  Devnum    Cuid  Device  VTS Library
003590   B1A  13-10800  00108000    1     0
003590   B1A  13-10800  00108001    1     1
003590   B18  13-01817  00018170    2     0         1
003590   B18  13-01817  00018171    2     1         1
003590   B18  13-01817  00018172    2     2         1
003590   B18  13-01817  00018175    2     3         1
003490   C2A  13-01817  00FF0100    3     0         1
003490   C2A  13-01817  00FF0101    3     1         1
003490   C2A  13-01817  00FF0110    3     2         1
003490   C2A  13-01817  00FF0111    3     3         1
003490   C2A  13-01817  00FF0120    3     4         1
003490   C2A  13-01817  00FF0121    3     5         1
003490   C2A  13-01817  00FF0130    3     6         1
003490   C2A  13-01817  00FF0131    3     7         1
```

Figure 42. Expanded Tape Library Device List

The report in Figure 43 was produced by:

```
mtlib -l /dev/lmcp0 -qL (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -qL  (for AIX VTS library 1)
mtlib -l libmgrc7 -qL  (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Library Data:
   Operational state..........Paused Operational State
                              Intervention Required
   functional state...........00
   input stations.............1
   output stations............1
   input/output status........All input stations empty
                              All output stations empty
   machine type...............3494
   sequence number............10491
   number of cells............1056
   available cells............1014
   subsystems.................6
   convenience capacity.......30
   accessor config............01
   accessor status............Accessor available
                              Gripper 1 available
                              Gripper 2 installed
                              Vision system operational
   comp avail status..........Primary library manager installed.
                              Primary library manager available.
                              Secondary library manager installed.
                              Secondary library manager available.
                              Primary hard drive installed.
                              Primary hard drive available.
                              Secondary hard drive installed.
                              Secondary hard drive available.
                              Convenience input station installed.
                              Convenience input station available.
                              Convenience output station installed.
                              Convenience output station available.
   library facilities.........00
   bulk input capacity........0
   bulk input empty cells.....0
   bulk output capacity.......0
   bulk output empty cells....0
   avail 3490 cleaner cycles..0
   avail 3590 cleaner cycles..91
```

Figure 43. Library Data

The report in Figure 44 was produced by:

```
mtlib -l /dev/lmcp0 -qS (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -qS  (for AIX VTS library 1)
mtlib -l libmgrc7 -qS  (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Statistical Data:
   hour index...........10
   machine type.........003494
   model number.........L10
   manufacturer.........IBM
   plant................13
   sequence number......000000010491
   drives...............6
   mounted drives.......1
   max mounted drives...2
   min mounted drives...1
   avg mounted drives...1
   max mounted time.....22
   min mounted time.....16
   avg mounted time.....19
   pending mounts.......0
   max pending mounts...2
   min pending mounts...0
   avg pending mounts...0
   mounts/hour..........18
   index mounts/hour....0
   pre-mounts/hour......0
   max mount time.......27
   min mount time.......16
   avg mount time.......19
   pending demounts.....0
   max pending demounts.2
   min pending demounts.0
   avg pending demounts.0
   demounts/hour........16
   index demounts/hour..0
   post-demounts/hour...0
   max demount time.....28
   min demount time.....19
   avg demount time.....24
   pending ejects.......0
   max pending ejects...0
   min pending ejects...0
   avg pending ejects...0
   ejects/hour..........0
   max eject time.......0
   min eject time.......0
   avg eject time.......0
   pending audits.......0
   max pending audits...0
   min pending audits...0
   avg pending audits...0
   audits/hour..........0
   max audit time.......0
   min audit time.......0
   avg audit time.......0
   input stores/hour....0
```

Figure 44. Statistical Data

The report in Figure 45 was produced by:

```
mtlib -l /dev/lmcp0 -qI-v (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -qI -v (for AIX VTS library 1)
mtlib -l libmgrc7 -qI -v (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Performing Query Inventory Data using /dev/lmcp0
Inventory Data:
   sequence number......10491
   number of volumes....44
   inventory records
        record   1......category...........012C
                        volser.............008273
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length
        record   2......category value......FF00
                        volser.............064435
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length
        record   3......category value......FF00
                        volser.............ALTML1
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length
                        •
                        •
                        •
        record  42......category...........FF00
                        volser.............TST039
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length
        record  43......category...........FF00
                        volser.............TST182
                        volume state........Volume present in Library,
                                             but Inaccessible
        record  44......category value......FF00
                        volser.............XYZ464
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length

Note:  All available records are produced as output per request.
       Fewer records are shown here for the sake of brevity.
```

Figure 45. Inventory Query

The report in Figure 46 was produced by:

```
mtlib -l /dev/lmcp0 -qC -sFF00 -v (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -qC -sFF00 -v  (for AIX VTS library 1)
mtlib -l libmgrc7 -qC -sFF00 -v  (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Performing Query Category Inventory Data using /dev/lmcp0
Inventory by Category Data:
   sequence number......10491
   number of volumes....30
   category.............FF00
   inventory records
        record   1......category ...........FF00
                        volser..............CS2017
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length
        record   2......category ...........FF00
                        volser..............FVT896
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length
        record   3......category ...........FF00
                        volser..............IHG319
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length
                        •
                        •
                        •
        record  28......category ...........FF00
                        volser..............SLT500
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length
        record  29......category ...........FF00
                        volser..............TAF195
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length
        record  30......category ...........FF00
                        volser..............MOO801
                        volume state........00
                        logical volume......No
                        volume class........3590 1/2 inch cartridge tape
                        volume type.........HPCT 320m nominal length

Note:  All available records are produced as output per request.
       Fewer records are shown here for the sake of brevity.
```

Figure 46. Category Inventory Query

The report in Figure 47 was produced by:

```
mtlib -l /dev/lmcp0 -r -#2 -h roadster (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -r -#2 -h roadster (for AIX VTS library 1)
mtlib -l libmgrc7 -r -#2 -h roadster (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Reserved Category List:
   sequence number............10491
   system token...............roadster
   total number reserved......0002
                     category......0101
                     category......0102
```

Figure 47. Reserve Category Command

The report in Figure 48 was produced by:

```
mmtlib -l /dev/lmcp0 -qR -h roadster (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -qR -h roadster (for AIX VTS library 1)
mtlib -l libmgrc7 -qR -h roadster (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Reserved Category List:
   sequence number............10491
   system token...............roadster
   total number reserved......0002
                     category......0101
                     category......0102
```

Figure 48. Reserve Category List

The report in Figure 49 was produced by:

```
mtlib -l /dev/lmcp0 -S -S101 -NSCRATCH (for AIX physical library)
mtlib -l /dev/lmcp0 -S -S102 -NWORKING (for AIX physical library)
mtlib -l /dev/lmcp0 -qA -h roadster (for AIX physical library)
mtlib -l /dev/lmcp0/vts1 -S -S101 -NSCRATCH (for AIX VTS library 1)
mtlib -l /dev/lmcp0/vts1 -S -S102 -NWORKING (for AIX VTS library 1)
mtlib -l /dev/lmcp0/vts1 -qA -h roadster (for AIX VTS library 1)
mtlib -l libmgrc7 -S -S101 -NSCRATCH (for Sun, HP, Linux, SGI, Tru64, and Windows)
mtlib -l libmgrc7 -S -S102 -NWORKING (for Sun, HP, Linux, SGI, Tru64, and Windows)
mtlib -l libmgrc7 -qA -h roadster (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Category Attribute List:
   sequence number............10491

   system token...............roadster
                     category 0101 —— name: SCRATCH
                     category 0102 —— name: WORKING
```

Figure 49. Category Attribute List

The report in Figure 50 was produced by:

```
mtlib -l /dev/lmcp0 -A (for AIX physical library)
mtlib -l tire -A (for Sun, HP, Linux, SGI, Tru64, and Windows)
```

```
Library Address Information:
   library name...............tire
   host identification........roadster
   primary address............9.115.45.52
   primary status.............Online
   alternate address..........9.115.45.51
   alternate status...........Offline
```

Figure 50. Library Address Information

MTEVENT Program

The *mtevent* program is a command-line interface to the MTIOCLEW command.

The *mtevent* program has the following flags:

Flag	Description
-l[filename]	Library special file name or logical name of the library, for example, */dev/lmcp0*, */dev/lmcp0/vts1*, or *libmgrc7*
-t[timeout]	Number of seconds to wait for the event to occur (0=no timeout)

Notes:

1. The *-l* flag is required.
2. If the *-t* flag is not supplied, then no timeout is performed.

Library Driver Information

The *lmcpd* communicates to the tape library through symbolic names defined in the */etc/ibmatl.conf* file. One or more symbolic names can be configured for each tape library online to the system. A symbolic name is used to perform the various library functions (such as mounting and demounting volumes).

Software Interface

The C object module provides three subroutines for communicating with the IBM TotalStorage 3494 Enterprise Tape Library. These subroutines are *open_ibmatl*, *close_ibmatl*, and *ioctl_ibmatl*. The *open_ibmatl* and *close_ibmatl* routines are used to open and close communication with the library (as the *open* and *close* system calls are used to open and close communication with a file). The *ioctl_ibmatl* subroutine is used to send commands to the Library Manager on the 3494 Tape Library.

To send commands to the 3494 Tape Library, one symbolic name must be defined for use on the library.

Library Manager Event Notification

In addition to performing library operations, the *lmcpd* is responsible for receiving the various Library Manager notifications. The daemon monitors several types of events. When the daemon receives an event, it checks a list of processes waiting for an event to determine where to deliver it. If no process is waiting for the event that has arrived, then the event is discarded. The applications can use the Library Event Wait call to request notification of all Library Manager events. Refer to the *IBM Tape Device Drivers: Programming Reference* for more information.

Synchronous and Asynchronous Operations

Two types of library operations are supported: synchronous and asynchronous. The Library Manager responds to the synchronous operation immediately. These operations are completed when the library responds to the request. An example of a synchronous operation is the Library Query call. An asynchronous operation (such as a mount operation) takes longer to complete. In an asynchronous operation, the library returns an initial response when the command is accepted for execution. When the command is completed, it returns a delayed response message indicating the status of the operation.

Operation Complete Notification

When the *Mount*, *Demount*, *Audit*, or *Set Volume* category operation is sent to the library, a message identifier is assigned to the operation. This identifier is returned to *lmcpd*. When the operation is completed, an operation complete notification is sent to the daemon. The daemon makes a determination (based on the initial message identifier) as to which user process to notify that the operation is complete. The *Mount*, *Demount*, *Audit*, or *Set Volume* category operations have an option that allows the user process to wait or not wait for the operation complete message. If the user process does not wait for the final complete, the initial message identifier is returned to the caller, and it can be used in subsequent Library Query Message ID operations to solicit the status of the asynchronous operation.

Unsolicited Notification

There are situations when the Library Manager has a condition to report that is not related to any I/O operation. In this case, an unsolicited notification is sent to the device daemon. If a notification is received, then any process waiting with the Library Event Wait call is notified. An example of an unsolicited notification is the operational state change of the library (from the Auto mode to the Pause mode). Refer to the *IBM Tape Device Drivers: Programming Reference* for a list of unsolicited notifications received by the *lmcpd*.

Driver Message Queue

The driver maintains a message queue for any process that has an open LMCP file descriptor. This queue, which has a depth of four entries, is implemented on a first-in, first-out basis. The purpose of this message is to reduce the possibility of missing a message when several messages occur in a short period of time. All queue entries are discarded when the LMCP file descriptor is closed. When an application issues the MTIOCLEW input/output control *(ioctl)* call, the driver returns the oldest entry from the message queue and deletes it. If no entries are in the queue, then the calling process is put to sleep until a message is received.

Volume Categories

To facilitate the management of the tape volumes within the 3494 Tape Library, the capability to associate the tape volumes into logical groupings is provided. Each logical grouping is known as a category. For example, an installation can have one or more scratch categories that are assigned by media type or class of user. Another potential use is managing volumes for daily, weekly, and monthly backups or volumes owned by a specific user. The *lmcpd* also allows a user process to assign a particular category to a tape device in the library. This process allows all the volumes associated with the category to be mounted as determined by the flags of the command on the specified device.

A category is a four digit hexadecimal number in the X'0000'–X'FFFF' range. The assigned categories are:

X'0000' NULL category (not usable or not assigned)

X'0001'–X'FEFF' General programming use

X'FFFF' VOLSER specific

The X'FF00'–X'FFFE' categories are reserved for hardware functions:

X'FF00' Insert

X'FF01'–X'FF0F' Reserved

X'FF10' Eject

X'FF11' Bulk eject

X'FF12'–X'FFF8' Reserved

X'FFF9' Service volume

X'FFFA' Manually ejected

X'FFFB' Purge volume

X'FFFE' Cleaner volume

The following three categories are available for programming use:

Insert When a tape volume is received in the input station, the volume label is read and assigned to this category. This category cannot be assigned by a user process.

Eject The volumes assigned to this category are moved from their cell into the output station. After the volume is delivered, it is removed from the inventory. The Library Set Volume Category command is the only command that can change the category of a volume to the eject category.

Bulk eject The volumes that are assigned to this category are moved from their current cell to the bulk output area in the 3494 Tape Library. Refer to the *IBM TotalStorage Enterprise Automated Tape Library Operator Guide* for more information about the bulk output area. The Library Set Volume Category command is the only command that can change the category of a volume to the bulk eject category.

The X'FFFF' category is available for general programming use, except that any mount request to this category must apply to a specific volume assigned to the category (and not based on the category alone).

IBM TotalStorage Virtual Tape Server Subsystem Attachment

The device driver supports the IBM TotalStorage Virtual Tape Server (VTS) subsystem attachment with a 3494 Enterprise Tape Library on AIX, HP-UX, Windows, and Sun/Solaris systems. An attached VTS subsystem provides virtual tape drives that are accessible with the VTS SCSI attachment feature. Volumes within the VTS subsystem are logical tape volumes. A logical tape volume appears to the host as a normal physical tape volume, except that the VTS logical volumes are unique to the VTS subsystem and are accessible only by virtual tape drives within the VTS subsystem.

Access to a VTS subsystem is specified to the device driver as a logical library within the attached Tape Library as *vts1*, *vts2*, and so on. For example, issuing a Query Inventory command to the *3494 lib* returns the physical volumes in the 3494 Tape Library, where *3494 lib* is a library configured in the */etc/ibmat1.conf* file. Issuing a Query Inventory command to *3494 lib/vts1* returns the logical volumes in the first VTS subsystem attached to the 3494 Enterprise Tape Library.

AIX

The software consists of an Automated Tape Library Device Driver (*atldd*), an application daemon (*lmcpd*) that communicates with the 3494 Library Manager (LM), and a utility program (mtlib), which provides a command-line interface to the library. Either an Ethernet or Token-Ring LAN or an RS-232 serial connection can be used for the physical connection between the host and the 3494 LM.

The daemon, known to AIX specifically as the IBM Library Manager Control Point Daemon (*lmcpd)*, communicates with tape libraries defined in a configuration file. This file, *ibmatl.conf*, is placed in the */etc* directory when the software package is installed. Customization of this file is described in "Defining Tape Libraries to the Daemon" on page 257. A typical environment for the AIX 3494 Enterprise Library Driver is an RS/6000 or System p (also known as pSeries) server on a local area network (LAN) with SCSI- or fibre-attached tape drives providing backup and restore functions.

Data Flow

The software described in this chapter covers the library device driver and the interface between the application and the library device driver. For more information refer to Figure 51.

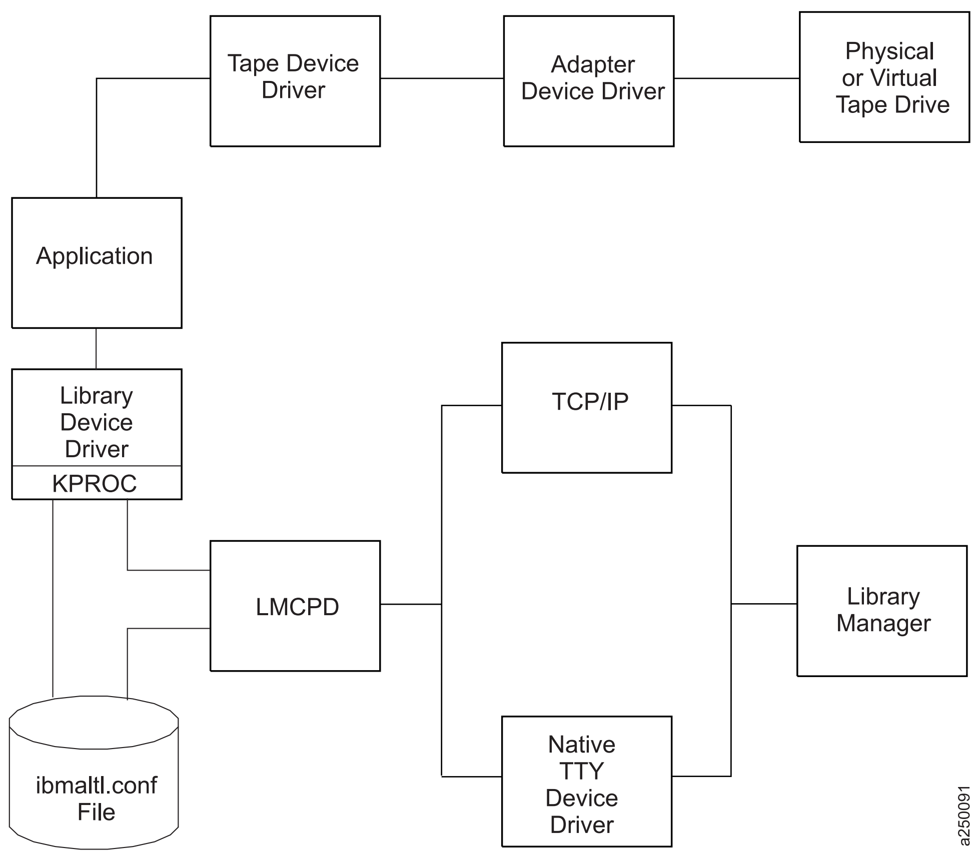


Figure 51. Data Flow for AIX 3494 Enterprise Tape Library Driver

Product Requirements

The following hardware and software components are supported by IBM.

Hardware Requirements

The library driver supports the following hardware:

- 3494 Enterprise Tape Library with SCSI-attached IBM 3490E (Model C1A, C2A, or F1A), 3590, and 3592 tape drives
- One of the following options, depending on the connection (RS-232 or LAN) required for the Library Manager:
  - RS-232:

    One standard 25 pin null modem D-shell RS-232 cable

    (maximum of 15.24 meters [50 feet])

    IBM TotalStorage Virtual Tape Server (models B10, B18, and B20)
  - LAN:
    - Token Ring or Ethernet Adapter support in an RS/6000 or pSeries workstation attached to the Library Manager
    - LAN cabling (as required)
- IBM System Storage TS1120 Tape Drive Model EO5
- IBM Virtualization Engine TS7510

Software Requirements

The following software is required for the library driver:

- AIX 5L, Versions 5.1 and 5.2, and later releases on IBM POWER-based servers.
- AIX Tape and Medium Changer Device Driver (Atape)

To obtain the most current service and documentation for this software, refer to Appendix A, “Accessing Documentation and Software Online,” on page 305

Installation Instructions

Use the standard set of AIX methods to install and configure the LMCP devices. You must have *root* authority to perform these operations.

Installation Procedure

Refer to Appendix A, “Accessing Documentation and Software Online,” on page 305 for information on obtaining the latest versions of the device driver and the latest documentation.

Enter the following command to list the currently installed version of the 3494 Enterprise Library Driver:

lslpp -l atldd.driver

Connecting the IBM TotalStorage Enterprise Tape Library

If a TTY connection is needed, use a standard 25 pin null modem D-shell RS-232 cable to connect the workstation to the 3494 Enterprise Tape Library. Place the cable on the native serial port or on the eight port or 16 port asynchronous adapter.

If a LAN connection is needed, connect the LAN cable from the Token Ring or Ethernet adapter card in the workstation to the 3494 Tape Library.

Note: The cable (RS-232 or LAN) must be in place before the software is installed to ensure proper initialization of the driver.

Configuring the Serial Port

After the 3494 Tape Library is connected to the workstation, you must configure the serial port to which it is connected. Use *smit* to configure the serial port. Enter the following command:

```
smit tty
```

Select the following options:

1. TTY Menu: Add a tty device.
2. Selection Menu: Select the appropriate parent serial adapter.
3. Add TTY Menu: Enter the port number.

   Press F4 to generate a list of possible values.
4. Add TTY Menu: Set baud rate to 9600.
5. Add TTY Menu: eight data bits, one stop bit, no parity.
6. Add TTY Menu: Set Enable program to Off.

   Press F4 to generate a list of possible values.
7. Add TTY Menu: Set Enable LOGIN to **disable**.

   Use the Tab key to toggle the value.
8. Press the Return key to configure the *tty* device.

   Repeat steps 1 through 8 for each tape library connected to the workstation.

Configuring the IBM 3490E or TotalStorage 3590 Tape Drive

To operate the 3494 Enterprise Tape Library successfully through the configured LMCPs, you must configure all 3490E or 3590 tape drives in the 3494 Tape Library. Refer to "Configuring the Library Manager Control Point" on page 258 for instructions.

Defining Tape Libraries to the Daemon

After the software is installed and the desired tape libraries are connected to the system, the */etc/ibmatl.conf* file must be edited to define the tape libraries to the *lmcpd*. The format of the file is:

```
Symbolic name     Connection type     Identifier     Alternate LAN Connection
```

Symbolic name is used to identify each library. The symbolic names must be unique across all libraries defined in the */etc/ibmatl.conf* file.

Connection type is used to define the type of connection to the library (either RS-232 or TCP/IP). For RS-232 connections, this type is the device special file name of the *tty* device (for example, */dev/tty0* or */dev/tty1*). For TCP/IP connections, this type is the IP address of the Library Manager. If the Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first IP address.

Note: Ensure that duplicate IP addresses are not used in this file or unpredictable results can occur.

Identifier is used to specify a name by which the Library Manager identifies the host machine. This identifier has a maximum of eight characters. The host name of the workstation is usually the best name to use (although it is not necessary). This parameter is used only to identify a particular host to the 3494 Enterprise Library Manager.

Alternate LAN Connection is used to specify an alternate IP address for a 3494 High Availability LAN attached configuration. If the High Availability Library has a dual

LAN card installed, the second IP address of the dual LAN card should be entered after the first alternate LAN connection IP address.

Note: For RS-232 attachments, the alternate LAN connection is not applicable. Failover for RS-232 connections proceeds over the single serial line.

The following examples show how to define the library name:

```
libmgrc7        /dev/tty0               mercury
```

This stanza defines the *libmgrc7* library connected by */dev/tty0* to the workstation. The library uses the *mercury* identifier for the host.

```
libmgrc8        9.115.32.21             jupiter
```

This stanza defines the *libmgrc8* library connected through TCP/IP to the workstation. The address of the Library Manager on the 3494 Tape Library is *9.115.32.21*. The library uses the *jupiter* identifier for the host.

```
libmgrc9        9.115.46.15             telos        9.115.46.17
```

This stanza defines the *libmgrc9* High Availability library connected through TCP/IP to the workstation. The IP addresses of the dual Library Managers are *9.115.46.15* and *9.115.46.17*.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury
```

This stanza defines libmgrc7 with a dual LAN card connected through TCP/IP to the workstation. The first address of the Library Manager on the 3494 Tape Library is 9.115.32.21 and the second address is 9.115.32.22. The library uses the mercury identifier for the host. The following stanza defines the libmgrc7 High Availability library with dual LAN cards connected through TCP/IP to the workstation. The IP addresses of the first Library Manager are 9.115.32.21 and 9.115.32.22. The IP addresses of the second Library Manager are 9.115.26.13 and 9.115.26.14. The library uses the mercury identifier for the host.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury  9.115.26.13   9.115.26.14
```

Configuring the Library Manager Control Point

After the driver is installed, a tape library is connected to the workstation, and the serial port is configured, you can configure the instances of the LMCP. Perform the following procedure to define and configure the LMCP:

1. Enter the following command:

   ```
   smit
   ```

   The system management interface tool (*smit*) main menu is displayed (refer to Figure 52).

```
                              System Management

Move cursor to desired item and press Enter.

  Software Installation and Maintenance
  Devices
  Physical & Logical Storage
  Security & Users
  Diskless Workstation Management & Installation
  Communications Applications and Services
  Spooler (Print Jobs)
  Problem Determination
  Performance & Resource Scheduling
  System Environments
  Processes & Subsystems
  Applications
  Using SMIT (information only)


F1=Help            F2=Refresh         F3=Cancel          F8=Image
F9=Shell           F10=Exit           Enter=Do
```

Figure 52. SMIT Main Menu

2. Select **Devices** (item 2) from the *smit* main menu.

   Use the up (↑) arrow and down (↓) arrow keys to move the cursor to the desired item on the menu.

   Press **Enter** or click **Do** if you are processing *smit* in a windowed environment.
3. Select **Tape Drive** from the Devices menu (refer to Figure 53).

```
                                   Devices

Move cursor to desired item and press Enter.

  Configure Devices Added after IPL
  Printer/Plotter
  TTY
  Asynchronous Adapters
  PTY
  Console
  Fixed Disk
  CD ROM Drive
  Optical Disk Drive
  Diskette Drive
  Tape Drive
  •
  •
  •
```

Figure 53. SMIT Devices Menu

The Tape Drive menu is displayed (refer to Figure 54).

```
                                Tape Drive

Move cursor to desired item and press Enter.

  List All Defined Tape Drives
  List All Supported Tape Drives
  Add a Tape Drive
  Change / Show Characteristics of a Tape Drive
  Remove a Tape Drive
  Configure a Defined Tape Drive
  Generate Error Report
  Trace a Tape Drive
```

Figure 54. SMIT Tape Drive Menu

4. Select **Add a Tape Drive** from the Tape Drive menu (Figure 54). The Tape Drive Type menu is displayed (refer to Figure 55).
5. Select the Library Manager control point from the Tape Drive Type menu (refer to Figure 55).

```
                             Tape Drive Type

    Move cursor to desired item and press Enter.

      8mm   scsi     2.3 GB 8 mm Tape Drive
      9trk  scsi     1/2 Inch 9 Track Tape Drive
      150mb scsi     150 MB 1/4 Inch Tape Drive
      ost   scsi     Other SCSI Tape Drive
      drive 3420     3420 Tape Drive
      drive 3480     3480 Tape Drive
      drive 3490     3490 Tape Drive
      atl   library  LAN/TTY Library Manager Control Point
```

Figure 55. Tape Drive Type Menu

The Add an LMCP Logical Device menu is displayed (refer to Figure 56).

The Logical Name field for the Library Manager control point is optional. If the field remains blank, a unique name is assigned automatically. The assigned name is *lmcpn*, where *n* is a number that indicates the LMCP entry and starts with zero for the first LMCP defined.

The Library Name field must be one of the library names defined in the */etc/ibmatl.conf* file. Refer to "Defining Tape Libraries to the Daemon" on page 257 for a description of this file.

The Command *Timeout in Minutes* field is optional and is used to specify the maximum amount of time an application waits for commands to complete, including *mount* and *demount* commands. The default is to wait forever until the command completes.

```
                        Add an LMCP Logical Device

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                  [Entry Fields]
  Logical Name of LMCP (optional)               []
  Library Name (F4 to list library names)       [libmgrc7]          +
  Command Timeout in Minutes                    []
```

Figure 56. Add an LMCP Logical Device Menu

6. The COMMAND STATUS window opens with the **Command: OK** status. The logical name of the Library Manager control point is displayed as **Defined** (refer to Figure 57).

```
                                  COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

lmcp0   Defined
```

Figure 57. LMCP COMMAND STATUS Window

7. Repeat steps 4 through 6 to define any additional Library Manager control points.
8. Return to the *smit* Tape Drive menu (refer to Figure 54).
9. Select **Configure a Defined Tape Drive** from the *smit* Tape Drive menu. A list of LMCP devices is displayed (refer to Figure 58).

```
                                    Tape Drive

    Move cursor to desired item and press Enter.

      rmt0     Available 00-00-0S-50 2.3 GB 8 mm Tape Drive
      rmt1     Defined   00-01-01-20 IBM SSD 3490E Tape Drive
      lmcp0    Defined               LAN/TTY Library Manager Control Point

    F1=Help                F2=Refresh             F3=Cancel
    F8=Image               F10=Exit               Enter=Do
```

Figure 58. Configure an LMCP Selection List

10. Select a defined Library Manager control point to configure from the LMCP Selection list.

    The COMMAND STATUS window opens with the **Command: OK** status, and the Library Manager control point is shown as **Available** (refer to Figure 59).

```
                                  COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

lmcp0  Available
```

Figure 59. Configure a Defined LMCP COMMAND STATUS Window

11. Cancel the COMMAND STATUS window by pressing F3. The LMCP Selection window opens again.
12. Continue selecting the Library Manager control points for configuration until all the devices are configured.
13. Exit from the *smit* menu.

    You have completed the configuration of Library Manager control points to the system.

Loading the Daemon

The *lmcpd* is loaded during the system initialization. During the initial installation of the driver, you must load the daemon manually using the following command:

```
cfgmgr
```

You can access the 3494 Enterprise Tape Library after the above command is executed.

To verify that the daemon is loaded correctly, enter the following command:

```
ps -efa | grep lmcpd
```

You can see multiple instances of the daemon running (which is typical), but only one instance has a parent process ID of 1.

Deconfiguring the Library Manager Control Point

In the following examples, replace the letter *n* with the appropriate number for the chosen device.

Deconfigure the LMCP device using one of the following procedures:

1. The first method deconfigures the device but leaves the device defined in the configuration database. It is similar to taking the device offline.

   Enter the following command to deconfigure the */dev/lmcpn* device but leave it defined in the device database:

   ```
   rmdev -l lmcpn
   ```
2. The second method takes the device offline and removes the device definition from the device database.

   Enter the following command:

   ```
   rmdev -l lmcpn -d
   ```

   The device driver modules are not unloaded from the kernel until the last tape device is deconfigured.

Uninstall Procedure

All devices using the *atldd* driver must be closed and not in use when *atldd* is uninstalled or the uninstall fails.

You can uninstall the *atldd* using the *smit* command menu to uninstall software and selecting *atldd.driver* or use the following *installp* command:

```
installp -u atldd.driver
```

Special Files

After the driver is installed and a Library Manager control point is configured and made available for use, access is provided through the special files. These special files are in the */dev* directory. Each instance of an LMCP has exactly one special file (for example, */dev/lmcp0*) associated with it.

Problem Determination

A set of tools is provided with the device driver to determine if it is functioning correctly. The standard AIX interface is provided for problem determination.

Error Logging

The driver provides logging to the system error log for various errors. View the error log using the *smit* or the *errpt* command. The error templates follow the same form as the default AIX error log entries.

AIX Device Driver Trace Facility

The AIX trace facility is supported for the device driver. The trace event is identified with a *hookword*. The *hookword* used by the device driver is 426. The trace can be initiated at any time before an operation on a tape device.

Enter the following command to start the trace:

```
trace -a -j 426
```

This command starts the trace in the background and collects only the trace events with the 426 *hookword*.

Enter the following command to stop the trace:

```
trcstop
```

This command stops the trace after the tape operations are performed.

Enter the following command to view the trace:

```
trcrpt > lmcp.trace.out
```

This command formats the trace output into a readable form and places it in a file for viewing. The */etc/lmcp.trcfmt* file is installed into */etc* during installation. It provides the formatting statements needed by *trcrpt*.

Daemon Trace Facility

The following trace facility is available for the *lmcp* daemon and device driver:

```
trcatl -[ald]
```

Arguments:

```
-a              (trace all libraries defined in /etc/ibmatl.conf)
-l <name>       (name of library)
-d              (delete all log files; requires root authority)
```

Examples:

```
trcatl -a >/tmp/trace-out
trcatl -l libmgrc7
```

Note: The *-a* argument takes precedence over the *-l* argument.

HP-UX

The software consists of a daemon that communicates directly with the Library Manager of the 3494 Enterprise Tape Library through Ethernet or Token Ring LAN, a utility program that provides a command-line interface to the daemon and a C object module that can be linked with user applications to provide a communication interface with the daemon.

Note: In the HP-UX operating system, this is a program that runs unattended in the background to perform a standard service. Some daemons are triggered automatically to perform their task, others operate periodically.

This software is known as a *driver* throughout this document because it provides the software and interface necessary to *drive* the 3494 Tape Library. However, the product does not consist of a *device driver* in the true sense of the term. That is, it is not an extension of the operating system kernel, such as a SCSI tape device driver.

A typical environment for the HP-UX Automated Tape Library Driver is an HP workstation that acts as a data server on a network with SCSI tape devices providing backup or restore and data server functions.

The Library Manager control point daemon (*lmcpd*) is provided in the package. The *lmcpd* is a process that is always running on the system. It provides direct communication with the 3494 Enterprise Library Manager. An application links with the supplied C object module using the interface described in the *IBM Tape Device Drivers: Programming Reference*. The subroutines in this module communicate with the *lmcpd* to perform the various library operations using standard UNIX namespace sockets.

The *lmcpd* communicates with the Library Manager through TCP/IP. The */etc/ibmatl.conf* configuration file is used to define the attachment for each library. Refer to “Defining the Symbolic Name to the lmcpd” on page 280 for more information.

Data Flow

The software described in this chapter covers the library driver. The interface between the application and the library driver is described in the *IBM Tape Device Drivers: Programming Reference*. Figure 61 on page 272 illustrates the data flow.

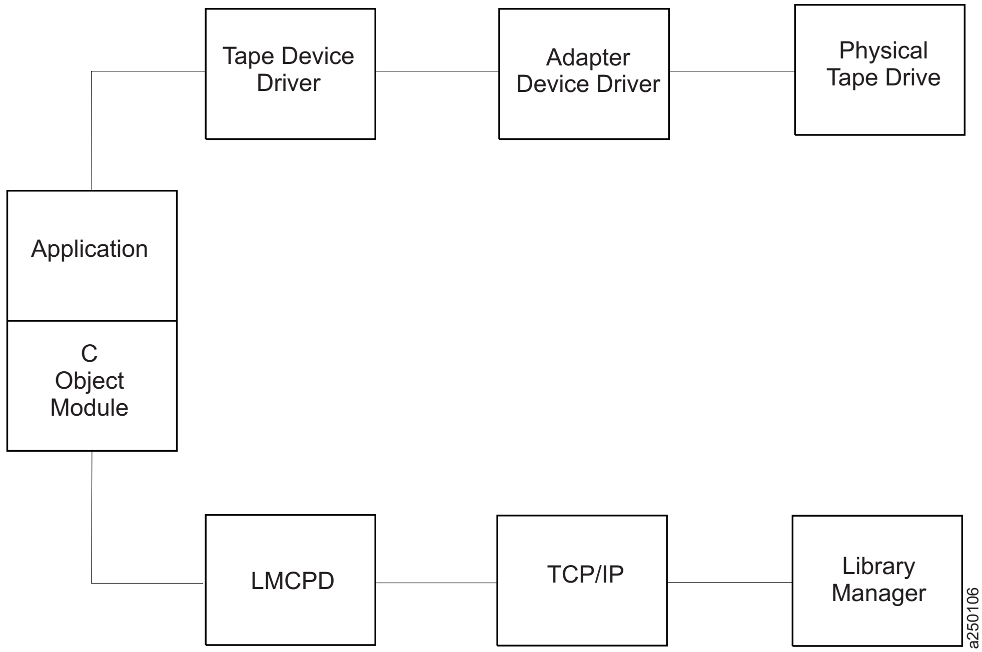


Figure 60. Data Flow for HP-UX 3494 Enterprise Tape Library Driver

Product Requirements

The following software and hardware are required to use this product.

Hardware Requirements

The following hardware is required for the library driver:

- 3494 Enterprise Tape Library with 3590 drives in HP-UX 11.0, 11i v1 and v2 and 3592 drives in 11.0, 11i v1, v2 and v3.
- IBM Virtualization Engine TS7510 and TS7520 in HP-UX 11.0, 11i v1 and v2
- Enterprise Model B10, B18, and B20 Virtual Tape Server (direct attached only)
- One of the following options depending on which LAN connection is used for the Enterprise Library Manager:
  - Token-Ring Attach:

    Enterprise FC 5219 (Token-Ring Adapter)

    Token-Ring adapter card for HP workstation

    Token-Ring cables (as required)
  - Ethernet Attach:

    Enterprise FC 5220 (Ethernet Adapter)

    Ethernet port or adapter for HP workstation

    Ethernet cables (as required)

Software Requirements

The following software is required for the library driver:

- HP-UX Version 10.20, 11.0, 11i v1, v2, and v3
- IBM Tape and Medium Changer Device Driver for HP-UX

To obtain the most current service and documentation for this software, refer to Appendix A, “Accessing Documentation and Software Online,” on page 305.

Software Compatibility

The following *optional* software is supported by the library driver:

IBM Tivoli Distributed Storage Manager for HP-UX

Installation Instructions

This chapter describes how to install, configure, and uninstall the HP-UX Automated Tape Library Support for the 3494 Enterprise Tape Library. The IBM Tape Library driver for HP-UX is installed using the standard *swinstall* process. The IBM Tape Library Driver for HP-UX is the *lmcpd* package.

Installation Overview

LAN support must exist on the machine before installing the IBM Tape Library driver for HP-UX. Install and configure the LAN support before proceeding with the installation of the *lmcpd* package. For more information, refer to the HP-UX documentation appropriate for the LAN support that you are using.

1. Copy the software from the distribution medium to the depot.
2. Check the README file and verify that your system is configured appropriately for installing the *lmcpd* software.
3. Install and configure the software.

You must have *root* authority to perform this installation procedure.

The distribution diskette contains a *swinstall* process that includes the *lmcpd* daemon with other associated files and utilities.

Command Sequence Information

Please take note of the following facts about the command sequences described in this section:

- In some of the examples, filenames given on the command line must be referenced with an absolute path. Using *'pwd'/filename* to reference a file instead of *filename* ensures this.
- All the SD commands (for example, *swinstall, swcopy*) can be run initially with the *'-p'* flag to preview the command. After observing the preview output, you can reissue the command without the *'-p'* flag to perform the actual operation.
- The SD commands are moderately complex scripts that usually proceed in several steps. The steps are typically *Selection*, *Analysis,* and *Execution*. Each step may produce useful information and error messages, so it is a good idea to observe carefully the results of the installation process as it occurs.

If you run into unexpected results during the installation, check the associated log file.

Note: If an earlier version of the product is installed on the machine, execute the uninstall procedure before you enter the following commands. Refer to "Uninstall Procedure" on page 282 for more information.

The following files are installed on the system:

- */etc/lmcpd* (Library Manager control point daemon)
- */etc/ibmatl.conf* (configuration file)
- */usr/lib/libibm.o* (32 bit application interface object module)
- */usr/lib/libibm64.o* (64 bit application interface object module)
- */usr/lib/libibmz.0* (32 bit application interface object module with +z option)

- */usr/lib/libibm64z.0* (64 bit application interface object module with +Z option)
- */usr/lib/libibm_ia64.0* (64 bit application interface object module with +z option for Itanium system)
- */usr/lib/libibm_ia64z.0* (64 bit application interface object module with +Z option for Itanium system)
- */usr/include/sys/mtlibio.h* (application interface header file)
- */usr/bin/mtlib* (tape library driver utility program)
- */usr/bin/intlib64* (64 bit tape library driver utility program)
- */usr/bin/mtevent* (tape library driver utility program)
- */usr/bin/trcatl* (tape library driver diagnostic program)
- */opt/lmcpd* (subdirectory)
- */opt/lmcpd/ibmatl.conf* (backup configuration file)

Note: If this is an update of the *lmcpd* package, the existing */etc/ibmatl.conf* file is preserved. It is not overwritten. A copy of *ibmatl.conf* is also installed in the */opt/lmcpd* directory.

Install the Product Manually

Installing the product manually requires two steps, detailed in the following sections:

1. "Copy the Software to the Software Depot" on page 267
2. "Install the Product" on page 267

Copy the Software to the Software Depot

Attention: If you do not copy the *lmcpd* software into a depot, you cannot easily uninstall the software.

Copy the appropriate driver to the Software Depot. For example:

 # swcopy -p -s /driver_location/lmcpd.hpux.x.x.x.x lmcpd (preview option)
 # swcopy -s /driver_location/lmcpd.hpux.x.x.x.x lmcpd

Use *swlist* to verify that the *lmcpd* software is in the depot:

 # swlist -d lmcpd

Install the Product

When the software is in the depot, the *lmcpd* software can be installed to the root file system using the HP-UX *swinstall* command.

The following commands install *lmcpd* from the depot to the default root file system:

 # swinstall -p lmcpd (preview option)
 # swinstall lmcpd

Use *swlist* to list the software installed on the default root file system as follows:

 # swlist lmcpd

Verify correct installation to the default root file system with the *swverify* command:

 # swverify lmcpd

Connecting the IBM TotalStorage 3494 Enterprise Automated Tape Library

You can use a LAN connection (either Ethernet or Token Ring) through a TCP/IP connection. The connection must be in place before the software is installed to ensure proper initialization of the daemon. For each 3494 Enterprise Tape Library connected through TCP/IP, ensure that your machine has access to the Library Manager on the 3494 Enterprise Tape Library. Use the *ping* utility to verify that you have network connectivity to the Library Manager. Consult your network administrator if you need help with this task.

Defining the Library Device to LMCPD

After the software is installed and all desired tape libraries are connected to the system, the */etc/ibmatl.conf* file must be edited to define the library devices to the *lmcp* daemon. The format of the file is:

```
Symbolic name     Connection type     Identifier     Alternate LAN Connection
```

Symbolic name is used to identify each tape library. The symbolic names must be unique across all libraries defined in the ibmatl.conf file.

Connection type defines the type of connection to the library. For TCP/IP connections, this type is the IP address of the 3494 Enterprise Library Manager. If the Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first IP address.

Identifier is used to specify a name by which the Library Manager identifies the host machine. This identifier has a maximum length of eight characters. The symbolic network name of the host is usually the best name to use, although any name is acceptable. This parameter is used only to identify a particular host to the 3494 Library Manager.

Alternate LAN Connection is used to specify an alternate IP address for an Enterprise High Availability LAN configuration. If the High Availability Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first alternate LAN connection IP address.

The following examples show how to define the library name:

```
3494a            9.115.32.21        jupiter
```

This stanza defines the *3494a* library connected to the host. The address of the Enterprise Library Manager is *9.115.32.21*. The tape library uses *jupiter* as the identifier for the host.

```
libmgrc9         9.115.46.15          telos         9.115.46.17
```

This stanza defines the *libmgrc9* High Availability library connected through TCP/IP to the workstation. The IP addresses of the dual Library Managers are *9.115.46.15* and *9.115.46.17*.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury
```

This stanza defines libmgrc7 with a dual LAN card connected through TCP/IP to the workstation. The first address of the Library Manager on the 3494 Tape Library is 9.115.32.21 and the second address is 9.115.32.22. The library uses the mercury identifier for the host. The following stanza defines the libmgrc7 High Availability library with dual LAN cards connected through TCP/IP to the workstation. The IP addresses of the first Library Manager are 9.115.32.21 and 9.115.32.22. The IP addresses of the second Library Manager are 9.115.26.13 and 9.115.26.14. The library uses the mercury identifier for the host.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury  9.115.26.13   9.115.26.14
```

Adding or Deleting Entries in the Library Device

You can add or delete entries in */etc/ibmatl.conf* at any time. However, changes do not take effect until the daemon is started again. Use the UNIX *kill* command (*kill -kill* <*pid*>) to kill the *lmcpd* process, then enter the */etc/lmcpd* command to start the daemon again.

Ensure that the library activity on all of the libraries is completed before starting the *lmcpd* again.

Uninstall Procedure

Attention: Do not try to uninstall the *lmcpd* software by deleting the files that make up the *lmcpd* fileset. It is best to use the *swremove* command.

Halt all library activity before starting the uninstall procedure. *Kill* the *lmcpd* daemon if it is currently executing.

To remove the *lmcpd* software from the root file system enter:

 # swremove -p lmcpd
 # swremove lmcpd

Other Administrative Tasks

To determine what versions of the *lmcpd* software are currently installed on the default root file system:

 # swlist -a state lmcpd

To determine what versions of the *lmcpd* software are stored in the default depot:

 # swlist -d state lmcpd

To view the set of files installed with the *lmcpd* software:

 # swlist -l file lmcpd

To remove the *lmcpd* software from the depot:

 # swremove -d lmcpd

If more than one level of *lmcpd* exists in the depot, explicitly specify the level to remove it. For example:

 # swremove -d lmcpd,r=4.1.8.0

Problem Determination

A set of tools is provided with the software to determine if the *lmcpd* is functioning correctly.

Error Logging

The *lmcpd* uses the *syslog* facility to log the errors. Errors are logged according to the */etc/syslog.conf* file. The *lmcpd* uses the daemon facility for logging errors. Only errors are logged with this facility. For more information about using *syslog,* refer to your system administration manuals.

Daemon Trace Facility

The following trace facility is available for the *lmcpd* daemon:

```
trcatl -[ald]
```

Arguments:

```
-a              (trace all libraries defined in /etc/ibmatl.conf)
-l <name>       (name of library)
-d              (delete all log files; requires root authority)
```

Examples:

```
trcatl -a >/tmp/trace-out
trcatl -l libmgrc7
```

Note: The *-a* argument takes precedence over the *-l* argument.

Linux

The software consists of a daemon that communicates directly with the Library Manager of the 3494 Enterprise Tape Library through Ethernet or Token Ring LAN, a utility program that provides a command-line interface to the daemon, and a C object module that can be linked with user applications to provide a communication interface with the daemon.

Note: In the Linux operating system, this is a program that runs unattended, in the background, to perform a standard service. Some daemons are triggered automatically to perform their task; others operate periodically.

This software is known as a *driver* throughout this document because it provides the software and interface necessary to *drive* the 3494 Tape Library. However, the product does not include a *device driver* in the true sense of the term. That is, it is not an extension of the operating system kernel, such as a tape device driver.

A typical environment for the Linux Automated Tape Library Driver is a Linux workstation that acts as a data server on a network with tape devices providing backup or restore and data server functions.

The Library Manager control point daemon (*lmcpd*) is provided in the package. The *lmcpd* is a process that is always running on the system. It provides direct communication with the Library Manager. An application links with the supplied C object module using the interface described in the *IBM Tape Device Drivers: Programming Reference*. The subroutines in this module communicate with the *lmcpd* to perform the various library operations using standard UNIX namespace sockets.

The *lmcpd* communicates with the Library Manager through TCP/IP. The */etc/ibmatl.conf* configuration file is used to define the attachment for each library. Refer to “Defining the Symbolic Name to the lmcpd” on page 280 for more information.

Data Flow

The software described in this chapter covers the library driver. The interface between the application and the library driver is described in the *IBM Tape Device Drivers: Programming Reference*. Figure 61 on page 272 illustrates the data flow.

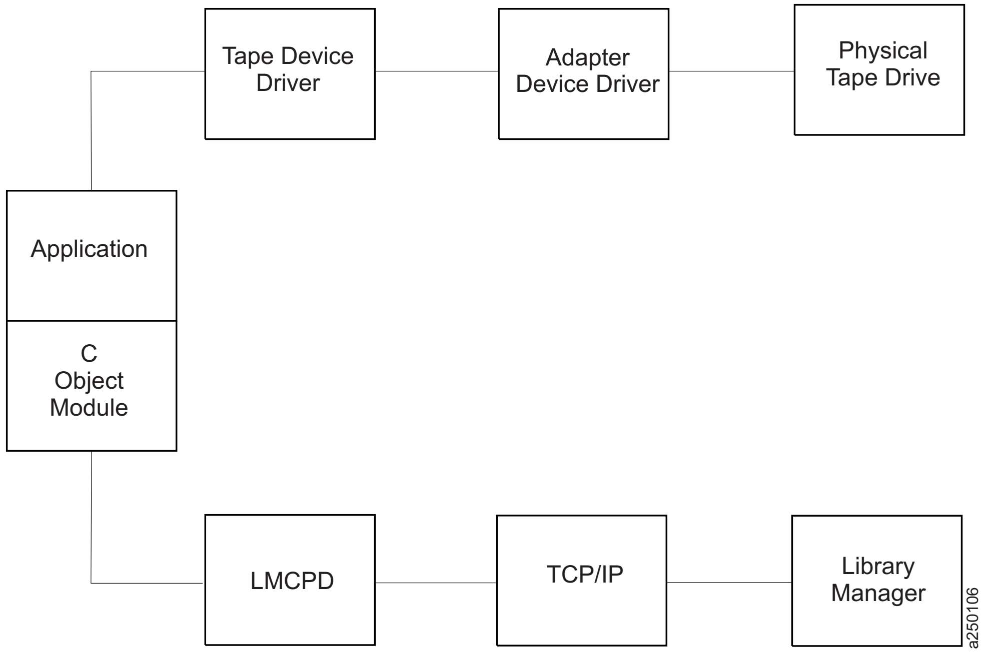


Figure 61. Data Flow for Linux 3494 Enterprise Tape Library Driver

Product Requirements

The following hardware and software components are required and supported by IBM.

Hardware Requirements

The following hardware is required for the library driver:

- IBM TotalStorage 3494 Enterprise Tape Library with IBM 3590 and 3592 drives with the Fibre Channel Attachment
- IBM System Storage TS1120 Tape Drive
- IBM Virtualization Engine TS7510
- One of the following options, depending on which LAN connection is used for the Enterprise Library Manager:
  - Token-Ring Attach:

    Enterprise FC 5219 (Token-Ring Adapter)

    Token-Ring adapter card for Linux workstation

    Token-Ring cables (as required)
  - Ethernet Attach:

    Enterprise FC 5220 (Ethernet Adapter)

    Ethernet port or adapter for Linux workstation

    Ethernet cables (as required)

Software Requirements

The following software is supported by the library driver:

- For Linux distribution support, refer to 110
- IBM Tape and Medium Changer Device Driver (IBMtape) for Linux

To obtain the most current service and documentation for this software, refer to Appendix A, "Accessing Documentation and Software Online," on page 305.

Installation and Configuration Instructions

The IBM Tape Library driver for Linux, (*ibmatl*), is supplied in an *rpm* package. The following sections describe installation, removal, configuration, and verification procedures for *ibmatl*. Refer to any Linux distribution supporting *rpm* for *rpm* command information. You must have root authority to proceed with the installation of the driver. In the subsequent pages, you will see file names with xxxx or x.x.x.x in them. The xxxx or x.x.x.x refer to the version of the driver, which will change as IBM releases new driver levels. Use the actual driver version numbers as you perform the procedures.

Install and configure the LAN support before proceeding with installation of the *ibmatl* package. For more information, refer to the Linux documentation appropriate to the LAN support you are using.

The following files are installed on your system:

```
/etc/lmcpd (library manager control point daemon)
/etc/ibmatl.conf (configuration file)
/usr/lib/libibm.o (application interface object module)
/usr/lib/libibm64.o (64-bit application interface object module for 64-bit
       IBM zSeries system only)
/usr/include/sys/mtlibio.h (application interface header file)
/usr/bin/mtlib (tape library driver utility program)
/usr/bin/mtlib64 (64-bit tape library driver utility program for 64-bit IBM
       zSeries system only)
/usr/bin/mtevent (tape library driver utility program)
/usr/bin/trcatl (tape library driver diagnostic program)
```

On the IBM Linux for IBM S/390 and zSeries systems, documentation files are installed at:

```
/usr/share/doc/packages/ibmatl/README (readme file)
/usr/share/doc/packages/ibmatl/license (license file)
```

On Red Hat Linux for Intel™ PC based systems, documentation files are installed at:

```
/usr/share/doc/ibmatl-x.x.x.x/README (readme file, where x.x.x.x is the version)
/usr/share/doc/ibmatl-x.x.x.x/license (license file, where .x.x.x.x is the version)
```

Note: If this is an update of the *ibmatl* package, the existing */etc/ibmatl.conf* file is preserved (it will not be overwritten).

Installation Procedure

If *ibmatl* is already installed on your system, refer to "Updating Procedure" on page 275. This section assumes that you are installing *ibmatl* on a system where it is not installed.

Run the following command to install ibmatl rpm package:

```
>rpm -ivv ibmatl.x.x.x.x.os.rpm
```

Connecting the TotalStorage Enterprise 3494 Tape Library

Use a LAN connection (either Ethernet or Token Ring) through TCP/IP. The connection must be in place before the software is installed to ensure proper initialization of the daemon. For each 3494 Tape Library connected through TCP/IP, ensure that your machine has access to the Library Manager on the 3494

Tape Library. Use the *ping* utility to verify that you have network connectivity to the Library Manager. Consult your network administrator if you need help with this task.

Defining the Library Device to the lmcpd

After the software is installed and tape libraries are connected, the */etc/ibmatl.conf* file must be edited to define the library devices to the *lmcpd*. The format of the file is:

```
Symbolic name    Connection type   Identifier   Alternate LAN Connection
```

Symbolic name is used to identify each tape library. The symbolic names must be unique across all libraries defined in the *ibmatl.conf* file.

Connection type is used to define the type of connection to the library. For TCP/IP connections, this type is the IP address of the 3494 Library Manager. If the Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first IP address.

Identifier parameter is used to specify a name by which the Library Manager identifies the host machine. This identifier has a maximum length of 8 characters. The symbolic network name of the host is usually the best name to use, although any name is acceptable. This parameter is used only to identify a particular host to the 3494 Library Manager.

Alternate LAN Connection is used to specify an alternate IP address for a 3494 High Availability LAN configuration. If the High Availability Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first alternate LAN connection IP address.

The following examples show how to define the library name:

```
3494a         9.115.32.21                 jupiter
```

This stanza defines the *3494a* library that is connected to the host. The address of the 3494 Library Manager is *9.115.32.21*. The tape library uses *jupiter* as the identifier for the host.

```
libmgrc9     9.115.46.15     telos     9.115.46.17
```

This stanza defines the *libmgrc9* High Availability library that is connected through TCP/IP to the workstation. The IP addresses of the dual Library Managers are *9.115.46.15* and *9.115.46.17*.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury
```

This stanza defines libmgrc7 with a dual LAN card connected through TCP/IP to the workstation. The first address of the Library Manager on the 3494 Tape Library is 9.115.32.21 and the second address is 9.115.32.22. The library uses the mercury identifier for the host.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury  9.115.26.13   9.115.26.14
```

This stanza defines the libmgrc7 High Availability library with dual LAN cards connected through TCP/IP to the workstation. The IP addresses of the first Library Manager are 9.115.32.21 and 9.115.32.22. The IP addresses of the second Library Manager are 9.115.26.13 and 9.115.26.14. The library uses the mercury identifier for the host.

Adding or Deleting Entries in the Library Device

You can add or delete entries in */etc/ibmatl.conf* at any time. However, changes do not take effect until the daemon is started again. Use the UNIX *kill* command (kill -kill <pid>) to kill the *lmcpd* process, and enter the */etc/lmcpd* command to start the daemon again.

Ensure that activity on all libraries is completed before restarting the *lmcpd*.

Updating Procedure

If your current *ibmatl* was installed from an *rpm* package previously, issue the following command:

```
rpm -Uvv  ibmatl.x.x.x.x.s390.rpm         --for IBM Linux on S/390 systems
rpm -Uvv  ibmatl.x.x.x.x.i386.rpm         --for Red Hat Intel PC based systems
```

Querying the Installed Package

The query function is supported for the *ibmatl rpm* package only.

The installed *rpm* package can be queried by running the following commands to display information associated with the package.

To display information about *ibmatl*:

>**rpm -qi ibmatl**

To display the package's file list, enter the command:

> **rpm -ql ibmatl**

To display the states of files in the package, for example, *normal*, *not installed*, or *replaced*:

>**rpm -qs ibmatl**

Verifying the Install/Update

If the IBMtape device driver is installed from the *rpm* package, issue the following command:

>**rpm -V ibmatl**

Starting ibmatl

Start the daemon using the following command:

/etc/lmcpd

Uninstall Procedure

Attention: Do not try to uninstall the *ibmatl* software simply by removing the files that make up the *ibmatl* fileset. It is best to use the *rpm -e* command or the *uninstall* script.

Note: Halt all library activity before starting the uninstall procedure.

For the *rpm* package, run the command *rpm -e*:

rpm -evv ibmatl

If more than one level of *ibmatl* exists on the system, explicitly specify the level to remove it.

rpm -evv ibmatl-5.0.7.0

Problem Determination

A set of tools is provided with the software to determine if the *ibmatl* is functioning correctly.

Error Logging

The *ibmatl* uses the *syslog* facility to log the errors. Errors are logged according to the */etc/syslog.conf* file. The *ibmatl* uses the daemon facility for logging errors. Only errors are logged with this facility. For more information about using *syslog*, refer to your system administration manuals.

Daemon Trace Facility

The following trace facility is available for the *lmcpd* daemon:

```
trcatl -[ald]
```

Arguments:

```
-a              (trace all libraries defined in /etc/ibmatl.conf)
-l <name>       (name of library)
-d              (delete all log files; requires root authority)
```

Examples:

```
trcatl -a >/tmp/trace-out
trcatl -l libmgrc7
```

Note: The *-a* argument takes precedence over the *-l* argument.

Solaris

The software consists of a daemon that communicates directly with the Library Manager of the 3494 Tape Library through RS-232 or LAN, a utility program that provides a command-line interface to the daemon, and a C object module that can be linked with user applications to provide a communication interface with the daemon.

This software is known as a *driver* throughout this document because it provides the software and interface necessary to *drive* the Tape Library. However, the product does not consist of a *device driver* in the true sense of the term. That is, it is not an extension of the operating system kernel, such as a SCSI tape device driver.

A typical environment for the IBM Sun Automated Tape Library Driver is a Sun workstation that acts as a data server on a network with SCSI tape devices providing backup or restore and data server functions.

The Library Manager control point daemon (*lmcpd*) is provided in the package. The *lmcpd* is a process that is always running on the system. It provides direct communication with the 3494 Library Manager. An application links with the supplied C object module using the interface described in the *IBM Tape Device Drivers: Programming Reference*. The subroutines in this module communicate with the *lmcpd* to perform the various library operations using standard UNIX namespace sockets.

The *lmcpd* communicates with the Library Manager either through a standard 25 pin null modem D-shell RS-232 cable or through TCP/IP. The */etc/ibmatl.conf* configuration file is used to define the type of attachment for each library. Refer to "Defining the Symbolic Name to the lmcpd" on page 280 for more information.

Data Flow

The software described in this chapter covers the library driver. The interface between the application and the library driver is described in the *IBM Tape Device Drivers: Programming Reference*. Figure 62 on page 278 illustrates the data flow.

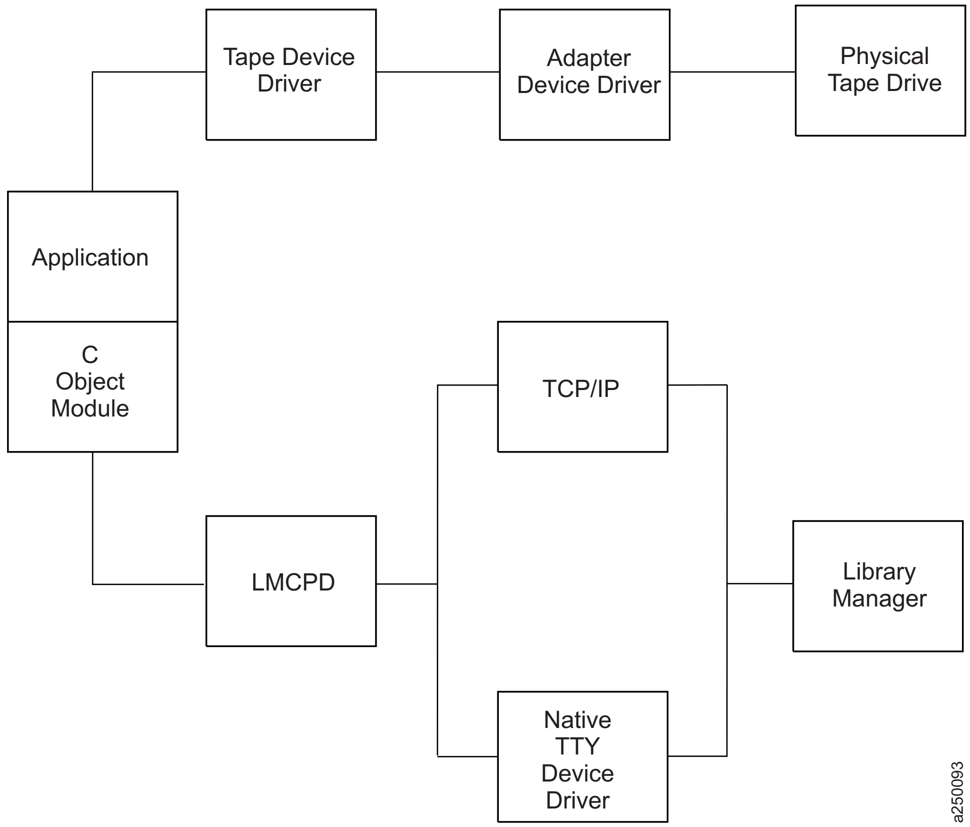


Figure 62. Data Flow for Solaris 3494 Enterprise Tape Library Driver

Product Requirements

The following software and hardware are supported by this product.

Hardware Requirements

The following hardware is supported by the library driver:

- 3494 Enterprise Tape Library with SCSI-attached 3490E (Model C1A or C2A), 3590 and 3592 drives
- IBM System Storage TS1120 Tape Drive
- IBM Virtualization Engine TS7510
- IBM TotalStorage Virtual Tape Server (Models B10, B18, and B20)
- One of the following options depending on which connection (RS-232 or LAN) is required for the Enterprise Library Manager:
  – RS-232:

    One standard 25 pin null modem D-shell RS-232 cable (maximum of 15.24 meters [50 feet])
  – LAN:
    - Token-Ring or Ethernet adapter card in a Sun workstation attached to the 3494 Enterprise Library Manager
    - LAN cable

Software Requirements

The following software is required for the library driver:

- Sun Microsystems Solaris Version 2.6, 7, 8, 9, or 10
- IBM Tape and Medium Changer Device Driver for Solaris

To obtain the most current service and documentation for this software, refer to Appendix A, "Accessing Documentation and Software Online," on page 305.

Software Compatibility

The following *optional* software is supported by the library driver:

IBM Tivoli Distributed Storage Manager for Solaris

Installation Instructions

The IBM Tape Library driver for Solaris is installed using the standard Sun *package* facility. The IBM Tape Library Driver for Solaris is the *lmcpd* package.

Preinstallation Considerations

The LAN or TTY support must exist on the machine before installing the IBM Tape Library daemon for SunOS. Install and configure the LAN or TTY adapter and the associated LAN or TTY adapter device driver before proceeding with the installation of the *lmcpd* package. For more information, refer to the documentation appropriate for the LAN or TTY adapter and the LAN or TTY adapter device driver that you are using.

You must have *root* authority to perform this installation procedure.

Note: If an earlier version of the product is installed on the machine, execute the uninstall procedure before you enter the following commands. Refer to "Uninstall Procedure" on page 282 for more information.

The following files are installed on the system:

/etc/lmcpd (Library Manager control point daemon)
/etc/ibmatl.conf (configuration file)
/usr/lib/libibm.o (application interface object module)
/usr/lib/libibm64.o (64 bit application interface object module)
/usr/include/sys/mtlibio.h (application interface header file)
/usr/bin/mtlib (tape library driver utility program)
/usr/bin/mtevent (tape library driver utility program)
/usr/bin/trcatl (tape library driver diagnostic program)
/opt/lmcpd (subdirectory)
/opt/lmcpd/ibmatl.conf (backup configuration file)

Note: If this is an update of the lmcpd package, the existing */etc/ibmatl.conf* file is preserved. It will not be overwritten. A copy of *ibmatl.conf* is also installed in the */opt/lmcpd* directory.

Installation Procedure

Use the following commands to install the *lmcpd* package. Substitute the device special file name for the installation source device that is appropriate for your system.

If a previous version is installed on the system, uninstall the previous version first, enter the following command:

```
/usr/sbin/pkgrm  lmcpd
```

To install the package from the directory that the driver is located, enter the following command:

```
/usr/sbin/pkgadd -d lmcpd.x.x.x.x
```

To verify that the installation was successful, enter the following command:

```
/usr/bin/pkginfo lmcpd
```

Connecting the 3494 Enterprise Tape Library

Use one of the following methods to connect the workstation to the 3494 Tape Library:

- Use a standard 25 pin null modem D-shell RS-232 cable. Place the cable on the native serial port or on the eight-port or 16 port asynchronous adapter.
- Use a LAN connection (either Ethernet or Token-Ring) through TCP/IP.

In either case, the connection must be in place before the software is installed to ensure proper initialization of the daemon.

For each 3494 Tape Library connected with an RS-232 cable, follow the directions in "Configuring the Serial Port" on page 280. For each 3494 Tape Library connected through TCP/IP, ensure that your machine has access to the Library Manager on the Enterprise device. Use the *ping* utility to verify that you have network connectivity to the Library Manager. Consult your network administrator if you need help with this task.

Identify the hostname of your workstation to the Enterprise Library Manager by following the instructions in the *IBM TotalStorage Enterprise Automated Tape Library Operator Guide*, in the **Add LAN Host** section.

Configuring the Serial Port

After the 3494 Tape Library is connected to the workstation, you must configure the serial port to which it is connected.

Set the following TTY port attributes:

Baud rate:	9600
Data bits:	8
Stop bits:	1
Parity:	None

Defining the Symbolic Name to the lmcpd

After the software is installed and all of the desired tape libraries are connected to the system, the */etc/ibmatl.conf* file must be edited to define the tape libraries to the *lmcpd*. The format of the file is:

```
Symbolic name     Connection type     Identifier     Alternate LAN Connection
```

Symbolic name is used to identify each tape library. The symbolic names must be unique across all libraries defined in the*/etc/ibmatl.conf* file.

Connection type is used to define the type of connection to the library (either RS-232 or TCP/IP). For RS-232 connections, this type is the device special file name of the *tty* device (for example, */dev/ttya* or */dev/tty1*). For TCP/IP connections, this type is the IP address of the Library Manager. If the Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first IP address.

Identifier is used to specify a name by which the Library Manager identifies the host machine. This identifier has a maximum length of eight characters. The host name of the workstation is usually the best name to use (although it is not necessary). This parameter is used only to identify a particular host to the 3494 Enterprise Library Manager.

Alternate LAN Connection is used to specify an alternate IP address for an Enterprise High Availability LAN-attached configuration. If the High Availability Library has a dual lan card installed, the second IP address of the dual LAN card should be entered after the first alternate LAN connection IP address.

Note: For RS-232 attachments, the alternate LAN connection is not applicable. Failover for RS-232 connections proceeds over the single serial line.

The following examples show how to define the library name:

```
3494a             /dev/ttya                mercury
```

This stanza defines the *3494a* library that is connected by */dev/ttya* to the workstation. The library uses the *mercury* identifier for the host.

```
3494b             9.115.32.21          jupiter
```

This stanza defines the *3494b* library that is connected through TCP/IP to the workstation. The address of the Library Manager on the Enterprise Tape Library is *9.115.32.21*. The library uses the *jupiter* identifier for the host.

```
  libmgrc9         9.115.46.15           telos         9.115.46.17
```

This stanza defines the *libmgrc9* High Availability library that is connected through TCP/IP to the workstation. The IP addresses of the dual Library Managers are *9.115.46.15* and *9.115.46.17*.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury
```

This stanza defines libmgrc7 with a dual LAN card connected through TCP/IP to the workstation. The first address of the Library Manager on the 3494 Tape Library is 9.115.32.21 and the second address is 9.115.32.22. The library uses the mercury identifier for the host. The following stanza defines the libmgrc7 High Availability library with dual LAN cards connected through TCP/IP to the workstation. The IP addresses of the first Library Manager are 9.115.32.21 and 9.115.32.22. The IP addresses of the second Library Manager are 9.115.26.13 and 9.115.26.14. The library uses the mercury identifier for the host.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury  9.115.26.13   9.115.26.14
```

Adding or Deleting Entries in the Library

You can add or delete entries in */etc/ibmatl.conf* at any time. However, changes do not take effect until the daemon is started again. Use the UNIX *kill* command (*kill -kill <pid>*) to kill the *lmcpd* process, then enter the */etc/lmcpd* command to start the daemon again.

Ensure that the library activity on all of the libraries is completed before restarting the *lmcpd*.

Uninstall Procedure

Halt all library activity before starting the uninstall procedure. Kill the *lmcpd* daemon if it is currently executing. Enter the following command to uninstall the library support from your workstation:

```
/usr/sbin/pkgrm lmcpd
```

Problem Determination

A set of tools is provided with the software to determine if the *lmcpd* is functioning correctly.

Error Logging

The *lmcpd* uses the *syslog* facility to log the errors. Errors are logged according to the */etc/syslog.conf* file. The *lmcpd* uses the daemon facility for logging errors. Only errors are logged with this facility. For more information about using *syslog*, refer to your system administration manuals.

Daemon Trace Facility

The following trace facility is available for the *lmcpd* daemon:

```
trcatl -[ald]
```

Arguments:

```
-a              (trace all libraries defined in /etc/ibmatl.conf)
-l <name>       (name of library)
-d              (delete all log files; requires root authority)
```

Examples:

```
trcatl -a >/tmp/trace-out
trcatl -l libmgrc7
```

Note: The *-a* argument takes precedence over the *-l* argument.

Windows

The software consists of a tape library service that communicates directly with the Library Manager of the 3494 Enterprise Tape Library through LAN, a utility program that provides a command-line interface to the service, and a static library and DLL that can be used to provide a communication interface with the service.

A typical environment for the Windows Automated Tape Library Service is a workstation running Microsoft Windows that acts as a data server on a network with SCSI tape devices providing backup or restore and data server functions.

The IBM Automated Tape Library service is provided in the installation package. This service is installed with STARTUP=AUTOMATIC, which means that it is started when the Microsoft Windows system is started. An application program may access those functions through API calls. The interface is described in the *IBM Tape Device Drivers: Programming Reference*. The subroutines in this module communicate with the Enterprise Library Manager to perform the various library operations.

The service communicates with the Library Manager through TCP/IP. The *c:\winnt\ibmatl.conf* configuration file is used to define the type of attachment for each library. Refer to “Defining the Symbolic Name to the Service” on page 287 for more information.

Data Flow

The software described in this chapter covers the library driver. The interface between the application and the library driver is described in the *IBM Tape Device Drivers: Programming Reference*. Figure 63 on page 284 illustrates the data flow.

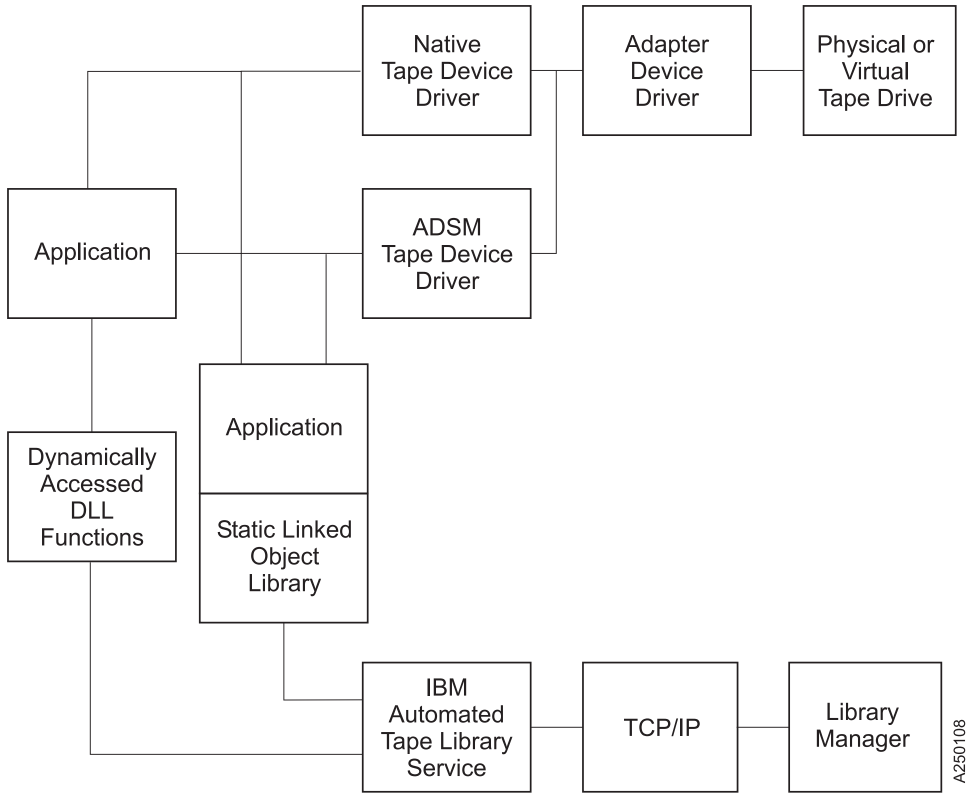


Figure 63. Data Flow for Windows 3494 Enterprise Tape Library Driver

Product Requirements

The following hardware and software components are required and supported by IBM.

Hardware Requirements

The following hardware is required for the library driver:

- 3494 Enterprise Tape Library with Enterprise Tape System 3590 and 3592 drives
- IBM System Storage TS1120 Tape Drive
- IBM Virtualization Engine TS7510
- IBM TotalStorage Virtual Tape Server (Models B10, B18, and B20)

The following options are required for operation of the Enterprise Library Manager:

- Token-Ring or Ethernet adapter card in an Intel- compatible workstation (486DX or higher) running Microsoft Windows NT or Microsoft Windows 2000 and attached to the 3494 Enterprise Tape Library
- LAN cable

Software Requirements

The following software is required for the library driver:

- A workstation running Microsoft Windows NT 4.0 with Fix Pack 3 or later
- A workstation running Microsoft Windows 2000 Build 2195 or later
- A workstation running Microsoft Windows Server 2003

- A workstation running Microsoft Windows Server 2008
- IBM Windows Tape Device Driver or the ADSM/TSM Device Driver for Microsoft Windows NT if using ADSM/TSM

To obtain the most current service and documentation for this software, refer to Appendix A, "Accessing Documentation and Software Online," on page 305.

Installation Instructions

The IBM Tape Library service for Microsoft Windows is installed by executing the install binary, which has the format *ibmatl.x.x.x.x.x86.exe* for 32-bit Windows 200*x* running on *x*86, *ibmatl.x.x.x.x.i64.exe* for 64-bit Windows 2003 running on IA64, and *ibmatl.x.x.x.x.x64.exe* for 64-bit Windows Server 2003 and Windows Server 2008 running on AMD64 and EM64T.

Installation Procedure

Use the following procedure to install the IBM Automated Tape Library Service or to upgrade the software level of your service on your workstation.

Follow these steps to perform the installation of the service and its associated software:

1. Verify that the prerequisites are satisfied. Refer to "Product Requirements" on page 284.
2. Ensure that the workstation is on the LAN.
3. Log on as Administrator.
4. If the IBM Automated Tape Library Service is currently installed, ensure that all programs and services that use the service are stopped, and then remove it from the system.
5. Ensure that the IBM Automated Tape Library Service is not running.
   - On Windows NT, click the **Start** button, move to **Settings**, then click **Control Panel**. Double-click the **Services** icon.
   - On Windows 2000 and 2003, click the **Start** button, move to **Settings**, then click **Control Panel**. Double-click the **Administrative Tools** icon, then double-click the **Services** icon.
   - On Windows 2008, click the **Start** button, double-click the **Administrative Tools** icon, then double-click the **Services** icon.

   The **Services** window opens. Scroll through the entries until you find **IBM Automated Tape Library**. If you do not find the entry for **IBM Automated Tape Library**, the service is not installed; click **Close**. If there is an entry and the status is blank, the service is not running; click **Close**. If the status is **Started**, click **IBM Automated Tape Library**, then click **Stop**.
   - On Windows NT, wait for the service to stop and the **Status** to be set to blank, then click **Close**.
   - On 32–bit Windows 200x, wait for the service to stop and the **Service status** to be set to **Stopped**, then close both windows.
   - On 64–bit Windows, the service is stopped and removed automatically during the uninstallation.
6. Obtain *ibmatl* driver from the IBM storage FTP server in binary.
7. Locate and execute *ibmatl* driver.
8. You see some windows displayed indicating the image is being unpacked and read and the InstallShield Wizard is being set up. After these are displayed,

the screen stops on a window entitled **Welcome to the IBM Automated Tape Library Setup Program**. Follow the instructions on the window, then select **Next**.

9. The **Software License Agreement** window opens. Read the contents, then click **Yes** if you accept the terms. If you do not accept the terms, the installation script exits.
10. The **Choose Destination Location** window opens. You can accept the default destination folder by selecting **Next >**, or you can select **Browse...** and select another destination folder by following the prompts in the subsequent dialog. If you do not select the default destination folder, you must record your destination folder for future reference. After you select the folder, click **Next >**.
11. The **Setup Complete** window opens. You can check the **Yes, I want to view the Read Me file** prompt to read the file, then click **Finish**, or just click **Finish**. The **Read Me** file is stored in the destination file that you indicated in step 10.
12. After you click **Finish**, if you did not select to view the **Read Me** file, the installation dialog exits. If you chose to view the **Read Me**, it is displayed. After you read it, exit the Notepad application. An informational message is displayed, indicating that the setup is complete. Click **OK** to exit the installation dialog.
13. Verify that you can use the *ping* utility successfully for any Tape Library you want to access from your workstation.
14. Update the *c:\winnt\ibmatl.conf* file, if necessary, with entries for each Tape Library you want to access from your workstation. Refer to "Defining the Symbolic Name to the Service" on page 287 for details.
15. On Windows NT, start the service by restarting the system or using the **Services** option in the **Control Panel** window.

    On Windows 2000 and 2003, start the service using the **Services** icon, which is on the **Administrative Tools** icon in the **Control Panel** window.

    On Windows 2008, start the service using the **Services** icon, which is on the **Administrative Tools** icon.

    Note: When a firewall is enabled on Windows, you must allow the lmcpd program to access the network. Do not block the network IP port of 3494, which is used for communication between the 3494 library and lmcpd.

    For example, to allow lmcpd.exe access to the network on Windows 2008 servers, click on the **Windows Firewall** icon in the **Control Panel** window. Next, click on the button to allow a program or an exception, and then click the **Add Program** button. Select your browser and then select lmcpd.exe in `/Windows/system32` for an x86 system or `/Windows/SysWOW64` for i64 or x64 systems.

Connecting the 3494 Enterprise Tape Library

Use a LAN connection (either Ethernet or Token-Ring) through TCP/IP to connect the workstation to the 3494 Enterprise Tape Library. The connection must be in place before the service is started to ensure proper initialization of the service.

Ensure that your machine has access to the Library Manager on the Tape Library. Use the *ping* utility to verify that you have network connectivity to the Library Manager. Consult your network administrator if you need help with this task.

Identify the hostname of your workstation to the Enterprise Library Manager by following the instructions in the *IBM TotalStorage Enterprise Automated Tape Library Operator Guide*, in the **Add LAN Host** section.

Defining the Symbolic Name to the Service

After the software is installed and all of the desired tape libraries are connected to the system, the *c:\winnt\ibmatl.conf* file must be edited to define the tape libraries to the service. The format of the file is:

```
Symbolic name     Connection type     Identifier     Alternate LAN Connection
```

Symbolic name identifies each tape library. The symbolic names must be unique across all libraries defined in the *c:\winnt\ibmatl.conf* file.

Connection type is the IP address of the Library Manager. If the Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first IP address.

Identifier specifies a name by which the Library Manager identifies the host machine. This identifier has a maximum length of eight characters. The host name of the workstation is usually the best name to use (although it is not necessary). This parameter only identifies a particular host to the Enterprise Library Manager.

The following stanza defines the *3494b* library that is connected through TCP/IP to the workstation. The address of the Library Manager on the 3494 Tape Library is *9.115.32.21*. The library uses the *jupiter* identifier for the host.

```
3494b            9.115.32.21             jupiter
```

The *Alternate LAN Connection* parameter specifies an alternate IP address for an Enterprise High Availability LAN configuration. If the High Availability Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first alternate LAN connection IP address.

The following stanza defines the *libmgrc9* High Availability library that is connected through TCP/IP to the workstation. The IP addresses of the dual Library Managers are *9.115.46.15* and *9.115.46.17*.

```
libmgrc9        9.115.46.15           telos        9.115.46.17
```

The following stanza defines libmgrc7 with a dual LAN card connected through TCP/IP to the workstation. The first address of the Library Manager on the 3494 Tape Library is 9.115.32.21 and the second address is 9.115.32.22. The library uses the mercury identifier for the host.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury
```

The following stanza defines the libmgrc7 High Availability library with dual LAN cards connected through TCP/IP to the workstation. The IP addresses of the first Library Manager are 9.115.32.21 and 9.115.32.22. The IP addresses of the second Library Manager are 9.115.26.13 and 9.115.26.14. The library uses the mercury identifier for the host.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury  9.115.26.13   9.115.26.14
```

Adding or Deleting Libraries to the Service

You can add or delete entries in *c:\winnt\ibmatl.conf* at any time. However, changes do not take effect until the tape service starts again.

On Windows NT, stop and start the service using the **Services** option in the **Control Panel** window.

On Windows 200x, stop and start the service using the **Services** icon, which is on the **Administrative Tools** icon in the **Control Panel** window.

Ensure that the library activity on all of the libraries is completed before starting the service again.

Uninstallation Procedure

Use the following procedure to remove the IBM Automated Tape Library Service permanently from your workstation.

Note: If you are upgrading the software level of your service, follow the steps in "Installation Procedure" on page 285.

Follow these steps to uninstall the service and its associated software:

1. Log on as Administrator.
2. Back up the *c:\winnt\ibmatl.conf* file if you want to keep a copy of this file because the uninstall procedure erases the file.
3. Ensure that all programs and services that use the IBM Automated Tape Library Service are stopped.
4. Ensure that the IBM Automated Tape Library Service is not running. Click the **Start** button, move to **Settings**, then click **Control Panel**.
   - On Windows NT, double-click the **Services** icon.
   - On Windows 200x, double-click the **Administrative Tools** icon, then double-click the **Services** icon.

   The **Services** window opens. Scroll through the entries until you find **IBM Automated Tape Library**. If the status is blank, the service is not running; click **Close**. If the status is **Started**, click **IBM Automated Tape Library**, then click **Stop**.
   - On Windows NT, wait for the service to stop and the **Status** to be set to blank, then click **Close**.
   - On 32–bit Windows 200x, wait for the service to stop and the **Service status** to be set to **Stopped**, then close both windows.
   - On 64–bit Windows, the service is stopped and removed automatically during the uninstallation.
5. Open a **Command Prompt** window, enter **lmcpd -remove**, press **Enter**, then wait for the command to complete.

   Note: If you forget this step, the next time that the system is started, Microsoft Windows NT tries (unsuccessfully) to start the service again. To resolve this situation, you must reinstall the service, then uninstall it.
6. Double-click **Add/Remove Programs** in Windows NT/200x or **Programs and Features** on Windows 2008 in the **Control Panel** window.
7. Scroll to and select **IBM Automated Tape Library**.
8. For Windows NT, click **Add/Remove...**.

   For Windows 200x, click **Change/Remove**.
9. A **Confirm File Deletion** window opens. Click **Yes**.
10. After various dialogs run, a **Remove Programs from Your Computer** window opens, which indicates the components that were removed successfully. Click **OK** to exit the dialog.

11. For Windows NT, click **Cancel** on the **Add/Remove Programs Properties** window.

    For Windows 2000 and 2003, close the **Add/Remove Programs Properties** window.

    For Windows 2008, close the **Programs and Features** window.

TRU64

The software consists of a daemon that communicates directly with the Library Manager of the 3494 Enterprise Tape Library through Ethernet or Token Ring LAN, a utility program that provides a command-line interface to the daemon, and a C object module that can be linked with user applications to provide a communication interface with the daemon.

Note: In the Tru64 operating system, this is a program that runs unattended, in the background, to perform a standard service. Some daemons are triggered automatically to perform their task; others operate periodically.

This software is known as a *driver* throughout this document because it provides the software and interface necessary to *drive* the 3494 Tape Library. However, the product does not include a *device driver* in the true sense of the term. That is, it is not an extension of the operating system kernel, such as a tape device driver.

A typical environment for the Tru64 Automated Tape Library Driver is a Tru64 workstation that acts as a data server on a network with tape devices providing backup or restore and data server functions.

The Library Manager control point daemon (*lmcpd*) is provided in the package. The *lmcpd* is a process that is always running on the system. It provides direct communication with the Library Manager. An application links with the supplied C object module using the interface described in the *IBM Tape Device Drivers: Programming Reference*. The subroutines in this module communicate with the *lmcpd* to perform the various library operations using standard UNIX namespace sockets.

The *lmcpd* communicates with the Library Manager through TCP/IP. The */etc/ibmatl.conf* configuration file is used to define the attachment for each library. Refer to "Defining the Symbolic Name to the lmcpd" on page 280 for more information.

Data Flow

The software described in this chapter covers the library driver. The interface between the application and the library driver is described in the *IBM Tape Device Drivers: Programming Reference*. Figure 61 on page 272 illustrates the data flow.

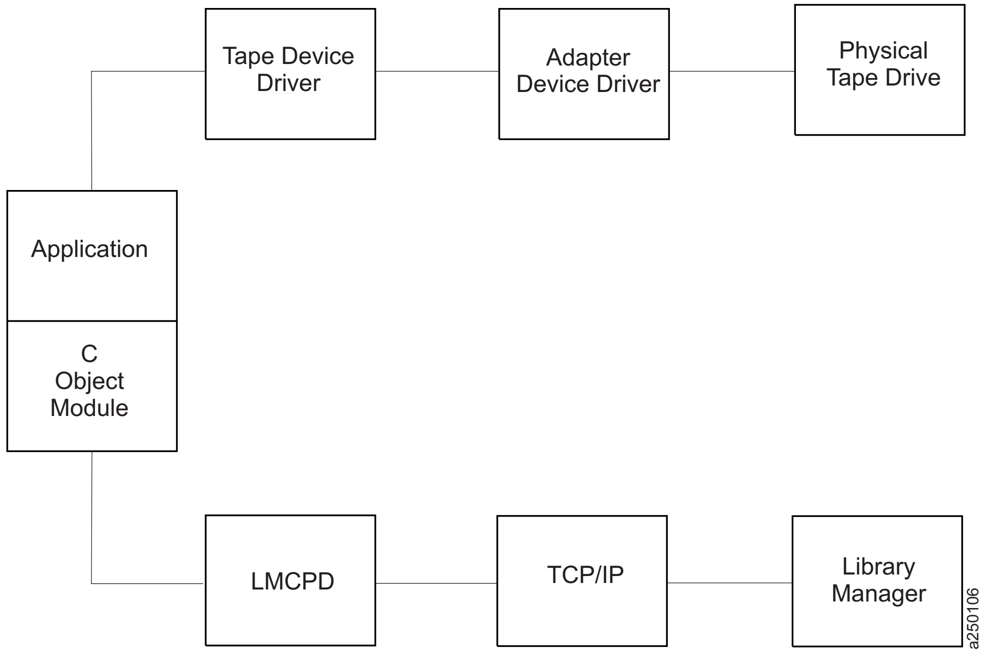


Figure 64. Data Flow for TRU 64 3494 Enterprise Tape Library Driver

Product Requirements

The following hardware and software components are required and supported by IBM.

Hardware Requirements

The following hardware is supported by and required for the library driver:

- IBM TotalStorage 3494 Enterprise Tape Library with IBM 3590 Model E1A or H1A drives with Fibre Channel interface.
- One of the following options, depending on which LAN connection is used for the Enterprise Library Manager:
  - Token-Ring Attach:

    Enterprise FC 5219 (Token-Ring Adapter)

    Token-Ring adapter card for Tru64 workstation

    Token-Ring cables (as required)
  - Ethernet Attach:

    Enterprise FC 5220 (Ethernet Adapter)

    Ethernet port or adapter for Tru64 workstation

    Ethernet cables (as required)

Software Requirements

The following software is required for the library driver:

- Tru64 5.1A operating system.

To obtain the most current service and documentation for this software, refer to Appendix A, “Accessing Documentation and Software Online,” on page 305.

Installation and Configuration Instructions

The IBM Tape Library driver for Tru64, (*ibmatl*), is a 64-bit software supplied in a *product kit package*. The following sections describe installation, removal, configuration, and verification procedures for *ibmatl*. You must have root authority to proceed with the installation of the driver.

In the subsequent pages, you will see file names with xxxx or x.x.x.x in them. The xxxx or x.x.x.x refer to the version of the driver, which will change as IBM releases new driver levels. Use the actual driver version numbers as you perform the procedures.

When you are installing from a *tar* package of the product kit from our web site, you need to unpack the *tar* files and keep them in directories that were created, along with the files the installation process creates, in those directories. You will also need to remember the name of the top level directory in order to uninstall and/or update the driver and utilities.

Install and configure the LAN support before proceeding with the installation of the *ibmatl* package. For more information, refer to the Tru64 documentation appropriate to the LAN support you are using.

The following files are installed on your system:

```
/etc/lmcpd (library manager control point daemon)
/etc/ibmatl.conf (configuration file)
/usr/lib/libibm.o (application interface object module)
/usr/include/sys/mtlibio.h (application interface header file)
/usr/bin/mtlib (tape library driver utility program)
/usr/bin/mtevent (tape library driver utility program)
/usr/bin/trcatl (tape library driver diagnostic program)
/usr/doc/ibmatl/README (readme file)
/usr/doc/ibmatl/license (license file)
```

Note: If this is an update of the *ibmatl* package, the existing */etc/ibmatl.conf* file is preserved (it will not be overwritten).

Installation Procedure

Obtain *ibmatl* driver from the IBM storage FTP server in binary and execute

```
setld
```

command to install *ibmatl* driver:

```
/usr/sbin/setld -l /driver_location/ibmatl.x.x.x.x.kit
```

Connecting the TotalStorage Enterprise 3494 Tape Library

Use a LAN connection (either Ethernet or Token Ring) through TCP/IP. The connection must be in place before the software is installed to ensure proper initialization of the daemon. For each 3494 Tape Library connected through TCP/IP, ensure that your machine has access to the Library Manager on the 3494 Tape Library. Use the *ping* utility to verify that you have network connectivity to the Library Manager. Consult your network administrator if you need help with this task.

Defining the Library Device to the lmcpd

After the software is installed and tape libraries are connected, the */etc/ibmatl.conf* file must be edited to define the library devices to the *lmcpd*. The format of the file is:

```
Symbolic name     Connection type     Identifier     Alternate LAN Connection
```

Symbolic name is used to identify each tape library. The symbolic names must be unique across all libraries defined in the *ibmatl.conf* file.

Connection type is used to define the type of connection to the library. For TCP/IP connections, this type is the IP address of the 3494 Library Manager. If the Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first IP address.

Identifier parameter is used to specify a name by which the Library Manager identifies the host machine. This identifier has a maximum length of 8 characters. The symbolic network name of the host is usually the best name to use, although any name is acceptable. This parameter is used only to identify a particular host to the 3494 Library Manager.

Alternate LAN Connection is used to specify an alternate IP address for a 3494 High Availability LAN configuration. If the High Availability Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first alternate LAN connection IP address.

The following examples show how to define the library name:

```
3494a     9.115.32.21     jupiter
```

This stanza defines the *3494a* library that is connected to the host. The address of the 3494 Library Manager is *9.115.32.21*. The tape library uses *jupiter* as the identifier for the host.

```
libmgrc9     9.115.46.15     telos     9.115.46.17
```

This stanza defines the *libmgrc9* High Availability library that is connected through TCP/IP to the workstation. The IP addresses of the dual Library Managers are *9.115.46.15* and *9.115.46.17*.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury
```

This stanza defines libmgrc7 with a dual LAN card connected through TCP/IP to the workstation. The first address of the Library Manager on the 3494 Tape Library is 9.115.32.21 and the second address is 9.115.32.22. The library uses the mercury identifier for the host. The following stanza defines the libmgrc7 High Availability library with dual LAN cards connected through TCP/IP to the workstation. The IP addresses of the first Library Manager are 9.115.32.21 and 9.115.32.22. The IP addresses of the second Library Manager are 9.115.26.13 and 9.115.26.14. The library uses the mercury identifier for the host.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury  9.115.26.13   9.115.26.14
```

Adding or Deleting Entries in the Library Device

You can add or delete entries in */etc/ibmatl.conf* at any time. However, changes do not take effect until the daemon is started again. Use the UNIX *kill* command (kill -kill <pid>) to kill the *lmcpd* process, and enter the */etc/lmcpd* command to start the daemon again.

Ensure that activity on all libraries is completed before restarting the *lmcpd*.

Updating Procedure

To update the *ibmatl* kit, remove the old version existing in the system first and them update it. Issue the command to check which version of *ibmatl* is installed:

```
setld -l
```

Uninstall the old version (yyyy):

```
setld -d IBMATL_Vyyyy
```

Install the new version:

```
setld -l /driver_location/ibmatl.x.x.x.x.kit
```

After the update, the *lmcpd* daemon is started automatically.

Querying the Installed Package

The query function is supported for the *ibmatl* product kit. Run the following commands to display whether the *ibmatl* kit is installed on the system. The system state is listed on standard output in three columns: Subset, Status, and Description.

```
setld -i
```

To display the *ibmatl* product kit fileset and located directory, enter:

setld -i IBMATL_Vxxxx

Verifying the Install/Update

To verify the existence of the installed *ibmatl* subset, the *setld -v* executes any V phase processing included in the subset control program, except during installation:

```
setld -v IBMATL_Vxxxx
```

Use the *fverify* command to verify the files of a specific subset. The *fverify* command reports missing files and inconsistencies in file size, checksum, user ID, group ID, permissions, and file type.

```
cd /
/usr/lbin/fverify -n < /usr / .smdb./IBMATL_Vxxxx.inv
```

Note: Once the *ibmatl.conf* is modified, it is normal to encounter verification errors on the file checksum and size.

Starting ibmatl

Start the daemon using the following command:

/etc/lmcpd

Uninstall Procedure

Attention: Do not try to uninstall the *ibmatl* product kit simply by removing the files that make up the *ibmatl* fileset. It is best to use the *setld -d* command.

Note: Halt all library activity before starting the uninstall procedure.

Run this command to uninstall the product kit:

```
setld -d IBMATL_Vxxxx
```

The existing *ibmatl.conf* file is always retained for future installation use after uninstall procedures. If this file is no longer needed, you have to remove it manually.

Problem Determination

A set of tools is provided with the software to determine if the *ibmatl* is functioning correctly.

Error Logging

The *ibmatl* uses the *syslog* facility to log the errors. Errors are logged according to the */etc/syslog.conf* file. The *ibmatl* uses the daemon facility for logging errors. Only errors are logged with this facility. For more information about using *syslog*, refer to your system administration manuals.

Daemon Trace Facility

The following trace facility is available for the *lmcpd* daemon:

```
trcatl -[ald]
```

Arguments:

```
-a              (trace all libraries defined in /etc/ibmatl.conf)
-l <name>       (name of library)
-d              (delete all log files; requires root authority)
```

Examples:

```
trcatl -a >/tmp/trace-out
trcatl -l libmgrc7
```

Note: The *-a* argument takes precedence over the *-l* argument.

SGI IRIX

The software consists of an application daemon that communicates with the 3494 Automated Tape Library's Library Manager (LM) through Ethernet or Token Ring LAN, or an RS-232 serial connection. A utility program (*mtlib*) provides a command-line interface to the daemon. A C object module (*libibm.o*) is also provided that links with user written applications to provide control communication with the tape library.

Note: In a UNIX system, such as SGI IRIX, a daemon is a program that runs unattended in the background to perform a standard service. Some daemons are triggered automatically to perform their task while others operate periodically. The 3494 Enterprise Automated Tape Library Daemon for SGI IRIX is designed to be invoked at system IPL time and run continuously to service applications communicating with the Tape Library.

A typical environment for the 3494 Automated Tape Library daemon for SGI IRIX is an SGI server on a Local Area Network with SCSI attached tape devices providing backup and restore functions for network attached clients.

The daemon, specifically known to the SGI IRIX system as the IBM Library Manager Control Point Daemon (*lmcpd*), communicates with 3494 Enterprise Automated Tape Libraries that are defined in a configuration file. This file, *ibmatl.conf*, is placed in the */etc* directory when the software package is installed. Customization of this file is described in "Defining the Library Device to the lmcpd" on page 300.

Application developers can use the supplied 'C' object module (*libibm.o*) to interface software applications with *lmcpd*. The subroutines in this module communicate with the daemon to perform mount, demount, and other library control operations. Refer to "SGI IRIX 3494 Enterprise Tape Library Driver in the *IBM Tape Device Drivers: Programming Reference* for details.

Data Flow

Figure 65 on page 297 illustrates the software and hardware components that comprise an application's communication paths with the 3494 Enterprise Tape Library. The path from the top left to the top right is the data path from the application to the actual tape devices. These software drivers are provided by the SGI IRIX operating system.

The path from the application box proceeding to the right at the bottom of the diagram shows an application linked with the supplied 'C' object module communicating with lmcpd and the 3494 Enterprise Library Manager. The *mtlib* program (shipped with the software package) is an example of an application program that interacts with *lmcpd*.

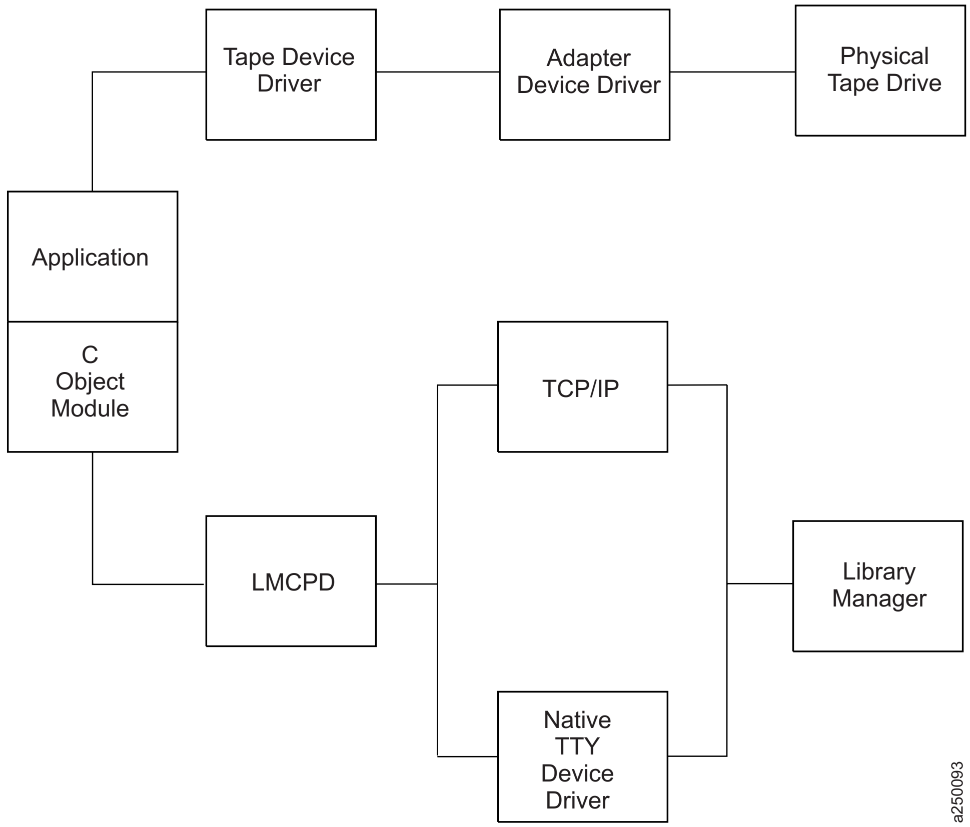


Figure 65. Data Flow for SGI IRIX 3494 Enterprise Tape Library Driver

Product Requirements

The following software and hardware are required to use this product.

Hardware Requirements

The following hardware is required for the library driver:

- 3494 Enterprise Tape Library with 3590 Model B1A, E1A, or H1A tape drives
- One of the following options depending on which connection (RS-232 or LAN) is required for the Enterprise Tape Library:
  - RS-232: One standard 25 pin null modem D-shell RS-232 cable (maximum of 15.24 m [50 feet])

    Note: One RS-422 to RS-232 converter is needed for direct attachment to an Enterprise Model HA1 (High Availability).
  - LAN:
    - Token-Ring or Ethernet adapter support in an SGI workstation attached to the 3494 Enterprise Tape Library
    - LAN cable
    - Enterprise FC 5219 (Token-Ring Adapter)
    - Enterprise FC 5220 (Ethernet Adapter)

Software Requirements

The following software is required:

- IRIX Version 6.4 or later

- 3494 Enterprise Tape Library Daemon for SGI IRIX

Note: The device driver for the 3590 Tape Drive is provided by SGI.

Installation Instructions

The 3494 Enterprise Tape Library Daemon for SGI IRIX can be installed by either using the *swmgr* program menus or the *inst* command.

Installation Procedure

The LAN or TTY support must exist on the machine before installing the Enterprise software for IRIX. Install and configure the LAN or TTY support before proceeding with the installation of the *lmcpd* package. For more information, refer to the documentation appropriate for the LAN or TTY configuration in SGI IRIX.

You must have *root* authority to perform this installation procedure.

Note: If you are updating software to a newer version, you will need to uninstall the existing IBM Automated Tape Library software before installing new software. Refer to "Uninstallation Procedure" on page 298 for more information.

The following files are installed on the SGI IRIX system:

/etc/lmcpd (library manager control point daemon)

/etc/ibmatl.conf (configuration file)

/usr/lib/libibm.o (application interface object module)

/usr/lib/libibm64.o (64 bit application interface object module)

/usr/include/sys/mtlibio.h (application interface header file)

/usr/bin/mtlib (tape library driver utility program)

/usr/bin/mtevent (tape library driver utility program)

/usr/bin/trcatl (tape library driver diagnostic program)

Note: If this is an update of the *lmcpd* package, the existing */etc/ibmatl.conf* file is preserved. (It will not be overwritten.)

To install the 3494 Enterprise Tape Library driver, use the *inst* command. The package is named

```
ibmatl.n.n.n.n.tardist
```

For example, to install Version 4.1.5.0 of the library driver, press Enter after you type: inst -a -u all -f ibmatl.4.1.5.0.tardist

The Tape Library Daemon for SGI IRIX can also be installed using the *swmgr* program.

After installing the new software, you must restart the Enterprise Tape Library Daemon by either restarting your computer or restarting manually from a command prompt by entering:

```
/etc/lmcpd
```

Uninstallation Procedure

Halt all library activity before starting the uninstall procedure. *Kill* the *lmcpd* daemon if it is executing currently.

The 3494 Enterprise Tape Library Daemon for SGI IRIX can be uninstalled by either using the *swmgr* program or from the command prompt by entering:

```
inst -a -R ibmatl
```

Connecting the 3494 Enterprise Tape Library

Use one of the following methods to connect the workstation to the 3494 Enterprise Tape Library:

- Use a standard 25 pin null modem D-shell RS-232 cable. Place the cable on the native serial port or on the 8 port or 16 port asynchronous adapter.
- Use a LAN connection (either Ethernet or Token Ring) through TCP/IP.

For each 3494 Enterprise Tape Library connected with an RS-232 cable, follow the directions in “Configuring the Serial Port” on page 299. For each Tape Library connected through TCP/IP, ensure that your system has access to the Library Manager on the 3494 Tape Library. Use the *ping* utility to verify that you have network connectivity to the Library Manager. Consult your network administrator if you need help with this task.

Identify the hostname of your workstation to the Enterprise Library Manager by following the instructions in the *IBM TotalStorage Enterprise Automated Tape Library Operator Guide*, in the **Add LAN Host** section.

Configuring the Serial Port

After the 3494 Tape Library is connected to the workstation, you must configure the serial port to which it is connected. Use *System Manager* to configure the serial port.

Select the following options:

1. Select **System** from the **Toolchest**.
2. Select **System Manager**.
3. Select **Hardware and Devices** from the **System Manager** window.
4. Select **Serial Device Manager**.
5. Select the right serial port in the **Serial Device Manager** window.
6. Click **Add...** to add or configure the serial port.
7. Follow the instructions to:
   a. Specify the type of serial device: **Input Device**
   b. Select the right port.
   c. Choose the name of the input device: **Other**
   d. Choose a name for the device.
   e. Click **OK**.

The serial port you have configured should show *SerialPort* under **Type**. By default, IRIX should set the port that you have just configured to the following settings that are needed to communicate with the 3494 Enterprise Tape Library:

Baud rate: 9600

Data bits: 8

Stop bits: 1

Parity: None

Login: Disabled

Defining the Library Device to the lmcpd

After the 3494 Enterprise Tape Library Daemon for SGI IRIX is installed, configure the attached Tape Libraries and start the daemon. Define the library devices to the daemon by editing the */etc/ibmatl.conf* file. The format of the file is:

```
Symbolic name     Connection type     Identifier     Alternate LAN Connection
```

Symbolic name is used to identify each tape library. The symbolic names must be unique across all libraries defined in the *ibmatl.conf* file.

Connection type is used to define the type of connection to the library (either RS-232 or TCP/IP). For RS-232 connections, this type is the device special file name of the TTY device (for example, */dev/ttyd1*). For TCP/IP connections, this type is the IP address of the Enterprise Library Manager. If the Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first IP address.

Identifier is used to specify a name by which the library manager identifies the host system. This identifier has a maximum length of eight characters. The symbolic network name of the host is usually the best name to use (although any name is acceptable). This parameter is used only to identify a particular host to the Enterprise Library Manager.

Alternate LAN Connection is used to specify an alternate IP address for an Enterprise High Availability LAN-attached configuration. If the High Availability Library has a dual LAN card installed, the second IP address of the dual LAN card should be entered after the first alternate LAN connection IP address.

Note: For RS-232 attachments, the alternate LAN connection is not applicable. Failover for RS-232 connections will proceed over the single serial line.

The following examples show how to define the library name:

```
tire            /dev/ttyd2                sitlab1
```

This stanza defines the *tire* library that is connected by */dev/ttyd2* (com2) to the workstation. For example:

```
3494a             9.115.32.100          jupiter
```

Note: The *ttydn* name is usually the special file name for a standard serial port on IRIX, where *n* is the com port number. For additional information, refer to the IRIX documentation.

This stanza defines the *3494a* library that is connected to the host. The address of the Enterprise Library Manager is *9.115.32.100*. The tape library uses the *jupiter* identifier for the host. For example:

```
libmgrc9          9.115.32.100          jupiter         9.115.23.54
```

This stanza defines the *libmgrc9* library that is connected through TCP/IP to the workstation. The IP addresses of the dual library managers are *9.115.32.100* and *9.115.23.54*.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury
```

This stanza defines libmgrc7 with a dual LAN card connected through TCP/IP to the workstation. The first address of the Library Manager on the 3494 Tape Library is 9.115.32.21 and the second address is 9.115.32.22. The library uses the mercury identifier for the host.

```
libmgrc7    9.115.32.21  9.115.32.22     mercury  9.115.26.13   9.115.26.14
```

This stanza defines the libmgrc7 High Availability library with dual LAN cards connected through TCP/IP to the workstation. The IP addresses of the first Library Manager are 9.115.32.21 and 9.115.32.22. The IP addresses of the second Library Manager are 9.115.26.13 and 9.115.26.14. The library uses the mercury identifier for the host.

Starting the 3494 Enterprise Tape Library Daemon

Start the 3494 Enterprise Tape Library Daemon for SGI IRIX by either restarting your computer or starting it manually from a command prompt by entering:

```
/etc/lmcpd
```

Problem Determination

A set of tools is provided with the software to determine if the*lmcpd* is functioning correctly.

Daemon Trace Facility: The following trace facility is available for the *lmcpd* daemon:

```
trcatl -[ald]
```

Arguments:

```
-a (trace all libraries defined in /etc/ibmatl.conf)
-l <name>(name of library)
-d (delete all log files; requires root authority)
```

Examples:

```
trcatl -a >/tmp/trace-out
trcatl -l libmgrc7
```

Note: The *-a* argument takes precedence over the *-l* argument.

3494 Library Emulation

The 3494 Library Emulation support provides the ability to use applications that were written for the 3494 API on host attached SCSI Medium Changer Libraries, specifically for migration to the 3584 Ultra Scalable Library with 3592 drives..

Overview

Existing customer scripts using the mtlib program and customer or ISV applications will require no or minor changes to use, depending on the 3494 functions being used. Customer and ISV applications will only need to be recompiled with the new libibm.o object module in order to use the emulation support. The mtlib 3494 user command line interface program that has the same syntax and output on every O/S platform will provide a common utility for SCSI Medium Changer Libraries also. Current O/S device driver utilities vary by syntax/output, require a knowledge of SCSI Medium Changers to use, and require multiple commands to perform a single mtlib command.

3494 Emulation Design

The 3494 Emulation support adds a libsmc extension to the current libibm.o object module that applications compile with for existing 3494 support. The libsmc extension contains the 3494 API Emulation support, which issues SCSI Medium changer commands to the library using the O/S SCSI Medium Changer device driver.

Figure 66 and Figure 67 on page 303 show the 3494/SMC data flow between existing applications and components and the new libsmc extension.

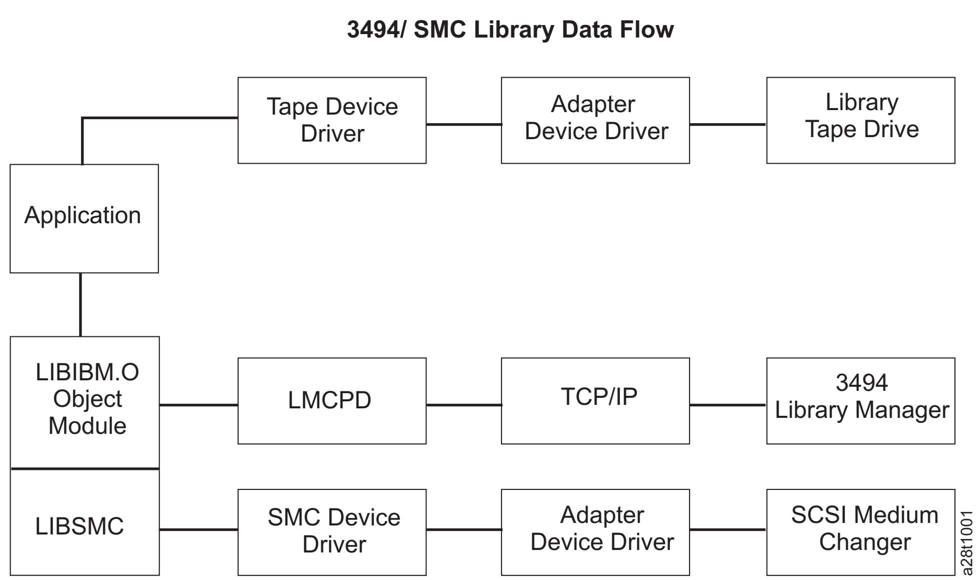


Figure 66. 3494/SMC Library Data Flow

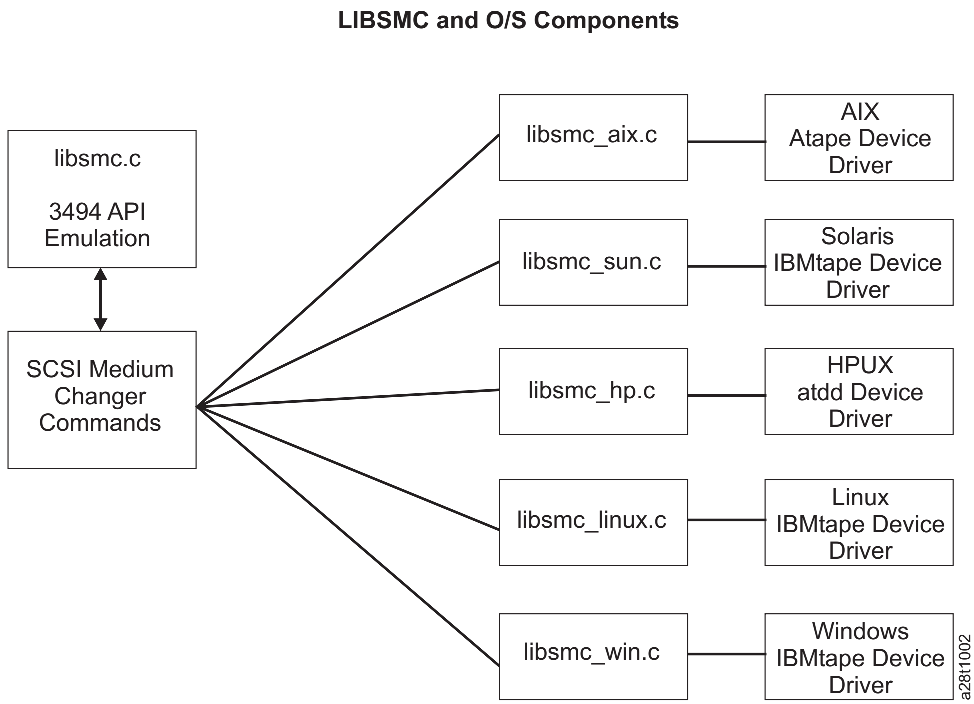


Figure 67. LIBSMC and OS Components

Using the 3494 API Emulation and MTLIB Program

Refer to the *IBM Tape Device Drivers Installation and User's Guide* and the *IBM Tape Device Drivers Programmer's Reference* for the complete description of 3494 library support and the mtlib program and syntax.

SMC Library Names

The library name on the open_ibmatl ("libname") or mtlib -l libname parameter determines whether a 3494 library or an SMC library is being used. If the library name is a logical O/S SMC library name then the libsmc support will be used. Otherwise, the lmcpd support will be used.

For example, SMC libraries on AIX are named smc0, smc1, and so on. To issue an **mtlib** query library command on AIX to a 3584 library named smc0:

```
mtlib -l smc0 -qL
```

To issue an **mtlib** command on Linux to a 3584 library named IBMchanger0:

```
mtlib -l IBMchanger0 -qL
```

Volume Categories

The 3494 library supports the grouping of one or more volumes into categories that can be assigned by an application and operations that can specify a category only rather than a specific volume. SCSI Medium Changers do not provide a similar function. All volumes in an SMC library are category 0000, the NULL category.

Because volume categories are not supported, the following API and **mtlib** command options cannot be used. Applications and mtlib scripts that use these functions might require changes.

- Reserve category mtlib -r option
- Release category mtlib -R option
- Change a volume category mtlib -C option
- Set category attribute mtlib -S option
- Assign a category mtlib -k option
- Mount from a category mtlib -m -s option

Asynchronous Library Operations

The 3494 library performs all mount, demount, audit, and eject operations asynchronously and initially returns a request id to the host for the operation. The request id can then be queried at any time to determine if the operation is still pending, completed, or is unknown. SMC libraries perform these operations synchronously and do not support a request id.

An application can select to wait for an asynchronous operation to complete by setting the API wait_flg to one or using mtlib without the -n option. The operation will be synchronous on both a 3494 and SMC library and no request id is returned to the application. These applications and mtlib scripts will not require any changes.

If an application selects to not wait for an asynchronous operation to complete by setting the API wait_flg to zero or using mtlib with the -n option, the operation on an SMC library will still be synchronous and request id 0 is returned to the application.

Most applications and scripts do not use the return request id to determine when the operation completes and use the no wait option for performance only. For example, using the no wait option on a demount operation so the application can continue while the library is demounting the volume. These applications and mtlib scripts will not require any changes but will take the same amount of time as using the wait option.

Applications that use the Query Message ID or mtlib -i option to determine when the return request id operation completes may require changes, such as polling for a mount issued with the no wait option since any request id will always be unknown on an SMC library because the operation has already completed.

Appendix A. Accessing Documentation and Software Online

IBM maintains the latest levels of TotalStorage tape drive and library device drivers and documentation on the Internet. Obtain them by accessing the following URL:

```
ftp://ftp.software.ibm.com/storage/devdrvr/tapedrivers.html
```

This web page has links to the device driver download web pages.

Information concerning supported fibre channel host bus adapters (HBAs) and associated HBA device drivers, firmware and BIOS levels can be obtained from the following URL:

```
http://www-03.ibm.com/systems/support/storage/config/hba/index.wss
```

You may also browse the FTP site and download drivers using one of the following URLs:

```
ftp://ftp.software.ibm.com/storage/devdrvr
ftp://207.25.253.26/storage/devdrvr
```

Or, you may access this information using anonymous FTP, as follows:

```
FTP site:   ftp.software.ibm.com
IP Address: 207.25.253.26
User ID:    anonymous
Password:   (Use your current e-mail address.)
Directory:  /storage/devdrvr
```

IBM provides Postscript- and PDF-formatted versions of its documentation in the */storage/devdrvr/Doc* directory. Postscript documents are gzip-ed.

The *IBM_Tape_Driver_IUG.pdf* file contains the current version of the *IBM Tape Device Drivers: Installation and User's Guide*.

The *IBM_Tape_Driver_PROGREF.pdf* file contains the current version of the *IBM Tape Device Drivers: Programming Reference*.

Device and Library Drivers for each supported platform can be found beneath */storage/devdrvr/* in the following directories:

```
AIX/
HPUX/
Linux/
SGI/
Solaris/
Tru64/
Windows/
```

For the most-current information for the device driver you are using, consult the README files in the directory pertaining to your device driver. For example, for information on exploiting the capacity scaling feature of your IBM TotalStorage Enterprise Tape Subsystem 3592, consult the README file.

There are numeric sequence numbers in each level of device and library driver, that is, *AIX/Atape.4.4.0.0.bin*. As newer levels of a driver are released, a higher numeric sequence is assigned.

Table 30 documents each driver by name and description:

Table 30. Driver Descriptions

Driver	Description
AIX/Atape.n.n.n.n.bin	AIX Device Driver (Atape)
AIX/atldd.n.n.n.n.bin	AIX 3494 Enterprise Tape Library Driver
HPUX/.../atdd.n.n.n.n.bin	HP-UX Device Driver
HPUX/lmcpd/lmcpd.hpux.n.n.n.n.bin	HP-UX 3494 Enterprise Library Driver
Linux/ibmatl/ibmatl.n.n.n.n.arch.rpm.bin	Linux 3494 Enterprise Library Driver
Linux/lin_tape_source-lin_taped/lin_tape-x.x.x.x-x.src.rpm.bin	Linux Device Driver (lin_tape) source code
Linux/lin_tape_source-lin_taped/lin_taped-x.x.x-*dist.arch*.rpm.bin	Linux lin_taped daemon program
Linux/lin_tape_source-lin_taped/IBMtapeutil/ IBMtapeutil.n.n.n.*arch*.tar.bin	Linux Tape Utility (IBMtapeutil)
SGI/ibmatl.n.n.n.n.tardist	SGI/IRIX Enterprise Library Driver
Solaris/IBMtape.n.n.n.n.bin	Solaris Device Driver (IBMtape)
Solaris/lmcpd.n.n.n.n.bin	Solaris 3494 Enterprise Library Driver
Tru64/ibmatl.tru64.n.n.n.n.kit.tar	Tru64 Enterprise Library Driver
Windows/ibmatl/ibmmatl.n.n.n.n.exe	32-bit Windows 3494 Enterprise Library Service
Windows/ibmatl/ibmatl.n.n.n.n.x86.exe	32-bit Windows 3494 Enterprise Library Service
Windows/ibmatl/ibmalt.n.n.n.n.i64.exe	64-bit Windows 3494 Library Service for IA64
Windows/ibmalt/ibmalt.n.n.n.n.x64.exe	64-bit Windows 3494 Library Service for AMD64 and EM64T
Windows/Win2000/Latest/IBMTape.x86_nnnn.zip	Windows Server 2000 Device Driver on x86
Windows/Win2003/Latest/IBMTape.x86_nnnn.zip	Windows Server 2003 32-bit Device Driver on x86
Windows/Win2003/Latest/IBMTape.i64_nnnn.zip	Windows Server 2003 64-bit Device Driver on IA64
Windows/Win2003Latest//IBMTape.x64_nnnn.zip	Windows Server 2003 64-bit Device Driver on AMD64 and EM64T
Windows/Win2008/Latest/IBMTape.x86_w08_nnnn.zip	Windows Server 2008 32-bit Device Driver on x86
Windows/Win2008/Latest/IBMTape.i64_w08_nnnn.zip	Windows Server 2008 64-bit Device Driver on IA64
Windows/Win2008Latest//IBMTape.x64_w08_nnnn.zip	Windows Server 2008 64-bit Device Driver on AMD64 and EM64T

Notes:
1. *dist* indicates a Linux distribution. *arch* indicates a machine architecture (for example, i386, ia64, s390).
2. The *n.n.n.* or *n.n.n.n* strings are replaced with digits on the FTP site to reflect the version of each driver.

Appendix B. Verifying Proper Attachment of Your Devices

Before you start to use your devices for production work with your applications, or if you encounter difficulties with your devices, you may want to verify that the hardware, connections, and device drivers are working together properly. Before you can do this, you must do the following:

1. Install your hardware as indicated in the appropriate hardware manuals.
2. Power On your hardware and verify that the hardware is functioning properly by executing commands from the front panel.
3. Attach your hardware to the host system as indicated in the appropriate hardware manuals and as indicated in the appropriate chapters from this manual.
4. Start your operating system as indicated in the appropriate chapters from this manual.
5. Log into the operating system as Administrator.
6. If there are device drivers being used by your device other than the ones documented in this manual, disable the other device drivers, and install or enable the drivers documented in this manual.
7. Follow the subsequent instructions for your host system.

AIX System

Tape Device Attachment Test

The following procedure tests the attachment of a tape device to an AIX system. The procedure assumes that your device is attached at */dev/rmt0* and that there is no cartridge in the drive.

When the **Hit <Enter> to Continue . . . ?** message is displayed or when you are told to enter information, press **Enter** after following the instruction. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Open an AIX window.
2. Enter *tapeutil*. A menu is displayed.
3. Enter **1** (Open a Device).
4. Enter */dev/rmt0* when prompted for the device name.
5. Enter **1** (Read/Write).
6. Enter **3** (Device Info).
7. Enter **5** (Inquiry). Specify **0** when prompted for an inquiry page. This concludes a very basic test of the device, SCSI connection, and the device driver. You may stop the test here or continue to perform a more complete test.
8. Enter **6** (Test Unit Ready) until no error occurs.
9. Enter **22** (Rewind).
10. Enter **30** (Read and Write Tests).
11. Enter **1** (Read/Write). Press the **Enter** key three times to accept the defaults and run the test.
12. Enter **31** (Unload Tape).

13. Enter **2** (Close a Device).
14. Enter **Q** (Quit Program).

Medium Changer Device Attachment Test

The following procedure tests the attachment of a medium changer device to an AIX system. The procedure assumes that your device is attached at */dev/rmt0*. You also need a cartridge in at least one of the slots.

When the **Hit <Enter> to Continue . . .?** message is displayed or when you are told to enter information, press **Enter** after following the instruction. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Open an AIX window.
2. Enter *tapeutil*. A menu is displayed.
3. Enter **1** (Open a Device).
4. Enter */dev/rmt0* when prompted for the device name.
5. Enter **1** (Read/Write).
6. Enter **3** (Device Info).
7. Enter **5** (Inquiry). Specify **0** when prompted for an inquiry page. This concludes a very basic test of the device, SCSI connection, and the device driver. You may stop the test here or continue to perform a more complete test.
8. Enter **6** (Test Unit Ready) until no error occurs.
9. Enter **18** (Initialize Element Status).
10. Enter **14** (Element Inventory).
11. From the output of the previous step, select a writable, scratch cartridge and determine its element ID. Also, select the element ID of an unoccupied slot.
12. Enter **16** (Move Medium), then supply the address of the cartridge followed by the address of the unoccupied slot. Verify that the cartridge moved.
13. Enter **14** (Element Inventory). Verify that the inventory was updated properly.
14. Enter **2** (Close a Device).
15. Enter **Q** (Quit Program).

3494 Enterprise Tape Library Attachment Test

The following procedure tests the attachment of a 3494 Enterprise Tape Library to an AIX system. The procedure assumes that your device is defined as */dev/lmcp0*.

1. Open an AIX window.
2. Enter mtlib -l */dev/lmcp0*-qI
3. Enter mtlib -l */dev/lmcp0* -qL

Tru64 System

Tape Device Attachment Test

The following procedure tests the attachment of a tape device to a Tru64 system. The procedure assumes that your device is attached at */dev/ntape/tape0*.

When you are asked to enter information, press the **Enter** key after following the instructions. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Logon to the system as root, or use the *su* command to gain superuser privileges.
2. Bring up a shell prompt.
3. Enter */sbin/scu-f/dev/ntape/tape0* (Open the Tape Device).
4. Enter *tur* (Test Unit Ready)
5. Enter *Show Inquiry* (Inquiry data).
6. Enter *quit* to quit *scu*.

Medium Changer Device Attachment Test

The following procedure tests the attachment of a medium changer device to a Tru64 system. The procedure assumes that your device is attached at */dev/changer/mc0*. You also need a cartridge in at least one of the slots.

When you are asked to enter information, press the **Enter** key after following the instructions. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Logon to the system as root, or use the *su* command to gain superuser privileges.
2. Bring up a shell prompt.
3. Enter */sbin/scu -f /dev/changer/mc0* (Open the medium changer device).
4. Enter *show inquiry* (Inquiry Data).

   This concludes a very basic test of the device, SCSI connection, and the device driver. You may stop the test here or continue to perform a more complete test.
5. Enter *tur* (Test Unit Ready) until no error occurs.
6. Enter *show element* (Element Inventory).
7. From the output in the previous step, select a writable, scratch cartridge and determine its element address. Also select the element address of an unoccupied slot.
8. Enter *move medium source Source_address destination Destination_address* to move the cartridge from the source element address to the destination element address. Verify that the cartridge moved.
9. Enter *show element* (Element Inventory) to verify that the inventory was updated properly.
10. Enter *quit* (Quit *scu*).

3494 Enterprise Tape Library Attachment Test

The following procedure tests the attachment of a 3494 Enterprise Tape Library to a Tru64 system. The procedure assumes that your device is defined in */etc/ibmatl.conf* as L3494.

1. Open a window.
2. Enter mtlib -l *L3494* -qI
3. Enter mtlib -l *L3494* -qL

HP-UX System

Tape Device Attachment Test

The following procedure tests the attachment of a tape device to an HP-UX system. The procedure assumes that your device is attached at */dev/rmt/0m*.

When you are told to enter information, press the **Enter** key after following the instruction. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Bring up a shell prompt.
2. Enter *tapeutil*. A menu is displayed.
3. Enter **1** (Open a Device).
4. Enter */dev/rmt/0m* when prompted for the device name.
5. Enter **1** (Read/Write).
6. Enter **D** (Device Info).
7. Enter Inquiry Option (Inquiry).
8. Enter **Q** to quit *tapeutil*.

Autochanger Device Attachment Test

The following procedure tests the attachment of an autochanger device to an HP-UX system. The procedure assumes that your device is attached at */dev/rmt/0chng* and that the ACF is in Random mode. You also need a cartridge in at least one of the slots.

When you are told to enter information, press **Enter** after following the instruction. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Bring up a shell prompt.
2. Enter *tapeutil*. A menu is displayed.
3. Enter **1** (Open a Device).
4. Enter */dev/rmt/0chng* when prompted for the device name.
5. Enter **1** (Read/Write).
6. Enter **D** (Device Info).
7. Enter **5** (Inquiry). This concludes a very basic test of the device, SCSI connection, and the device driver. You may stop the test here or continue to perform a more complete test.
8. Enter Test Unit Ready option until no error occurs.
9. Enter Element Inventory option.
10. From the output in the previous step, select a writable, scratch cartridge and determine its element ID. Also, select the element ID of an unoccupied slot.
11. Enter Move Medium option, then supply the address of the cartridge and the address of the unoccupied slot. Verify that the cartridge moved.
12. Enter Element Inventory option. Verify that the inventory was updated properly.
13. Enter **2** (Close a Device).
14. Enter **Q** (Quit Program).

3494 Enterprise Tape Library Attachment Test

The following procedure tests the attachment of a 3494 Enterprise Tape Library to an HP-UX system. The procedure assumes that your device is defined in */etc/ibmatl.conf* as *L3494*.

1. Bring up a shell prompt.
2. Enter *mtlib -l L3494 -qI*
3. Enter *mtlib -l L3494 -qL*

Linux System

Issue the following command to verify if your host system has recognized all the attached IBM tape and medium changer devices after you installed the lin_tape device driver:

```
cat /proc/scsi/scsi
```

You can follow the instructions below to test the attachment.

Tape Device Attachment Test

The following procedure tests the attachment of a tape device to a Linux system. The procedure assumes that your device is attached at */dev/IBMtape0* and that there is no cartridge in the drive.

When the **Hit<Enter>to Continue...** message is displayed or when you are told to enter information, press **Enter** after following the instruction. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Open a Linux window.
2. If you want to perform a more complete test, mount a writable scratch cartridge manually into the driver, or, if your device has an autoloader attached, follow the ″Medium Changer Device Attachment Test″ in the next section to mount a writable scratch cartridge.
3. Enter *IBMtapeutil*. A menu is displayed.
4. Enter **1** (Open a Device).
5. Enter */dev/IBMtape0* when prompted for the device name.
6. Enter **1** (Read/Write).
7. Enter **3** (Inquiry). Specify 0 when prompted for an inquiry page. This concludes a very basic test of the device, SCSI connection, and the device driver. You may stop the test here or continue to perform a more complete test as follows:
8. Enter **4** (Test Unit Ready) until no error occurs.
9. Enter **20** (Rewind).
10. Enter **28** (Read and Write Tests).
11. Enter **1** (Read/Write). Press the **Enter** key three times to accept the defaults and run the test.
12. Enter **38** (Unload Tape).
13. Enter **2** (Close a Device).
14. Enter **Q** (Quit Program).

Medium Changer Device Attachment Test

The following procedure tests the attachment of a medium changer device to a Linux system. The procedure assumes that your device is attached at */dev/IBMchanger0*. You also need a cartridge in at least one of the slots.

When the **Hit <Enter> to Continue ...** message is displayed or when you are told to enter information, press **Enter** after following the instructions. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Open a Linux window.
2. Enter *IBMtapeutil*. A menu is displayed.

3. Enter **1** (Open a Device).
4. Enter */dev/IBMchanger0* when prompted for the device name.
5. Enter **3** (Inquiry). Specify 0 when prompted for an inquiry page. This concludes a very basic test of the device, SCSI connection, and the device driver. You may stop the test here or continue to perform a more complete test as follows:
6. Enter **4** (Test Unit Ready).
7. Enter **60** (Element Information).
8. Enter **62** (Element Inventory).
9. From the output of the previous step, select a writable, scratch cartridge and determine its element ID. Also, select the element ID of an unoccupied slot.
10. Enter **64** (Move Medium), then supply the address of the cartridge, followed by the address of the unoccupied slot. Verify that the cartridge moved.
11. Enter **62** (Element Inventory). Verify that the inventory was updated correctly.
12. Enter **2** (Close a Device).
13. Enter **Q** (Quit Program).

3494 Enterprise Tape Library Attachment Test

The following procedure tests the attachment of a 3494 Enterprise Tape Library to a Linux system. The procedure assumes that your device is defined in */etc/ibmatl.conf* as *L3494*.

1. Bring up a shell prompt.
2. Enter *mtlib -l L3494 -ql*
3. Enter *mtlib -l L3494 -qL*

SGI/IRIX System

3494 Enterprise Tape Library Attachment Test

The following procedure tests the attachment of a 3494 Enterprise Tape Library to a SGI/IRIX system. The procedure assumes that your device is defined in */etc/ibmatl.conf* as *L3494*.

1. Bring up a shell prompt.
2. Enter *mtlib -l L3494 -qI*
3. Enter *mtlib -l L3494 -qL*

Solaris System

Tape Device Attachment Test

The following procedure tests the attachment of a tape device to a Sun Solaris system. The procedure assumes that your device is attached at */dev/rmt/0st*. When you are told to enter information, press **Enter** after following the instruction. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Bring up a shell prompt.
2. Enter *tapeutil*. A menu is displayed.
3. Enter **1** (Open a Device).
4. Enter */dev/rmt/0st* when prompted for the device name.
5. Enter **1** (Read/Write).
6. Enter **D** (Device Info).
7. Enter **10** (Inquiry).
8. Enter **Q** to quit *tapeutil*.

Autochanger Device Attachment Test

The following procedure tests the attachment of an autochanger device to a Sun Solaris system. The procedure assumes that your device is attached at */dev/rmt/0smc* and that the ACF is in Random mode. You also need a cartridge in at least one of the slots.

When you are told to enter information, press **Enter** after following the instruction. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Bring up a shell prompt.
2. Enter *tapeutil*. A menu is displayed.
3. Enter **1** (Open a device).
4. Enter */dev/rmt/0smc* when prompted for the device name.
5. Enter **1** (Read/Write).
6. Enter **D** (Device Info).
7. Enter **5** (Inquiry). This concludes a very basic test of the device, SCSI connection, and the device driver. You may stop the test here or continue to perform a more complete test.
8. Enter **6** (Test Unit Ready) until no error occurs.
9. Enter **17** (Element Inventory).
10. From the output in the previous step, select a writable, scratch cartridge and determine its element ID. Also, select the element ID of an unoccupied slot.
11. Enter **14** (Move Medium), then supply the address of the cartridge and the address of the unoccupied slot. Verify that the cartridge moved.
12. Enter **17** (Element Inventory).
13. Verify that the inventory was updated properly.
14. Enter **2** (Close a Device).
15. Enter **Q** (Quit Program).

3494 Enterprise Tape Library Attachment Test

The following procedure tests the attachment of a 3494 Enterprise Tape Library to a Sun Solaris system. The procedure assumes that your device is defined in */etc/ibmatl.conf* as *L3494*.

1. Bring up a shell prompt.
2. Enter *mtlib -l L3494 -qI*
3. Enter *mtlib -l L3494 -qL*

Microsoft Windows System

Tape Device Attachment Test

The following procedure tests the attachment of a tape device to a Microsoft Windows system. The procedure assumes that your device is attached at *tape0* and that there is no cartridge in the drive.

When the message **Return to continue:** message is displayed or when you are told to enter information, press **Enter** after following the instruction. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Open a Windows command shell window.
2. For Windows NT users, if you want to perform a more complete test and your device has an autoloader, use the instructions in "Autochanger Device Attachment Test - Windows NT Only" on page 318 to mount a writable scratch cartridge in the drive. This is accomplished by following the steps in the procedure, except that in steps 11 and 12, select the element ID of a drive rather than the unoccupied slot ID.

   If you want to perform a more complete test and your device does not have an autoloader, mount a writable scratch cartridge manually into the drive.
3. Enter *ntutil*.
4. Select **1** (Manual test).
5. Enter **50** (poll registered devices). All devices detected by adapters (that were attached and powered-On at system start time) should be displayed.
6. Enter **20** (open).
7. Enter **1** (Read/Write).
8. Enter **49** (inquiry). Reply with **0** (zero) to the *Drive=0, Library=1* prompt. This step concludes a very basic test of the device, adapter connection, and the device driver. You may stop the test here or continue to perform a more complete test.
9. Enter **39** (test unit ready) until no error occurs. Each time you are prompted with a *Drive=0, Library=1* prompt, reply with **0** (zero).
10. Enter **31** (rewind).
11. Enter **87** (Read/Write Test). Enter **Y** in response to the confirmation message.
12. Enter **33** (unload).
13. Enter **21** (close).
14. Enter **99** (return to main menu).
15. Enter **9** (Exit *ntutil*).

Autochanger Device Attachment Test - Windows NT Only

The following procedure tests the attachment of an autochanger device to a Windows NT system. The procedure assumes that your device is attached at *lb1.0.0.1* and that the tape device is attached at *tape0*. You also need a cartridge in at least one of the slots.

When the **Return to continue:** message is displayed or when you are told to enter information, press **Enter**after following the instruction. When the utility displays information, use the appropriate hardware manual to verify that the information is reasonable.

1. Open a Windows NT command shell window.

2. Enter *ntutil*.
3. Select **1** (Manual test).
4. Enter **50** (poll registered devices). All devices detected by adapters (that were attached and powered-On at system start time) should be displayed. You should see *lb1.0.0.1* in the list of found devices.
5. Enter **8** (Library Mode).
6. Enter **20** (open).
7. Enter **1** (Read/Write).
8. Enter **49** (inquiry). Reply with **0** (zero) to the *Drive=0, Library=1* prompt. This step concludes a very basic test of the device, adapter connection, and the device driver. You may stop the test here or continue to perform a more complete test.
9. Enter **39** (test unit ready) until no error occurs. Each time you are prompted with a *Drive=0, Library=1* prompt, reply with **1** (zero).
10. Enter **12** (*ioctl* initialize element status).
11. Enter **10** (*ioctl* return library inventory).
12. From the output in the previous step, select a writable, scratch cartridge and determine its element ID. Also, select the element ID of an unoccupied slot.
13. Enter **11** (*ioct*l move medium), then supply the address of the cartridge and the address of the unoccupied slot. Verify that the cartridge moved.
14. Enter **10** (*ioctl* return library inventory). Verify that the inventory was updated properly.
15. Enter **21** (close).
16. Enter **99** (return to main menu).
17. Enter **9** (Exit *ntutil*).

3494 Enterprise Tape Library Attachment Test

The following procedure tests the attachment of a 3494 Enterprise Tape Library to a Windows system. The procedure assumes that your device is defined in *c:\winnt\ibmatl.conf* as *L3494*.

1. Open a Windows command shell window.
2. Enter *mtlib -l L3494 -qI*
3. Enter *mtlib -l L3494 -qL*

Verifying Proper Attachment of Your Devices

Appendix C. Managing the Microcode on the IBM Tape Drive

Microcode is computer software that is stored in nonvolatile storage on your tape device or library hardware. It controls the operation of your hardware. When your tape device or library hardware was manufactured, a microcode load was installed and shipped with your device.

If you are having trouble with your hardware, IBM service personnel will ask what level of microcode you have on your hardware. If they believe you need a new level of microcode, they may instruct you to install a newer level of microcode on your hardware. They will provide you with updated microcode.

You can query the current level of microcode by issuing commands on the front panel of your hardware. Consult the appropriate hardware reference manual for specific instructions on querying your microcode level.

If your device is connected to a host system that has device or library support, you can also query the last four digits of the current level of microcode using software, refer to Table 31. The unit must be powered On, configured properly, and ready. For additional information, refer to the appropriate chapter in this document (based on the operating system/platform) for details on how to have the device ready.

Table 31. Query for Current Microcode Level

OS	Command
AIX	Use the *tapeutil* command with the *vpd* subcommand. Refer to the Revision Level output field.
Tru64	Use the *IBMrasutil-f* device command where device is the device special file of the attached tape or medium changer device.
HP-UX	Use the *tapeutil -f drive -o qmc -v* command, where *drive* is the device special file of the attached tape device.
Linux	Use the *IBMtapeutil –f drive inquiry* command where drive is the device special file of the attached tape device.
Solaris	Use the *tapeutil -f drive -o qmc -v* command, where *drive* is the device special file of the attached tape device.
Microsoft Windows	Use the *ntutil* command with the inquiry subcommand. Refer to the **Microcode Revision Level** output field.

The following instructions will guide you to install another version of microcode on your tape drive:

1. Ensure that the tape drive is connected to a host system and that the tape device driver is powered-On and configured properly with no tape cartridge in the drive. Follow the instructions in Appendix B, “Verifying Proper Attachment of Your Devices,” on page 307 to ensure that the drive is configured properly and ready.
2. Follow the platform-specific instructions.

AIX System

1. Open an AIX window.
2. Enter *tapeutil*. A menu is displayed.
3. Enter **4** (Tape Drive Service Aids).
4. Select **Microcode Load**.
5. Select the device special file identifier for the device on which to load the microcode, then press **Enter**.
6. Specify the special file from where the microcode image is to be read and press **F7** (Commit) to load the microcode onto the drive. When the loading is complete, press **F10** (Exit) to return to *tapeutil*.
7. Enter **Q** (Quit Program).

Tru64 System

1. Bring up a shell prompt.
2. Enter *IBMrasutil*. It will prompt you to enter a device filename.
3. Enter */dev/device name* when prompted.
4. Enter **2** (Update Microcode).
5. Specify the special file from which the microcode image is to be read, then press the **Enter** key.
6. Enter **3** (Quit).

HP-UX System

This procedure assumes that the */dev/rmt/0m* device is being updated.

1. Bring up a shell prompt.
2. Enter *tapeutil*. A menu is displayed.
3. Enter **1** (Open a Device).
4. Enter */dev/rmt/0m* when prompted for the device name.
5. Enter **1** (Read/Write).
6. Select Download Microcode option.
7. Specify the special file from where the microcode image is to be read, then press the **Enter** key.
8. Enter **Q** (Quit Program).

Linux System

This procedure assumes that the */dev/IBMtape0* tape device is being updated.

1. Open a Linux window.
2. Enter *IBMtapeutil*. A menu is displayed.
3. Enter **1** (Open a Device).
4. Enter */dev/IBMtape0* when prompted for the device name.
5. Enter **1** (Read/Write).
6. Select **72** (Load Ucode).
7. Specify the file name from which the microcode image is to be read, then press the **Enter** key.
8. Enter **Q** (Quit Program).

Solaris System

This procedure assumes that the */dev/rmt/0st* tape device is being updated.

1. Bring up a shell prompt.
2. Enter *tapeutil*. A menu is displayed.
3. Enter **1** (Open a Device).
4. Enter */dev/rmt/0st* when prompted for the device name.
5. Enter **1** (Read/Write).
6. Select **7** (Download Microcode).
7. Specify the special file from where the microcode image is to be read, then press the **Enter** key.
8. Enter **Q** (Quit Program).

Microsoft Windows System

This procedure assumes that the new microcode is stored in *c:\mydata\d0i9_430.fmr* and that the drive at *tape0* is being updated. The name of the file containing the microcode must have the *fmr* extension.

1. Open a Windows command shell window.
2. Change to the *c:\mydata* directory.
3. Enter *ntutil*.
4. Select **1** (Manual test).
5. Enter **20** (open).
6. Enter **1** (Read/Write).
7. Enter **6** (test unit ready) until no error occurs.
8. Enter **82** (Update Code).
9. Enter *d0i9_430*.
10. Enter **21** (close).
11. Enter **99** (return to main menu).
12. Enter **9** (Exit *ntutil*).

Notices

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

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

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:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

The following paragraph does not apply to the United Kingdom or any other country (or region) where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states (or regions) do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or program(s) described in this publication at any time without notice.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries (or regions), or both:

AIX	Netfinity	System p	xSeries
AIX 5L	POWER5	System Storage	z9
FICON	pSeries	System x	zSeries
IBM	RS/6000	System z	
IBMLink	S/390	Tivoli	

Magstar	StorageSmart	TotalStorage
Micro Channel	StorWatch	Virtualization Engine

Adobe and Acrobat are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Itanium, and Pentium are trademarks of Intel Corporation in the United States, other countries (or regions), or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and Windows 2000 are trademarks of Microsoft Corporation in the United States, other countries (or regions), or both.

UNIX is a registered trademark of The Open Group in the United States and other countries (or regions).

Other company, product, and service names may be trademarks or service marks of others.

Index

Numerics

A

T

U

V

W

Readers' Comments — We'd Like to Hear from You

IBM Tape Device Drivers
Installation and User's Guide

Publication No. GC27-2130-06

We appreciate your comments about this publication. Please comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. The comments you send should pertain to only the information in this manual or product and the way in which the information is presented.

For technical questions and information about products and prices, please contact your IBM branch office, your IBM business partner, or your authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you. IBM or any other organizations will only use the personal information that you supply to contact you about the issues that you state on this form.

Comments:

Thank you for your support.

Send your comments to the address on the reverse side of this form.

If you would like a response from IBM, please fill in the following information:

Name	Address
Company or Organization	
Phone No.	E-mail address

Readers' Comments — We'd Like to Hear from You
GC27-2130-06

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Information Development
Department GZW
9000 South Rita Road
Tucson, Arizona 85775-4401

Fold and Tape | **Please do not staple** | Fold and Tape

Cut or Fold
Along Line

Printed in USA

GC27-2130-06

Spine information:

IBM Tape Device Drivers

IBM Tape Device Drivers Installation and User's Guide