VERITAS NetBackup™ 4.5

Media Manager Device Configuration Guide

for UNIX

March 2002
30-000522-011

VERITAS

Disclaimer

The information contained in this publication is subject to change without notice. VERITAS Software Corporation makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. VERITAS Software Corporation shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this manual.

Copyright

VERITAS Software Corporation
350 Ellis Street
Mountain View, CA 94043
USA
Phone 650–527–8000
Fax 650-527-2908
www.veritas.com

Contents

About This Guide

Introduction

This guide contains configuration information that VERITAS has found useful when adding storage peripherals to UNIX servers (device hosts) controlled by Media Manager. See the NetBackup release notes for information on the supported UNIX server platforms.

Media Manager is the component of NetBackup DataCenter, NetBackup BusinesServer, and VERITAS Storage Migrator that manages devices and media.

This guide is intended for use with the NetBackup DataCenter and the NetBackup BusinesServer products. In this guide, the term *NetBackup* refers to NetBackup DataCenter and NetBackup BusinesServer (where applicable).

VERITAS Storage Migrator is not supported with NetBackup BusinesServer.

Audience

The intended audience for this guide is the system administrator responsible for adding the storage peripherals, and assumes a thorough knowledge of UNIX system and device configuration.

Scope

This guide is intended to be used with the NetBackup DataCenter and NetBackup BusinesServer products.

The information in this guide supplements the manuals provided by UNIX hardware and operating system vendors.

Organization

"How To Use This Guide" on page 1 provides important instructions for using this guide that you should read first.

This guide contains a chapter for each of the UNIX server platforms that are supported by NetBackup as follows:

- "Sun4/SPARC Running Solaris 2.6/7/8/9" on page 3 provides configuration information for Sun device hosts.
- "IBM RS6000 Running AIX 4.3.3.10/5.1" on page 29 provides configuration information for IBM RS6000 device hosts.
- "HP9000 Running HP-UX 11.0/11.11" on page 57 provides configuration information for HP9000 device hosts.
- "IRIX 6.5.10-6.5.14" on page 79 provides configuration information for SGI device hosts.
- "Compaq Alpha Running TRU64 UNIX 4.0F/4.0G" on page 97 provides configuration information for Compaq Alpha device hosts.
- "Compaq Alpha Running TRU64 UNIX 5.0a/5.1/5.1a" on page 109 provides configuration information for Compaq Alpha device hosts.
- "Intel Hosts Running Red Hat Linux 6.2/7.0/7.1" on page 121 provides configuration information for Intel-based Linux device hosts.
- "NCR Running SVR4MP-RAS 3.02" on page 127 provides configuration information for NCR device hosts.
- "Sequent Running DYNIX/ptx 4.4.2/4.4.4-4.4.8/4.5/4.5.2" on page 131 provides configuration information for Sequent device hosts.

In addition to these chapters, there is a glossary of terms that you may encounter in this guide, and an index.

Using This Guide

Each UNIX platform that is supported by NetBackup DataCenter or NetBackup BusinesServer as a media server is described in a separate chapter in this guide. You should have to use only the chapters for the platforms on which you are configuring devices for Media Manager.

Portions of this guide include topics and examples that may *not* be applicable to your system hardware configuration.

Configuration file settings found in portions of this guide were tested and are known to work, but other configuration settings may also work.

It is important to refer to the VERITAS support web site (`http://www.support.veritas.com`) to determine which Media Manager robot types, robots, and drives are supported for your NetBackup product before using this guide.

For NetBackup DataCenter

All of the chapters in this guide are applicable for NetBackup DataCenter.

For NetBackup BusinesServer

The following chapters in this guide are applicable for NetBackup BusinesServer:

- "Sun4/SPARC Running Solaris 2.6/7/8/9" on page 3
- "HP9000 Running HP-UX 11.0/11.11" on page 57
- "Intel Hosts Running Red Hat Linux 6.2/7.0/7.1" on page 121

Related Documents

NetBackup documents that may be useful are listed below. For a complete list of related documents, see the NetBackup release notes. Depending on your configuration, other documents may also be required.

- *NetBackup BusinesServer Getting Started Guide for UNIX*

  Gives you the information you need to quickly get NetBackup BusinesServer server software installed and running. This is the first document that you should read after opening the NetBackup BusinesServer package.

- *NetBackup BusinesServer Media Manager System Administrator's Guide for UNIX*

  Explains how to configure and manage the storage devices and media on UNIX servers running NetBackup BusinesServer. Media Manager is part of the NetBackup BusinesServer product.

- *NetBackup Installation Guide for UNIX*

  Explains how to install NetBackup DataCenter software on UNIX-based platforms.

- *NetBackup DataCenter Media Manager System Administrator's Guide for UNIX*

  Explains how to configure and manage the storage devices and media on UNIX servers running NetBackup DataCenter. Media Manager is part of the NetBackup DataCenter product.

- *NetBackup Release Notes for UNIX and Windows*

  Provides important information about NetBackup DataCenter and BusinesServer products on UNIX- and Windows-based servers, such as the platforms and operating systems that are supported and operating notes that may not be in the NetBackup manuals or the online help.

- *NetBackup SAN Shared Storage Option System Administrator's Guide for UNIX and Windows*

  Provides information on installing and configuring NetBackup Shared Storage Option (SSO) software on UNIX and Windows-based servers. SSO is an extension to tape drive allocation and configuration for NetBackup DataCenter Media Manager.

- *NetBackup Troubleshooting Guide for UNIX*

  Provides troubleshooting information for UNIX-based NetBackup DataCenter and BusinesServer products.

VERITAS Storage Migrator is not supported with the NetBackup BusinesServer product.

- *VERITAS Storage Migrator Release Notes for UNIX*

  Provides information such as the platforms and operating systems that are supported and operating notes that may not be in the Storage Migrator manuals.

- *VERITAS Storage Migrator System Administrator's Guide for UNIX*

  Explains how to configure and manage Storage Migrator on a UNIX system.

Accessibility

NetBackup contains features that make the user interface easier to use by people who are visually impaired and by people who have limited dexterity. Accessibility features include:

- Support for assistive technologies such as screen readers and voice input (Windows servers only)
- Support for keyboard (mouseless) navigation using accelerator keys and mnemonic keys

For more information, see the NetBackup system administrator's guide.

Conventions

The following explains typographical and other conventions used in this guide.

Type Style

Typographic Conventions

Typeface	Usage
`Bold fixed width`	Input. For example, type **`cd`** to change directories.
`Fixed width`	Paths, commands, filenames, or output. For example: The default installation directory is `/opt/VRTSxx`.
Italics	Book titles, new terms, or used for emphasis. For example: *Do not* ignore cautions.
Sans serif (italics)	Placeholder text or variables. For example: Replace *filename* with the name of your file.
Serif (no italics)	Graphical user interface (GUI) objects, such as fields, menu choices, etc. For example: Enter your password in the **Password** field.

Notes and Cautions

Note This is a Note. Notes are used to call attention to information that makes using the product easier or helps in avoiding problems.

Caution This is a Caution. Cautions are used to warn about situations that could cause data loss.

Key Combinations

Some keyboard command sequences use two or more keys at the same time. For example, holding down the **Ctrl** key while pressing another key. Keyboard command sequences are indicated by connecting the keys with a plus sign. For example:

Press Ctrl+t

Command Usage

The following conventions are frequently used in the synopsis of command usage.

brackets [ ]

The enclosed command line component is optional.

Vertical bar or pipe (|)

Separates optional arguments from which the user can choose. For example, when a command has the following format:

`command` *arg1*|*arg2*

the user can use either the *arg1* or *arg2* variable.

Terms

The terms listed in the table below are used in the VERITAS NetBackup documentation to increase readability while maintaining technical accuracy.

Term	Definition
Microsoft Windows, Windows	Terms used as nouns to describe a line of operating systems developed by Microsoft, Inc. A term used as an adjective to describe a specific product or noun. Some examples are: Windows 95, Windows 98, Windows NT, Windows 2000, Windows servers, Windows clients, Windows platforms, Windows hosts, and Windows GUI. Where a specific Windows product is identified, then only that particular product is valid with regards to the instance in which it is being used. For more information on the Windows operating systems that NetBackup supports, refer to the VERITAS support web site at `http://www.support.veritas.com`.
Windows servers	A term that defines the Windows server platforms that NetBackup supports; those platforms are: Windows NT and Windows 2000.
Windows clients	A term that defines the Windows client platforms that NetBackup supports; those platforms are: Windows 95, 98, ME, NT, 2000, XP (for 32- and 64-bit versions), and LE.

Getting Help

For updated information about this product, including system requirements, supported platforms, supported peripherals, and a list of current patches available from Technical Support, visit our web site:

```
http://www.support.veritas.com/
```

VERITAS Customer Support has an extensive technical support structure that enables you to contact technical support teams that are trained to answer questions to specific products. You can contact Customer Support by sending an e-mail to support@veritas.com, or by finding a product-specific phone number from the VERITAS support web site. The following steps describe how to locate the proper phone number.

1. Open `http://www.support.veritas.com/` in your web browser.
2. Click **Contact Support**. The *Contacting Support Product List* page appears.
3. Select a product line and then a product from the lists that appear. The page will refresh with a list of technical support phone numbers that are specific to the product you just selected.

How To Use This Guide 1

Before You Start Configuring Devices

Observe the following important points before using this guide to perform device configurations.

Device Configuration Sequence

Use the following sequence when configuring your devices:

1. Physically attach the storage devices to the device host and perform any configuration steps specified by the device or operating system vendor.

   See the appropriate chapter of this guide for your UNIX platform.

2. Create any required system device files for the drives and robotic control. This is usually done during installation. Device files are created automatically on some UNIX platforms.

   See the appropriate chapter of this guide for your UNIX platform.

3. Use one of the available Media Manager configuration interfaces to add the storage devices to your Media Manager configuration.

   See the NetBackup Media Manager system administrator's guide for instructions.

Considerations When Using This Guide

- This guide is intended for use with the NetBackup DataCenter and NetBackup BusinesServer products. In this guide, the term *NetBackup* refers to NetBackup DataCenter *and* NetBackup BusinesServer.
- It is important to read the **Before You Start** section in each chapter of this guide that you use. The **Before You Start** section provides any important platform-specific instructions. This section may also contain specific instructions or limitations pertaining to NetBackup BusinesServer.

- Portions of this guide include hardware-specific topics and examples that may *not* be applicable to your system hardware configuration.
- To help avoid configuration errors, you can copy configuration examples from the text version of this guide, that is in the following file: `/usr/openv/volmgr/MediaMgr_DeviceConfig_Guide.txt`. This file is installed along with the NetBackup Media Manager software.

Read the NetBackup Release Notes

Refer to the NetBackup release notes to determine the UNIX platforms that are supported on NetBackup BusinesServer. This dictates the chapters of this guide that are applicable for the NetBackup BusinesServer product and which chapters are only applicable for the NetBackup DataCenter product.

Each UNIX platform that is supported by NetBackup as a media server is described in a separate chapter in this guide. You should have to use only the chapters for the platforms on which you are configuring devices for Media Manager.

See “Using This Guide” on page xii for more information on applicable chapters for NetBackup DataCenter and NetBackup BusinesServer.

See the VERITAS Support Web Site

Visit the VERITAS support web site (`http://www.support.veritas.com`) to determine which Media Manager robot types, robots, and drives are supported on

- Your UNIX platform.
- Your NetBackup product (DataCenter or BusinesServer).

Configuration Cautions

- Starting with release 4.5, NetBackup uses SCSI reserve / release to improve data integrity. SCSI reserve / release operates at the SCSI target level and depends on the fibre-to-scsi bridge hardware working correctly.

  The use of SCSI reserve / release is *enabled* by default, but can be disabled using the NetBackup Administration Console configuration GUI.
- VERITAS does not recommend or support the use of single-ended to differential SCSI converters on Media Manager controlled devices. You may encounter problems if you use these converters.

Sun4/SPARC Running Solaris 2.6/7/8/9 2

This chapter explains how to configure devices for use with Media Manager on a Sun4/SPARC platform. Configure drives and robots using one of the available Media Manager administrative interfaces.

The major topics included are as follows:

- Before You Start
- Preventing Possible System Problems
- SSO Configurations With More Than 16 Tape Drives
- Understanding the SCSI Passthru Drivers
- Configuring SG and ST Drivers
- Configuring the Sun StorEdge Network Foundation HBA Driver
- Configuring Third-Party Fibre Channel HBA Drivers
- Configuring Robotic Controls
- Configuring Tape Drives
- Configuring Optical Disk Drives
- Command Summary

Before You Start

Observe the following points when performing the configurations described in this chapter:

- When configuring devices, you should attach all peripherals and reboot the system with the reconfigure option (`boot -r` or `reboot -- -r`).
- When removing or replacing adapter cards, remove all device files previously associated with the adapter card.

- If you use the Automated Cartridge System (ACS) robotic software, you must ensure that the SunOS/BSD Source Compatibility Package is installed, so that the ACS software can make use of shared libraries in `/usr/ucblib`.

If You Are Using NetBackup BusinesServer

Portions of this chapter include configuration topics and examples for peripherals that are not supported in NetBackup BusinesServer. It is important to refer to the VERITAS support web site to determine which Media Manager robot types, robots, and drives are supported for NetBackup BusinesServer, before using this chapter.

Topics Not Applicable to NetBackup BusinesServer

- "SSO Configurations With More Than 16 Tape Drives" on page 4.
- "Configuring Optical Disk Drives" on page 24.

Preventing Possible System Problems

When system memory gets low, Solaris unloads unused drivers from memory and reloads drivers as needed. Tape drivers are a frequent candidate for unloading, since they tend to be less heavily used than disk drivers. Depending on the timing of these unload and load events for the st (Sun), sg (VERITAS), and fibre channel drivers, various problems may result. These problems can range from devices "disappearing" from a SCSI bus to system panics.

VERITAS recommends adding the following `forceload` statements to the `/etc/system` file. These statements prevent the st and sg drivers from being unloaded from memory.

```
forceload: drv/st
forceload: drv/sg
```

Other statements may be necessary for various fibre channel drivers, such as the following example for JNI:

```
forceload: drv/fcaw
```

SSO Configurations With More Than 16 Tape Drives

Changes in tape device status may not be visible to all media servers in an Shared Storage Option (SSO) configuration if there are more than 16 tape devices configured.

When the number of tape devices configured in an SSO configuration approaches 16, the default maximum size of Solaris IPC message queues may not be large enough. In these cases, communication between the `rdevmi` process on scan hosts and `oprd` processes on media servers can be interrupted when the number of messages sent exceeds the maximum size of the queue.

VERITAS recommends adding the following statement to the `/etc/system` file. This statement sets the maximum number of bytes in an IPC message queue to 65536. A reboot is necessary for the statement to take effect.

```
set msgsys:msginfo_msgmnb=65536
```

Be aware that increasing the maximum size of the IPC message queue may increase the amount of memory allocated to other IPC message queues on the same system. It is recommended that the impact of this change should be fully assessed before it is implemented.

Understanding the SCSI Passthru Drivers

NetBackup Media Manager provides its own driver for communicating with SCSI-controlled robotic peripherals. This driver is called the SCSA (Generic SCSI passthru driver), also referred to as the sg driver.

Note Since NetBackup uses its own passthru driver, the Solaris 8.0 sgen scsi passthru driver is not supported.

The sg driver is also used

- By the `avrd` process to scan drives.
- By NetBackup and Storage Migrator for
  - Locate-block positioning.
  - SAN error recovery.
  - Quantum SDLT performance optimization.
  - SCSI reserve / release.
- To set the optical drive type (as explained in “Setting the HP Optical Drive Type in Nonvolatile Memory” on page 26).
- By the NetBackup device configuration GUIs to collect robot and drive information.

Use the following procedure to manipulate the sg driver. Perform these steps as the root user.

1. Determine if an sg driver is loaded by using the following command:

   ```
   /usr/sbin/modinfo | grep sg

   141 fc580000  2d8c 116   1  sg (SCSA Generic Revision: 3.4d)
   153 fc7fa000  1684  49   1  msgsys (System V message facility)
   ```

2. Remove the existing driver:

   ```
   /usr/sbin/rem_drv sg
   ```

3. Install or reconfigure the sg driver.

   a. If reconfiguration is desired, run the following command first:

      ```
      /usr/bin/rm -f /kernel/drv/sg.conf
      ```

   b. To install the driver run the following command:

      ```
      /usr/openv/volmgr/bin/driver/sg.install
      ```

      Once the driver has been installed, it is not necessary to reboot the system or run the `sg.install` command during or after each system boot.

Configuring SG and ST Drivers

This procedure contains instructions for configuring the sg driver for SCSI targets 0 thru 6 and 8 thru 15 for fast or wide adapter cards.

In this procedure, you execute `sg.build` to add these targets to the `st.conf`, `sg.conf` and `sg.links` files. Adjust the `-mt` and `-ml` parameters to create the range of targets and LUNs required by your configuration.

1. Execute the `sg.build` script to add target IDs 0-6, 8-15, and LUNs 0-1 to the following files:

   ```
   /usr/openv/volmgr/bin/driver/st.conf
   /usr/openv/volmgr/bin/driver/sg.conf
   /usr/openv/volmgr/bin/driver/sg.links

   cd /usr/openv/volmgr/bin/driver
   /usr/openv/volmgr/bin/sg.build all -mt 15 -ml 1
   ```

   The `-mt 15` parameter specifies the maximum target ID that is in use on any SCSI bus (or bound to a fibre channel device).

   The `-ml 1` parameter specifies the maximum target LUN that is in use on any SCSI bus (or by a fibre channel device).

2. The file `/usr/openv/volmgr/bin/driver/st.conf` is used to replace the following seven entries in the `/kernel/drv/st.conf` file:

   ```
   name="st" class="scsi"
      target=0 lun=0;
   name="st" class="scsi"
      target=1 lun=0;
   name="st" class="scsi"
      target=2 lun=0;
   name="st" class="scsi"
      target=3 lun=0;
   name="st" class="scsi"
      target=4 lun=0;
   name="st" class="scsi"
      target=5 lun=0;
   name="st" class="scsi"
      target=6 lun=0;
   ```

   a. Make a copy of the `/kernel/drv/st.conf` file.

   b. Edit the `/kernel/drv/st.conf` file.

      Place a # in column one of each line of the seven default entries.

The temporary file `./st.conf` contains the entries that you need to insert into `/kernel/drv/st.conf`.

c. Reboot the system with the reconfigure option (`boot -r` or `reboot -- -r`).

d. Verify that the system created device nodes for all the tape devices using the following command: `ls -l /dev/rmt/*cbn`

3. The following is an example of the `/usr/openv/volmgr/bin/driver/sg.conf` file to add targets 0-6, 8-15, and LUNs 0-1:

```
name="sg" class="scsi" target=0 lun=0;
name="sg" class="scsi" target=0 lun=1;
name="sg" class="scsi" target=1 lun=0;
name="sg" class="scsi" target=1 lun=1;
name="sg" class="scsi" target=2 lun=0;
name="sg" class="scsi" target=2 lun=1;
name="sg" class="scsi" target=3 lun=0;
name="sg" class="scsi" target=3 lun=1;
name="sg" class="scsi" target=4 lun=0;
name="sg" class="scsi" target=4 lun=1;
name="sg" class="scsi" target=5 lun=0;
name="sg" class="scsi" target=5 lun=1;
name="sg" class="scsi" target=6 lun=0;
name="sg" class="scsi" target=6 lun=1;
name="sg" class="scsi" target=8 lun=0;
name="sg" class="scsi" target=8 lun=1;
name="sg" class="scsi" target=9 lun=0;
name="sg" class="scsi" target=9 lun=1;
name="sg" class="scsi" target=10 lun=0;
name="sg" class="scsi" target=10 lun=1;
name="sg" class="scsi" target=11 lun=0;
name="sg" class="scsi" target=11 lun=1;
name="sg" class="scsi" target=12 lun=0;
name="sg" class="scsi" target=12 lun=1;
name="sg" class="scsi" target=13 lun=0;
name="sg" class="scsi" target=13 lun=1;
name="sg" class="scsi" target=14 lun=0;
name="sg" class="scsi" target=14 lun=1;
name="sg" class="scsi" target=15 lun=0;
name="sg" class="scsi" target=15 lun=1;
```

4. The following is an example of the `/usr/openv/volmgr/bin/driver/sg.links` file to add targets 0-6, 8-15, and LUNs 0-1:

```
# begin SCSA Generic devlinks file - creates nodes in /dev/sg
type=ddi_pseudo;name=sg;addr=0,0;  sg/c\N0t0l0
type=ddi_pseudo;name=sg;addr=0,1;  sg/c\N0t0l0
type=ddi_pseudo;name=sg;addr=1,0;  sg/c\N0t1l0
type=ddi_pseudo;name=sg;addr=1,1;  sg/c\N0t1l1
type=ddi_pseudo;name=sg;addr=2,0;  sg/c\N0t2l0
type=ddi_pseudo;name=sg;addr=2,1;  sg/c\N0t2l1
type=ddi_pseudo;name=sg;addr=3,0;  sg/c\N0t3l0
type=ddi_pseudo;name=sg;addr=3,1;  sg/c\N0t3l1
type=ddi_pseudo;name=sg;addr=4,0;  sg/c\N0t4l0
type=ddi_pseudo;name=sg;addr=4,1;  sg/c\N0t4l1
type=ddi_pseudo;name=sg;addr=5,0;  sg/c\N0t5l0
type=ddi_pseudo;name=sg;addr=5,1;  sg/c\N0t5l1
type=ddi_pseudo;name=sg;addr=6,0;  sg/c\N0t6l0
type=ddi_pseudo;name=sg;addr=6,1;  sg/c\N0t6l1
type=ddi_pseudo;name=sg;addr=8,0;  sg/c\N0t8l0
type=ddi_pseudo;name=sg;addr=8,1;  sg/c\N0t8l1
type=ddi_pseudo;name=sg;addr=9,0;  sg/c\N0t9l0
type=ddi_pseudo;name=sg;addr=9,1;  sg/c\N0t9l1
type=ddi_pseudo;name=sg;addr=a,0;  sg/c\N0t10l0
type=ddi_pseudo;name=sg;addr=a,1;  sg/c\N0t10l1
type=ddi_pseudo;name=sg;addr=b,0;  sg/c\N0t11l0
type=ddi_pseudo;name=sg;addr=b,1;  sg/c\N0t11l1
type=ddi_pseudo;name=sg;addr=c,0;  sg/c\N0t12l0
type=ddi_pseudo;name=sg;addr=c,1;  sg/c\N0t12l1
type=ddi_pseudo;name=sg;addr=d,0;  sg/c\N0t13l0
type=ddi_pseudo;name=sg;addr=d,1;  sg/c\N0t13l1
type=ddi_pseudo;name=sg;addr=e,0;  sg/c\N0t14l0
type=ddi_pseudo;name=sg;addr=e,1;  sg/c\N0t14l1
type=ddi_pseudo;name=sg;addr=f,0;  sg/c\N0t15l0
type=ddi_pseudo;name=sg;addr=f,1;  sg/c\N0t15l1
# end SCSA devlinks
```

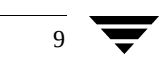
Caution The field separator between `addr=`*x*`,`*y*`;` and `sg/` is a tab.

The `addr=` field uses hexadecimal notation, while the `sg/` field uses decimal values.

5. Install the new sg driver configuration.

```
/usr/bin/rm -f /kernel/drv/sg.conf
/usr/openv/volmgr/bin/driver/sg.install
```

6. Verify that the sg driver found all the robots, tape drives, and optical disk drives (see the appropriate hardware configuration sections in this chapter for instructions).



Configuring the Sun StorEdge Network Foundation HBA Driver

The StorEdge Network Foundation HBA requires special configuration to bind device World Wide port names for use by the VERITAS sg driver.

The script `/usr/openv/volmgr/bin/sg.build` adds the proper entries to the `sg.links` and `sg.conf` files. Before running the script, make sure that all devices are powered on and connected to the HBA.

An example of the additional entries in `/usr/openv/volmgr/bin/driver/sg.conf` follows:

```
name="sg" parent="fp" target=0 lun=0 fc-port-wwn="22000090a50001c8";
name="sg" parent="fp" target=0 lun=1 fc-port-wwn="22000090a50001c8";
```

An example of the additional entries in `/usr/openv/volmgr/bin/driver/sg.links` follows:

```
type=ddi_pseudo;name=sg;addr=w22000090a50001c8,0;       sg/c\N0t\A1l0
type=ddi_pseudo;name=sg;addr=w22000090a50001c8,1;       sg/c\N0t\A1l1
```

Note Note: Each time a new device is added or an old device removed, re-create and re-install the new sg configuration (see “Configuring SG and ST Drivers” on page 7).

The script `/usr/openv/volmgr/bin/sgscan` checks for devices that are not configured, and produces output similar to the following example:

```
#WARNING: detected StorEdge Network Foundation connected devices not
        in sg configuration file:
#
#    Device World Wide Port Name 21000090a50001c8
#
#    See /usr/openv/volmgr/MediaMgr_DeviceConfig_Guide.txt chapter
#    "Configuring the Sun StorEdge Network Foundation HBA/Driver"
#    for information on how to use sg.build and sg.install
#    to configure these devices
```

Configuring Third-Party Fibre Channel HBA Drivers

Fibre channel devices should be bound to specific target IDs by modifying the HBA driver's configuration files. The binding process assures that the target ID will not change after a system reboot or a fibre channel reconfiguration. In some instances, VERITAS products are configured to use a specific target ID, which if changed will cause the products to fail until they are reconfigured.

The binding process is vendor and product unique. Please refer to the documentation available for your specific HBA.

The binding may be based on the fibre channel World Wide name of the port (WWPN) or the node (WWNN), or the destination ID (AL-PA or fabric assigned).

Once the selected binding is in place, the configuration proceeds in the same manner as for parallel SCSI installations (see "Configuring SG and ST Drivers" on page 7).

Note Each time a new device is added or an old device removed, the binding must be updated to the new configuration.

Configuring Robotic Controls

Robots are controlled through a SCSI or a network connection.

- SCSI control is covered in the following sections.
- Configuration of network controlled robotic libraries (for example, ACS robot types) is discussed in the appendices of the UNIX Media Manager system administrator's guide.

Configuring SCSI Robotic Controls

Read this topic if you plan to use a robotic storage device that is controlled through a SCSI robotic connection. Information on supported SCSI robots (vendor models) can be found on the VERITAS support web site.

When communicating with SCSI-controlled robotic peripherals, Media Manager utilizes the SCSA Generic (`sg`) driver. This driver is provided with NetBackup.

Note You must install the `sg` driver before continuing with the instructions in this topic (see "Configuring SG and ST Drivers" on page 7 for details).

To display the device files that are available to be used through the `sg` driver, use the `sgscan` command with the `all` parameter and note the lines that indicate changer devices (robotic libraries), as in the following example:

```
# /usr/openv/volmgr/bin/sgscan all
/dev/sg/c0t5l0: Tape (/dev/rmt/0): "HP      C1537A"
/dev/sg/c0t6l0: Cdrom: "TOSHIBA XM-5401TASUN4XCD"
/dev/sg/c1t2l0: Tape (/dev/rmt/7): "EXABYTE EXB-85058HE-0000"
/dev/sg/c1t4l0: Tape (/dev/rmt/9): "EXABYTE EXB-8900MH000202"
/dev/sg/c1t5l0: Changer: "EXABYTE EXB-210"
/dev/sg/c2t2l0: Tape (/dev/rmt/10): "Quantum DLT4000"
/dev/sg/c2t5l0: Tape (/dev/rmt/11): "QUANTUM DLT7000"
/dev/sg/c3t0l0: Disk (/dev/rdsk/c1t0d0): "FUJITSU M2952ESP SUN2.1G"
/dev/sg/c3t3l0: Disk (/dev/rdsk/c1t3d0): "FUJITSU M2952ESP SUN2.1G"
/dev/sg/c4t4l0: Tape (/dev/rmt/4): "Quantum DLT4000"
/dev/sg/c4t5l0: Tape (/dev/rmt/5): "Quantum DLT4000"
/dev/sg/c5t0l0: Disk (/dev/rdsk/c5t0d0): "SONY    SMO-F541"
/dev/sg/c5t1l0: Disk (/dev/rdsk/c5t1d0): "SONY    SMO-F541"
/dev/sg/c5t2l0: Disk (/dev/rdsk/c5t2d0): "SEAGATE ST11200N SUN1.05"
/dev/sg/c5t6l0: Disk (/dev/rdsk/c5t6d0): "SEAGATE ST11200N SUN1.05"
/dev/sg/c6t3l0: Changer: "SONY    DMS-B35"
/dev/sg/c6t5l0: Tape (/dev/rmt/6): "SONY    GY-2120"
/dev/sg/c7t0l0: Disk (/dev/rdsk/c7t0d0): "SEAGATE ST32550W SUN2.1G"
/dev/sg/c7t3l0: Disk (/dev/rdsk/c7t3d0): "MICROP  4221-09   1128RA"
/dev/sg/c7t4l0: Disk (/dev/rdsk/c7t4d0): "MICROP  4221-09MZ  Q4D"
/dev/sg/c8t2l0: Tape (/dev/rmt/14): "Quantum DLT4000"
/dev/sg/c8t3l0: Changer: "STK     9740"
/dev/sg/c8t4l0: Tape (/dev/rmt/13): "STK     SD-3"
/dev/sg/c8t6l0: Changer: "STK     9710"
/dev/sg/c9t0l0: Changer: "EXABYTE Exabyte 18D"
/dev/sg/c9t1l0: Tape (/dev/rmt/15): "Quantum DLT4000"
```

Note Specific device types can be filtered from the output using other forms of `sgscan`.

Usage: `sgscan [all|basic|changer|disk|tape] [conf] [-v]`

Examples of SCSI Robotic Control Device Files

Example 1

Using the previous `sgscan` output, if the SCSI robotic control for an Exabyte 210 is SCSI ID 5 of adapter 1, use the following path:

```
/dev/sg/c1t5l0
```

Example 2

Using the previous `sgscan` output, if the SCSI robotic control for a Sony library is SCSI ID 3 of adapter 6, use the following path:

```
/dev/sg/c6t3l0
```

Example 3

Using the previous `sgscan` output, if the SCSI robotic control for an STK 9710 is SCSI ID 6 of adapter 8 and you want to use TLD robotics, use the following path:

```
/dev/sg/c8t6l0
```

Example 4

If the SCSI robotic control for a DLT2700, DLT4700, or HP C1560B was SCSI ID 5 of adapter 0, use the following path:

```
/dev/sg/c0t5l1
```

Note that logical unit number 1 is used for those devices. The sg driver configuration can be modified so `sgscan` lists LUN 1 devices. In the sample `sgscan` output the configuration was not modified.

Example 5

Using the previous `sgscan` output, even if the SCSI robotic control for an STK 9740 is SCSI ID 3 of adapter 8, you would not enter any path to configure ACS robotic control.

Instead, assuming ACS control over the network, enter the appropriate ACSLS Host name. If you want to use TLD robotics to control the 9740, specify the following path:

```
/dev/sg/c8t3l0
```

Example 6 (IBM 3570 B-series Robotic Libraries)

If there is one drive in the robotic library, the robotic control is LUN 1 of the drive's SCSI ID. If there are two drives, the robotic control is LUN 1 of the Drive 1 SCSI ID. The SCSI ID's are viewed and configured by using the front panel on the robot.

The robotic control for the IBM 3570 B01/B02 is TLD, so if there are two drives, they may be connected to different host systems. If this is the case, the host system which is connected to drive 1 must also have the robotic control. Also, the library should be in RANDOM mode and BASE configuration. See the operator's guide supplied with the unit for information on setting library mode and configuration.

Assume a configuration as follows:

```
# /usr/openv/volmgr/bin/sgscan

/dev/sg/c0t0l0: Disk (/dev/rdsk/c0t0d0): "IBM   DCAS32160SUN2.1G"
/dev/sg/c0t6l0: Cdrom: "TOSHIBA XM5701TASUN12XCD"
/dev/sg/c1t5l0: Tape (/dev/rmt/1): "IBM     03570B02"
/dev/sg/c1t6l0: Tape (/dev/rmt/2): "IBM     03570B02"
```

If drive 1 is SCSI ID 5, the robotic control for the robotic library is `/dev/sg/c1t5l1`.

Example 7 (Fujitsu M8100 Stackers)

The robotic control for the Fujitsu M8100 stacker is TSH. The unit must be set up to run in SYSTEM Mode and 2LUN Mode. See the M8100 Cartridge Tape Drive product guide supplied with the unit for information on setting the library modes.

The robotic control is LUN 1 of the drive's SCSI ID. The SCSI ID's are viewed and configured by using the front panel on the stacker.

Assume a configuration as follows:

```
# /usr/openv/volmgr/bin/sgscan

/dev/sg/c1t0l0: Tape (/dev/rmt/0): "FUJITSU M8100AA2"
/dev/sg/c1t0l1: Changer: "FUJITSU M8100AA2"
```

If the drive is SCSI ID 0, the robotic control for the stacker is `/dev/sg/c1t0l1`.

Configuring Tape Drives

Using Berkeley-Style Close

The examples in this section use Berkeley-style close for tape drives. This is indicated by the letter `b` after the density specification. You must specify Berkeley-style close for tape devices that you configure under Media Manager.

The terms *Berkeley-style close* and *AT&T style close* refer to where a tape is left logically positioned after a close operation (in relation to a tape mark). One style leaves an application logically positioned before a tape mark and the other leaves it after. Applications must assume where the tape is left after a close in order to establish the correct orientation the next time they do a tape-position or read operation. Some operating systems allow tape devices to be configured with either type of close. NetBackup assumes it is using Berkeley-style close.

Fast-Tape Positioning (locate-block)

For AIT, DLT, Exabyte, DTF, and half-inch tape drives, Media Manager supports the SCSI `locate-block` command for positioning to a specific block on a tape. This approach improves tape-positioning greatly over the alternative method.

Enabling locate-block

NetBackup and Storage Migrator use the `locate-block` command by default if you did not uninstall the sg passthru driver as explained in “Understanding the SCSI Passthru Drivers” on page 5. The driver is automatically installed with Media Manager.

Disabling locate-block

To disable locate-block positioning, execute the following command:

```
touch /usr/openv/volmgr/database/NO_LOCATEBLOCK
```

With locate-block positioning disabled, NetBackup uses the `forward-space-file/record` method and Storage Migrator skips file marks.

No Rewind Device Files

When adding tape drives to a Media Manager configuration, you need only specify a no rewind on close device path. To display the tape device files that are configured on your system, use the `sgscan` command with the `tape` parameter.

```
# /usr/openv/volmgr/bin/sgscan tape
/dev/sg/c0t5l0: (/dev/rmt/0): "HP      C1537A"
/dev/sg/c1t2l0: (/dev/rmt/7): "EXABYTE EXB-85058HE-0000"
/dev/sg/c1t4l0: (/dev/rmt/9): "EXABYTE EXB-8900MH000202"
/dev/sg/c2t2l0: (/dev/rmt/10): "Quantum DLT4000"
/dev/sg/c2t5l0: (/dev/rmt/11): "QUANTUM DLT7000"
/dev/sg/c4t4l0: (/dev/rmt/4): "Quantum DLT4000"
/dev/sg/c4t5l0: (/dev/rmt/5): "Quantum DLT4000"
/dev/sg/c6t5l0: (/dev/rmt/6): "SONY    GY-2120"
/dev/sg/c8t2l0: (/dev/rmt/14): "Quantum DLT4000"
/dev/sg/c8t4l0: (/dev/rmt/13): "STK     SD-3"
/dev/sg/c9t1l0: (/dev/rmt/15): "Quantum DLT4000"
```


Note All device types can be displayed in the output using the `all` parameter with `sgscan`. This command can be helpful for associating tape devices with other SCSI devices that may be configured on the same adapter.

Usage: `sgscan [all|basic|changer|disk|tape] [conf] [-v]`

No rewind on close device files are in the `/dev/rmt` directory, and have the following format:

`/dev/rmt/`*LOGICAL_DRIVE*`cbn`

Where:

LOGICAL_DRIVE is the logical drive id, as shown by the `sgscan` command.

The `c` indicates compression.

The `b` indicates Berkeley-style close.

The `n` indicates no rewind on close.

Examples of No Rewind Device Files

Example 1

Using the `sgscan` output, if an Exabyte 8505C drive is SCSI ID 2 of adapter 1, the device path you use follows:

`/dev/rmt/7cbn`

Example 2

Using the `sgscan` output, if a DLT7000 drive is SCSI ID 5 of adapter 2, the device path you use follows:

`/dev/rmt/11cbn`

Configuring Nonstandard Tape Drives

This topic applies to the following drive types.

Note These are nonstandard drive types that require changes to the kernel before you can use them on some of the supported versions of Solaris.

- Exabyte (models 8500, 8505, 8505XL, 8500C, 8900, or Mammoth2)
- Fujitsu M2488 and M8100

- HP 4-mm DAT
- IBM 3570 and 3590
- Quantum DLT2000, DLT4000, DLT7000, or DLT8000
- Sony AIT, AIT-2, and DTF
- STK half-inch cartridge
- Tandberg QIC and QIC 150

Caution As shown by the `st.conf` examples in this section, you must configure non-QIC tape drives as variable-mode devices if they are to be used by Media Manager on Solaris platforms. Otherwise, NetBackup is able to write data, but not read it. During a read, you see a "not in tar format" error.
The terms *variable mode* or *fixed mode* refers to the behavior of reads and writes and the way the kernel packs physical tape records into logical tape records for an application. Variable-mode devices allow more flexibility in reading previously written tapes. Many tape devices can be accessed in either mode. NetBackup assumes variable mode for non-QIC drives.

Note on Case and Spaces in st.conf Entries

Upper and lower case are significant. For example, using Hp instead of HP would not work.

Spaces are significant within quoted strings in the `/kernel/drv/st.conf` file. The area that users most frequently have trouble with is the vendor field, which must always be eight characters in length.

For example, the vendor/product string for an HP C1533A drive is as follows (HP and 6 spaces is the vendor field):

```
"HP      C1533A"
```

If you were to omit some of the spaces (HP and 2 spaces is now the vendor field) in the vendor field as in the following example, the drive would not be recognized correctly.

```
"HP  C1533A"
```

The best way to ensure that your entries are accurate is to copy them from the `MediaMgr_DeviceConfig_Guide.txt` file. See "Considerations When Using This Guide" on page 1.

Additions to the st.conf File

An entry must be included in this file for the drive types you are using.

Note The entries in this section were tested and are known to work, but other settings may also work in your configuration.

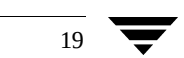
Caution Note the second portion of this list, where the third parameter (variable mode) must be 0. Not using 0 causes restores to fail and may result in data loss. (The entry for ARCHIVE_VIP is the only exception.)

```
tape-config-list =

"ARCHIVE VIPER 150", "Archive 150 Tape", "ARCHIVE_VIP",
"BNCHMARKDLT1", "Benchmark DLT1", "BM-DLT",
"Compaq DLT8000", "Quantum DLT8000 Tape Drive", "DLT8k-data",
"COMPAQ  SuperDLT1", "Compaq SuperDLT", "SDLT-data",
"DEC     DLT2000", "DEC DLT Tape Drive", "DEC-DLT",
"DEC     DLT2700", "DEC DLT Tape Stacker", "DEC-DLT",
"DEC     TZ89", "DEC DLT Tape Drive", "Q-DLT7000",
"EXABYTE EXB8500C", "Exabyte EXB-8500C 8mm Helical Scan", "EXB-8500C",
"EXABYTE EXB-8505", "Exabyte EXB-8505 8mm Helical Scan", "EXB-8505",
"EXABYTE EXB-8500", "Exabyte EXB-8500 8mm Helical Scan", "EXB-8500",
"EXABYTE EXB-8900", "Exabyte EXB-8900 Mammoth", "EXB-8900",
"EXABYTE Mammoth2", "Mammoth2 8MM Helical Scan Drive", "EXB-MAMMOTH2",
"FUJITSU M2488", "Fujitsu M2488", "FJ-D3",
"FUJITSU M8100", "Fujitsu M8100 1/2 Inch Cartridge", "FJ-M8100",
"HP      HP354", "HP 4mm DAT Drive", "HP-DAT",
"HP      C1533A", "HP DAT Autoloader", "HP-DAT",
"HP      C1557A", "HP Dat DDS3 Autoloader", "HP-DAT-DDS3",
"HP      C5683A", "HP DDS-4 4mm DAT", "HP_DAT_4",
"HP      Ultrium", "HP Ultrium", "Ultrium",
"IBM     03590", "IBM 3590 1/2 Inch Cartridge", "IBM-3590",
"IBM     03570", "IBM 3570 1/2 Inch Cartridge", "IBM-3590",
"IBM ULTRIUM-TD1", "IBM Ultrium", "CLASS_3580",
"IBM ULT3580-TD1", "IBM 3580 Ultrium", "CLASS_3580",
"SEAGATE ULTRIUM06242-XXX","Seagate LTO","SEAGATE_LTO",
"SONY    GY-2120", "Sony DTF Drive", "gy20-data",
"SONY    GY-8240", "DTF2", "gy2120-data",
"SONY    SDX-300C", "SONY 8mm AIT", "SONY_AIT",
"SONY    SDX-400C", "SONY 8mm AIT", "SONY_AIT",
"SONY    SDX-500C", "SONY 8mm AIT2", "SONY_AIT",
"SONY    SDX-700C",  "Sony AIT3 8mm", "SONY_AIT3",
"SONY    TSL-A300C", "SONY 8mm AIT", "SONY_AIT",
"SONY    TSL-A500C", "SONY 8mm AIT2", "SONY_AIT",
"STK     4781", "STK 1/2 Inch Cartridge (4480)", "STK-4781",
"STK     4791", "STK 1/2 Inch Cartridge (Silverton)", "STK-4791",
"STK     4890", "STK 1/2 Inch Cartridge (Twin Peaks)", "STK-4890",
"STK     9840", "STK 1/2 Inch Cartridge (9840)", "STK-9840",
"STK     SD-3", "STK 1/2 Inch Cartridge (Redwood)", "STK-SD-3",
```

```
"STK     T9940A", "STK 60 Gig Tape Drive", "CLASS_STK",
"SUN DLT4000", "SUN DLT Tape Drive", "DEC-DLT",
"SUN DLT7000", "SUN DLT7000 Tape Drive", "Q-DLT7000"
"TANDBERG SLR5 4/8GB", "Tandberg 8 Gig QIC", "TAND-8G-VAR",
"TANDBERGDLT4000", "Tandberg DLT4000", "DEC-DLT",
"TANDBERGDLT7000", "Tandberg DLT7000", "Q-DLT7000",
"TANDBERGDLT8000", "Tandberg DLT8000 Tape Drive", "DLT8k-data",
"TANDBERGSuperDLT1", "TANDBERGSuperDLT1", "SDLT-data",
"Quantum DLT2000", "Quantum DLT Tape Drive", "DEC-DLT",
"Quantum DLT4000", "Quantum DLT Tape Drive", "DEC-DLT",
"Quantum DLT4500", "Quantum DLT Tape Stacker", "DEC-DLT",
"Quantum DLT4700", "Quantum DLT Tape Stacker", "DEC-DLT",
"QUANTUM DLT7000", "Quantum DLT7000 Tape Drive", "Q-DLT7000",
"QUANTUM DLT8000", "Quantum DLT8000 Tape Drive", "DLT8k-data",
"Quantum DLT2700", "Quantum DLT Tape Stacker", "DEC-DLT",
"QUANTUM SuperDLT1", "QuantumSuperDLT", "SDLT-data";

ARCHIVE_VIP = 1,0x32,512,0x163a,4,0x0,0x0,0x0,0x0,3;
BM-DLT = 1,0x38,0,0x18639,4,0x40,0x40,0x40,0x40,3;
CLASS_3580 = 1,0x24,0,0x45863d,2,0x00,0x01,0;
CLASS_STK  = 1,0x36,0,0x1d639,1,0x00,0;
DEC-DLT = 1,0x36,0,0x9639,4,0x0,0x0,0x0,0x0,3;
DLT8k-data = 1,0x38,0,0x19639,4,0x1a,0x1b,0x41,0x41,3;
EXB-8500C = 1,0x35,0,0x9639,4,0x14,0x15,0x8C,0x00,3;
EXB-8505 = 1,0x35,0,0x9639,4,0x14,0x15,0x8C,0x00,3;
EXB-8500 = 1,0x35,0,0x9639,4,0x14,0x00,0x00,0x15,2;
EXB-8900 = 1,0x35,0,0x9639,4,0x27,0x27,0x27,0x00,3;
EXB-MAMMOTH2 = 1,0x35,0,0x19639,4,0,0x27,0x28,0x7f,2;
FJ-D3 = 1,0x21,0,0xCA19,4,0x09,0x09,0x09,0x09,0;
FJ-M8100 = 1,0x24,0,0x1d63d,4,0x0,0x0,0x0,0x0,3;
gy20-data = 1,0x36,0,0xd659,1,0x00,0;
gy2120-data = 1,0x36,0,0x19659,1,0x00,0;
HP-DAT = 1,0x34,0,0x9639,4,0x0,0x0,0x0,0x0,3;
HP-DAT-DDS3 = 1,0x34,0,0,0x9639,4,0x0,0x8c,0x8c,0x8c,3;
HP_DAT_4 = 1,0x34,0,0x9639,4,0x00,0x8c,0x8c,0x8c,1;
IBM-3590 = 1,0x24,0,0x1c63d,4,0x0,0x0,0x0,0x0,3;
Q-DLT7000 = 1,0x38,0,0x19639,4,0x82,0x83,0x84,0x85,3;
SDLT-data = 1,0x38,0,0x19639,4,0x90,0x91,0x90,0x91,3;
SEAGATE_LTO = 1,0x36,0,0x1d639,4,0x00,0x00,0x00,0x00,1;
SONY_AIT = 1,0x36,0,0x9639,4,0x0,0x0,0x0,0x0,0;
SONY_AIT3 = 1,0x36,0,0xd679,4,0x00,0x00,0x00,0x00,0;
STK-4781 = 1,0x24,0,0x1d43d,1,0x00,0;
STK-4791 = 1,0x24,0,0x1d67d,1,0x00,0;
STK-4890 = 1,0x24,0,0x1d67d,1,0x00,0;
STK-9840 = 1,0x36,0,0x1d639,1,0x00,0;
STK-SD-3 = 1,0x24,0,0x1d67d,1,0x00,0;
TAND-8G-VAR = 1,0x37,0,0x963b,4,0xa0,0xd0,0xd0,0xd0,3;
```

```
Ultrium = 1,0x36,0,0x19639,4,0x00,0x00,0x00,0x00,3;
```

Caution Reboot the system when you are done changing the kernel, using the reconfigure option (`boot -r` or `reboot -- -r`) to allow the kernel's SCSI tape (st) driver to recognize the drives as the correct type during system initialization.

Adding Logical Unit Number Entries

If the devices you are adding utilize the logical unit number (LUN) concept, (such as a half-inch cartridge drives that attach to an STK Automated Cartridge System) you must also add entries to the following files:

- `st.conf`
- `sg.conf`
- `sg.links`

See the "Configuring SG and ST Drivers" on page 7 for information on `sg.build`, a script that is used to create these files, and examples of the proper syntax to use.

Adding HP 4-mm Drives and HP DAT Autoloaders

Read this section if you plan to use Hewlett-Packard (HP) 4-mm DAT tape drives or HP C1560B DAT Autoloaders.

Note Other switch settings may work, but these settings were functional with an HP35480 drive and HP C1560B Autoloader during testing at VERITAS.

In the tables, 1 = On and 0 = Off. Use the following hardware (tape drive) switch settings on HP35480 4-mm (DAT) drives:

Switch	Setting
1	1
2	1
3	1
4	1
5	1

Switch	Setting
6	1
7	1
8	1

Use the following settings on HP C1533A drives in an HP C1560B DAT Autoloader:

Switch	Setting
1	1
2	1
3	1
4	1
5	1
6	1
7	0
8	0

Adding Sony AIT or AIT-2 Drives

Review this section if you plan to use Sony AIT or AIT-2 tape drives in your configuration.

No Rewind Device Files

When adding tape drives to a Media Manager configuration, you need only specify a no rewind on close device path. To display the no rewind device files that are configured on your system, use the `sgscan` command with the `tape` parameter.

```
# /usr/openv/volmgr/bin/sgscan tape

/dev/sg/c2t5l0: Tape (/dev/rmt/6): "SONY SDX-300C"
```

Using the `sgscan` output, if the drive is SCSI ID 5 of adapter 2, the device path you use follows:

```
/dev/rmt/6cbn
```

Dip Switch Settings

Sony drives have 8 dip switches located on the bottom of the drive. It is important to set these switches correctly, even if it means taking the drives out of robots and checking them.

Some robots (for example, Spectra Logic robots) provide a way to set the drive switches from the robot itself. For Spectra Logic robots, it doesn't matter what the drive switches are. The Treefrog (215) robot has a dial in the back to set the appropriate OS. The Bullfrog (10000) robot has a means of setting the OS through a touch screen.

Depending on the version of the AIT drive, drives are shipped from Sony with one of the following settings shown in the tables (in the tables, 1 = On and 0 = Off).

Note Robot vendors and hardware resellers may change the default drive switch settings.

Switch	Setting
1	0
2	0
3	0
4	0
5	0
6	0
7	1
8	1

Switch	Setting
1	0

Switch	Setting
2	0
3	0
4	0
5	1
6	0
7	1
8	0

Switches 1 thru 4 are for setting the OS type. Switches 5 thru 8 can be usually be left set at the default position.

Sony documentation (*UNIX Configuration Guide, V2.xx*) states that switches 1 thru 4 can be left set at the default position. However, the drive firmware must be at (or above) one of the following levels:

- 0404 (SDX-300C)
- 0700 (SDX-400C)
- 0107 (SDX-500C)

If the drive has older firmware, update the firmware or use the following settings for switches 1 thru 4:

Switch	Setting
1	0
2	1
3	0
4	1

You can use the following command to determine the current dip switch settings without removing the drives and checking them:

```
/usr/openv/volmgr/bin/scsi_command -d /dev/sg/c2t5l0 -ait
```


The output is as follows:

```
Physical AIT drive switch setting = 0x0 (Default configuration)
Logical AIT drive switch setting = 0xa (SUN - SunOS and Solaris)
```

Configuring Optical Disk Drives

Configuring HP Optical Disk Drives

To use standalone Hewlett-Packard optical-disk drives, the sg driver must be installed (see "Understanding the SCSI Passthru Drivers" on page 5). The system must also be configured to recognize the optical drives as disk drives at system boot time.

If you are adding HP 1.2 gigabyte or equivalent model magneto-optical disk drives, the system may not recognize these as disk drives and thus cannot use them. See "Setting the HP Optical Drive Type in Nonvolatile Memory" on page 26 for more information.

The Solaris 8 6/00 release introduced volume manager (`vold`), which attempts to manage all removable media devices. If `vold` manages an optical disk, NetBackup cannot access it.

Edit `/etc/vold.conf` and comment out the following line. Optical disks will then work as they did before this Solaris 8 change.

```
#use rmdisk drive /dev/rdsk/c*s2 dev_rmdisk.so rmdisk%d
```

Creating Device Files

When adding optical disk drives to a Media Manager configuration, you must specify the following device paths:

- Volume header disk device path (partition 0).
- Character device path (partition 6).

To display the disk device files that are configured on your system, use the `sgscan` command with the `disk` parameter:

```
# /usr/openv/volmgr/bin/sgscan disk

/dev/sg/c0t0l0: (/dev/rdsk/c0t0d0): "IBM     DCAS32160SUN2.1G"
/dev/sg/c0t1l0: (/dev/rdsk/c0t1d0): "HP      C1113F"
/dev/sg/c0t2l0: (/dev/rdsk/c0t2d0): "HP      C1113F"
/dev/sg/c0t5l0: (/dev/rdsk/c0t5d0): "HP      C1160F"
/dev/sg/c1t0l0: (/dev/rdsk/c1t0d0): "SONY    SMO-F541"
```

```
/dev/sg/c1t1l0: (/dev/rdsk/c1t1d0): "SONY    SMO-F541"
/dev/sg/c1t2l0: (/dev/rdsk/c1t2d0): "SEAGATE ST11200N SUN1.05"
```

Note All device types can be displayed using the `all` parameter when executing `sgscan`. This command can be helpful for associating disk devices with other SCSI devices that may be configured on the same adapter.

Usage: `sgscan [all|basic|changer|disk|tape] [conf] [-v]`

Optical disk device files are located in the `/dev` directory and have the following formats.

Volume header device:

`/dev/rdsk/c`*ADAPTER*`t`*TARGET*`d0s0`

Character device:

`/dev/rdsk/c`*ADAPTER*`t`*TARGET*`d0s6`

Where:

ADAPTER is the logical adapter number as shown in the `sgscan` output.

TARGET is the SCSI ID.

Examples of Optical Disk Device Files

Example 1

Using the previous sample `sgscan` output, if the desired optical disk drive connects to SCSI ID 5 of adapter card 0, you would use the following device paths:

Volume header device:

`/dev/rdsk/c0t5d0s0`

Character device:

`/dev/rdsk/c0t5d0s6`

Example 2

Using the previous sample `sgscan` output, if the desired optical disk drive connects to SCSI ID 0 of S bus 1 adapter card 1, you would use the following device paths:

Volume header device:

`/dev/rdsk/c1t0d0s0`

Character device:

```
/dev/rdsk/c1t0d0s6
```

Setting the HP Optical Drive Type in Nonvolatile Memory

To use Hewlett-Packard optical disk drives, the system must recognize the optical drives as disk drives at system boot time. If you are adding HP 1.2 gigabyte or equivalent model magneto-optical disk drives, the system may not recognize these as disk drives. The following steps explain how to correct this condition:

1. Install the sg loadable driver, if it is not already installed. See "Configuring SG and ST Drivers" on page 7 for information on installing this driver.

2. Use the `scsi_command` command to change the device type (stored in the drive's nonvolatile memory) from optical memory to disk. The format of the command follows.

   `/usr/openv/volmgr/bin/scsi_command -d /dev/sg/`*`sg_id`* `-disk`

   sg_id is the logical identifier assigned to the optical disk drive for use by the sg driver. See "Configuring SCSI Robotic Controls" on page 11 for information on determining the logical identifier.

Note The `/dev` path allows Media Manager to access the optical disk drive through the sg driver. This is an exception to the usual case where Media Manager uses the sg driver to access robotic controls. Therefore be sure to specify the SCSI ID for the optical disk drive, *not* the SCSI ID for the robotic control.

3. Reboot the system with the reconfigure option (`boot -r` or `reboot -- -r`) to allow the drive to be recognized as a disk drive by the kernel's SCSI disk (sd) driver during system initialization.

Command Summary

The following is a summary of commands that may be useful when configuring devices. See the procedures in this chapter for examples of their usage.

`/usr/sbin/modinfo | grep sg`

Displays whether or not the sg driver is installed.

`/usr/openv/volmgr/bin/driver/sg.install`

Installs or updates the sg driver.

`/usr/sbin/rem_drv sg`

Uninstalls the sg driver. This command is usually not necessary, since `sg.install` does this before performing a driver update.

`/usr/openv/volmgr/bin/sg.build all -mt` *max_target* `-ml` *max_lun*

Updates `st.conf`, `sg.conf`, and `sg.links`, and generates SCSI Target IDs with multiple LUNs.

`/usr/openv/volmgr/bin/sgscan all`

Scans all connected devices with a SCSI inquiry and provides correlation between physical and logical devices using all device files in `/dev/sg`.

Checks for devices connected to the Sun StorEdge Network Foundation HBA that are not configured for use by VERITAS products.

`/usr/openv/volmgr/bin/scsi_command -d /dev/sg/`*sg_id* `-disk`

Changes the device type (stored in the drive's nonvolatile memory) from optical memory to disk.

sg_id is the logical identifier assigned to the optical disk drive for use by the sg driver. See "Configuring SCSI Robotic Controls" on page 11 for information on determining the logical identifier.

`boot -r or reboot -- -r`

Reboot the system with the reconfigure option (`-r`) to allow a drive to be recognized as a disk drive during system initialization by the kernel's SCSI disk (sd) driver.

IBM RS6000 Running AIX 4.3.3.10/5.1 3

This chapter describes how to configure devices for use with Media Manager on an IBM RS6000 system. Configure drives and robots using one of the available Media Manager administrative interfaces.

The topics covered are as follows:

- Before You Start
- RS6000 AIX Adapter Number Conventions
- The SCSI Passthru Driver (ovpass)
- Configuring Robotic Controls
- Configuring Tape Drives
- Configuring Optical Disk Drives
- Command Summary

Before You Start

Observe the following points when performing the configurations described in this chapter:

- Attach all peripherals and reboot the system before configuring devices. Many of these steps may be accomplished using `smit` (the System Management Interface Tool). See the `smit(1)man` page for more information.
- To obtain error and debugging information about devices and robotic software daemons, the `syslogd` daemon must be configured to be active. See `syslogd(1)` for more information.

RS6000 AIX Adapter Number Conventions

The location code for an adapter consists of two pairs of digits with the format *AA-BB*; where *AA* identifies the location code of the drawer containing the adapter card and *BB* identifies both the I/O bus and slot containing the card.

A value of 00 for *AA* means that the adapter card is located in the CPU drawer or system unit, depending on the type of system. Any other value for *AA* indicates that the card is located in an I/O expansion drawer; in which case the value for *BB* identifies the I/O bus and slot number in the CPU drawer that contains the asynchronous expansion adapter. The first digit identifies the I/O bus with 0 corresponding to the standard I/O bus and 1 corresponding to the optional I/O bus. The second digit identifies the slot on the indicated I/O bus.

The first digit of *BB* identifies the I/O bus containing the adapter card. If the card is in the CPU drawer or system unit, this digit will be 0 for the standard I/O bus or 1 for the optional I/O bus. If the card is in an I/O expansion drawer, this digit is 0. The second digit identifies the slot number on the indicated I/O bus (or slot number in the I/O expansion drawer) that contains the card.

A location code of 00-00 is used to identify the Standard I/O Planar.

Examples

`00-05` identifies an adapter card that is in slot 5 of the standard I/O bus in either the CPU drawer or system unit, depending on the type of system.

`00-12` identifies an adapter card that is in slot 2 of the optional I/O bus in the CPU drawer.

`18-05` identifies an adapter card located in slot 5 of an I/O expansion drawer. The drawer is the one connected to the asynchronous expansion adapter located in slot 8 of the optional I/O bus in the CPU drawer.

The SCSI Passthru Driver (ovpass)

Read this topic if you plan to use SCSI-controlled robotic peripherals or Hewlett-Packard 1.2 gigabyte or equivalent model magneto-optical disk drives.

When communicating with SCSI-controlled robotic peripherals on an IBM RS6000 system, Media Manager utilizes a SCSI passthru driver called `ovpass`. This driver is also used to set the optical drive type, as documented in "Setting the HP Optical Drive Type in Nonvolatile Memory" on page 51. This driver is not required if the only peripheral is the IBM 3590 B11/E11 tape stacker.

Note You cannot use `smit` to configure `ovpass` device files.

Installing The SCSI Passthru Driver

To install the `ovpass` driver, enter the following command:

```
/usr/openv/volmgr/bin/driver/install_ovpass
```

To ensure the driver device files are accessible after each system boot, the following command should be placed in the system startup script:

```
/usr/openv/volmgr/bin/driver/mkdev_ovpass
```

Uninstalling The SCSI Passthru Driver

To uninstall the `ovpass` driver, enter the following command:

```
/usr/openv/volmgr/bin/driver/remove_ovpass
```

Upgrading The SCSI Passthru Driver

Note The following upgrade procedure is required only if you get an error while trying to configure a FCP SCSI device. See step 5 on page 33 for details.

Some versions of Media Manager before release 4.5 used a version of the passthru driver that did not support FCP SCSI controllers. To upgrade to the latest passthru driver, use the following procedure.

Caution Any device files (`/dev/ovpass`*n*) will be removed and must be re-created.

1. Uninstall the old `ovpass` driver.

   ```
   /usr/openv/volmgr/bin/driver/remove_ovpass
   ```


2. Install the new ovpass driver.

   ```
   /usr/openv/volmgr/bin/driver/install_ovpass
   ```

Configuring Robotic Controls

Robots are controlled through a SCSI or a network connection.

- SCSI control is covered in the following section.
- Configuration for network controlled robotic libraries is discussed in the appendices of the Media Manager system administrator's guide for UNIX.

Configuring SCSI Robotic Controls

Read this topic if you plan to use a robotic storage device that is controlled through a SCSI robotic connection. Information on supported SCSI robots (vendor models) can be found on the VERITAS support web site.

Perform the following steps to check for and create the necessary device files.

1. Install the SCSI passthru driver as explained in "The SCSI Passthru Driver (ovpass)" on page 31.

2. Display which SCSI controllers are physically available on your machine by using the following command:

   ```
   /usr/sbin/lsdev -C  | grep I/O
   ```

   In the following sample output, SCSI controller 1 (01) has been assigned the logical identifier scsi0:

   ```
   scsi0  Available 00-01    SCSI I/O Controller
   ```

   In the following sample output, FCP SCSI controller 3A-08 has been assigned the logical identifier fscsi0:

   ```
   fscsi0 Available 3A-08-01 FC SCSI I/O Controller Protocol Device
   ```

3. Display the SCSI device files that have already been created by using the following command:

   ```
   /usr/sbin/lsdev -C -s scsi
   ```

   The example output follows:

```
hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
```

This output shows that two disk drives and one tape drive are configured as follows:

- hdisk0 is a disk drive at controller 1 (01) and SCSI ID 0 (0,0)
- hdisk1 is a disk drive at controller 1 (01) and SCSI ID 1 (1,0)
- rmt0 is a tape drive at controller 1 (01) and SCSI ID 3 (3,0)

If the device files for the SCSI robotic control already exist, they appear in the `lsdev` output as ovpass0, ovpass1, etc. The output for this example does not show any ovpass files so you would have to create them as explained in the next step.

4. Display the FCP SCSI device files that have already been created by using the following command:

```
/usr/sbin/lsdev -C -s fcp
```

5. If device files for the desired robotic control SCSI ID do not exist, create the files using the following commands.

Note If you get an error (`mkdev` error code 0514-0520) while trying to configure a FCP SCSI device, review the topic "Upgrading The SCSI Passthru Driver" on page 31.

```
mkdev -c media_changer -s scsi -t ovpass -p controller -w id,lun
mkdev -c media_changer -s fcp -t ovpass -p controller -w scsi_id,lun
```

Where:

controller is the logical identifier of the drive's SCSI adaptor, such as scsi0, scsi1 or vscsi1.

id is the SCSI ID of the robotic connection.

scsi_id is the fibre channel identifier for the N_Port address (D_ID) of the robotic connection.

lun is the logical unit number of the robotic connection.

Several methods exist for determining the D_ID of the robot as follows:

a. Inspect the name server for the switch (if available). Each vendor uses a unique method to make this information available, consult the documentation for the switch vendor.

b. Inspect the bridge for mapping information (if available). Each vendor uses a unique method to make this information available, consult the documentation for the bridge vendor.

c. Inspect the odm database for any tape devices in the robot using the following command:

```
/usr/bin/odmget -q "name=rmtX" CuAt
```

Where *rmtX* is the name of the tape device (for example: rmt0 or rmt1).

The following output shows that rmt0 is at SCSI ID 0x1009ef and the robot might also be at this address:

```
CuAt:
name = "rmt0"
attribute = "scsi_id"
value = "0x1009ef"
type = "R"
generic = "DU"
rep = "s"
nls_index = 6
```

d. Use the smc (Library Medium Changer) entry for the robot to determine the correct SCSI ID and lun values. Inspect the `lsdev` output from step 4 and look for an entry like the following:

```
smc0 Available 14-08-01 IBM 3584 Library Medium Changer
(FCP)
```

If you see an entry for the robot, inspect the odm database using the following command:

```
/usr/bin/odmget -q "name=smc0" CuAt
```

```
CuAt:
name = "smc0"
attribute = "scsi_id"
value = "0x111ca"
type = "R"
generic = "DU"
rep = "s"
nls_index = 25
```

```
CuAt:
name = "smc0"
attribute = "lun_id"
value = "0x1000000000000"
type = "R"
generic = "DU"
rep = "s"
nls_index = 26
```

From the example output, the SCSI ID is 0x111ca and the lun is 1. The lun is derived from the lun ID, by right shifting it 48 bits.

6. Display the newly created logical identifier for the device by using one of the following commands:

```
/usr/sbin/lsdev -C -s scsi
/usr/sbin/lsdev -C -s fcp
```

In the following example output, `ovpass0` is a SCSI robotic control device file:

```
hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
ovpass0 Available 00-01-5,0    VERITAS Media Changer
```

The path name for these types of device files has the following form, where *ovpass_id* is the logical identifier assigned to the device.

`/dev/`*ovpass_id*

In this example, you would use the following device file path:

```
/dev/ovpass0
```

Examples of SCSI Robotic Control Device Files

Example 1

Assume this robot is not a TSD or an HP C1560B. The `ovpass` driver has been installed and the desired SCSI robotic controller is controller 1 at SCSI ID 5, but the device files do not exist.

1. Determine the logical identifier for the SCSI controller as follows:

```
/usr/sbin/lsdev -C -c adapter | grep SCSI
```

The following output shows that `scsi0` is the logical name for SCSI controller 1.

```
scsi0  Available 00-01    SCSI I/O Controller
```


2. Check if the device files exist for ovpass at SCSI ID 5.

   ```
   /usr/sbin/lsdev -C -s scsi
   ```

   The output shows that the device files exist for tape and disk, but not for the SCSI robotic control at controller 1 (scsi0) and SCSI ID 5 (5,0).

   ```
   hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
   rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
   ```

3. Create the device files by using the following command:

   ```
   mkdev -c media_changer -t ovpass -s scsi -p scsi0 -w 5,0
   ```

4. Display the device files by issuing the lsdev command:

   ```
   /usr/sbin/lsdev -C -s scsi

   hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
   hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
   rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
   ovpass0 Available 00-01-5,0    VERITAS Media Changer
   ```

   For this example use the following device file path to configure the SCSI robotic control connected to controller 1 and SCSI ID 5:

   ```
   /dev/ovpass0
   ```

Example 2

Assume the robot is a DLT2700 / DLT4700 (TSD) or an HP C1560B (TL4). The ovpass driver has been installed, but the device files for SCSI robotic control at controller 1 with SCSI ID 3 and logical unit number 1 do not exist.

1. Determine the logical identifier for the SCSI controller:

   ```
   /usr/sbin/lsdev -C -c adapter | grep -i SCSI
   ```

   The following output shows that scsi0 is the logical name for SCSI controller 1:

   ```
   scsi0  Available 00-01    SCSI I/O Controller
   ```

2. Check if the device files exist for ovpass at SCSI ID 5.

   ```
   /usr/sbin/lsdev -C -s scsi
   ```

   The following output shows that the device files exist for tape and disk, but not for the SCSI robotic control at controller 1 (scsi0), SCSI ID 3, and logical unit number 1 (3,1):

```
hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
```

3. The device files can now be created using the following command:

```
mkdev -c media_changer -t ovpass -s scsi -p scsi0 -w 3,1
```

4. Display the device files by issuing the `lsdev` command.

```
/usr/sbin/lsdev -C -s scsi

hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
ovpass0 Available 00-01-3,1    VERITAS Media Changer
```

For this example, the device file to use for the TSD SCSI robotic control connected at controller 1 with SCSI ID 3 and logical unit number 1 would be:

```
/dev/ovpass0
```

Example 3

Assume the robot is an STK 9710 connected to a F/W Differential SCSI board and the passthru driver has been installed. Assume the drives are at SCSI ID's 4 and 5, and the robotics is at SCSI ID 6.

1. Determine the correct scsi controller.

```
lsdev -C | grep scsi

scsi0    Available 00-02       SCSI I/O Controller
ascsi0   Available 00-04       Wide SCSI I/O Controller Adapter
vscsi0   Available 00-04-0,0   SCSI I/O Controller Protocol Device
vscsi1   Available 00-04-0,1   SCSI I/O Controller Protocol Device

lsdev -C -c tape

rmt2  Available 00-04-01-4,0 Other SCSI Tape Drive
rmt3  Available 00-04-01-5,0 Other SCSI Tape Drive
```

2. The drives are on Adapter 00-04-01. Therefore, `vscsi1` is the correct adapter for making the ovpass device file as follows:

```
mkdev -c media_changer -t ovpass -s scsi -p vscsi1 -w 6,0
```


Note Never use the scsi adapter name.

Example 4 (IBM 3570 B-series Robotic Libraries)

If there is one drive in the robotic library, the robotic control is LUN 1 of the drive's SCSI ID. If there are two drives, the robotic control is LUN 1 of the Drive 1 SCSI ID. The SCSI IDs can be set or viewed using the front panel on the robot. The robotic control for the IBM 3570 B01/B02 is TLD, so if there are two drives they may be connected to different host systems.

If this is the case, the host system which is connected to Drive 1 must also have the robotic control. Also, the library should be in RANDOM mode and BASE configuration. See the operator's guide supplied with the unit for information on setting library mode and configuration.

Assume a configuration as follows:

```
lsdev -C -c tape

rmt0 Available 00-02-01-5,0 Other SCSI Tape Drive
rmt0 Available 00-02-01-6,0 Other SCSI Tape Drive
```

If drive 1 is SCSI ID 5, the robotic control for the robotic library will be LUN 1 of this SCSI ID. Assuming vscsi1 is the correct adapter, make the passthru device (ovpass) as follows:

```
mkdev -c media_changer -t ovpass -s scsi -p vscsi1 -w 5,1
```

Configuring IBM 3590 Stacker Robotic Controls

If you plan to use a Tape Stacker Half-inch (TSH) robotic storage device, see the VERITAS support web site for the vendor model associated with the TSH robot type.

Perform the following steps to check for and create the necessary device files:

1. Display the SCSI tape devices configured in the system using the following command:

```
/usr/sbin/lsdev -C -c tape

rmt0  Defined   00-02-00-4,0 Other SCSI Tape Drive
rmt1  Available 00-08-00-6,0 2.3 GB 8mm Tape Drive
.
.
rmt12 Available 00-04-01-6,0 IBM 3590 Tape Drive and Medium Changer
```

2. The SCSI robotic path for the IBM 3590 is the same as the no rewind on close tape path. When configuring the TSH SCSI robotic path, the robotic control path for the 3590 would be `/dev/rmt12.1`. The tape drive path would also be `/dev/rmt12.1`.

Configuring Tape Drives

Configuring Non-QIC Tape Drives

The terms *variable length block* or *fixed length block* refers to the behavior of reads and writes and the way the kernel packs physical tape records into logical tape records for an application. Variable-mode devices allow more flexibility in reading previously written tapes. Many tape devices can be accessed in either mode. NetBackup assumes variable length for non-QIC drives.

For more information, see `chdev(1)`, `smit(1)` and the system management guide. The `smit` application is the most convenient way to manually change from fixed to variable-length-block devices.

Caution Non-QIC tape drives must be configured as variable-length-block devices if they will be used by Media Manager. Otherwise, NetBackup is able to write data but may not be able to read it correctly. During a read, you may see a "not in tar format" error.

Automatic Configuration

When a non-QIC tape drive is configured using NetBackup, NetBackup will automatically issue the `chdev` command to configure the drive as a variable length block device. It is normally not necessary to manually run the `chdev` command, but the details of the command issued by NetBackup are provided (see "Using the chdev Command" on page 40).

Using the chdev Command

Ensure that the device being used is configured for variable mode by using the `chdev` command as follows:

`/usr/sbin/chdev -l` *Dev* `-a block_size=0`

Where *Dev* is the logical identifier for the drive (for example: `rmt0` or `rmt1`).

Using Extended-File Marks for Drives

Tape drives must be configured to use extended file marks, if those tape drives are capable of supporting them (for example, 8-mm drives). See `chdev(1)` and `smit(1)` for additional information. Otherwise, NetBackup may not be able to use those drives.

Automatic Configuration

When a tape drive is configured using NetBackup, NetBackup will automatically issue the `chdev` command to configure the drive to use extended file marks. It is normally not necessary to manually run the `chdev` command, but the details of the command issued by NetBackup are provided (see "Using the chdev Command" on page 41).

Using the chdev Command

Ensure that the device being used is configured for extended file marks as required by Media Manager by using the `chdev` command as follows:

`/usr/sbin/chdev -l` *`Dev`* `-a extfm=yes`

Where *Dev* is the logical identifier for the drive (for example: `rmt0` or `rmt1`)

Fast-Tape Positioning (locate-block)

For DLT, Exabyte, and half-inch cartridge tape drives, Media Manager supports the SCSI `locate-block` command for positioning tape to a specific block. This improves tape-positioning greatly over what can be obtained with the alternative.

Media Manager uses the `locate-block` command by default unless you disable it by executing the following:

```
touch /usr/openv/volmgr/database/NO_LOCATEBLOCK
```

With locate-block positioning disabled, NetBackup uses the `forward-space-file/record` method.

Creating No Rewind Device Files

When adding tape drives to a Media Manager configuration, you need only specify a no rewind on close device path. These SCSI device files are in the `/dev` directory and have the following format:

`/dev/rmt`*`ID`*`.1`

Where *ID* is the logical identifier assigned to the device by the system.

Perform the following steps to check for and create the necessary device files:

1. Display which SCSI controllers are physically available by using the `lsdev` command as follows:

   ```
   /usr/sbin/lsdev -C  | grep I/O
   ```


This sample output shows that SCSI controller 1 (`00-01`) has been assigned the logical identifier `scsi0`.

```
scsi0  Available 00-01    SCSI I/O Controller
```

2. Display the SCSI device files that have already been created by using one of the following commands:

```
/usr/sbin/lsdev -C -s scsi
/usr/sbin/lsdev -C -s fcp

hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
```

This example output shows that two disk drives and one tape drive exist as follows:

- `hdisk0` is a disk drive at controller 1 (00-01) and SCSI ID 0 (`0,0`)
- `hdisk1` is a disk drive at controller 1 (00-01) and SCSI ID 1 (`1,0`)
- `rmt0` is a tape drive at controller 1 (00-01) and SCSI ID 3 (`3,0`)

If the device files for the SCSI tape drives exist, they appear in the output as `rmt0`, `rmt1`, and so on. The previous example output shows `rmt0`.

For `rmt0` and `rmt1`, you would use the following no rewind on close device files:

```
/dev/rmt0.1
/dev/rmt1.1
```

3. If the device files for the SCSI ID of the desired tape drive do not exist, create them using the following command:

`/usr/sbin/mkdev -c tape -s scsi -t ost -p` *controller* `-w` *id*`,`*lun*

Where:

controller is the logical identifier of the SCSI adapter for the device, such as `scsi0` or `scsi1`.

id is the SCSI ID of the drive connection.

lun is the logical unit number of the drive connection.

An example for an 8-mm drive connected to controller 0 and SCSI ID 5 follows:

```
mkdev -c tape -s scsi -t ost -p scsi0 -w 5,0
```

You can display the newly created logical identifier for the device by using the `lsdev` command.

```
/usr/sbin/lsdev -C -s scsi

hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
rmt1    Available 00-01-00-5,0 Other SCSI Tape Drive
ovpass0 Available 00-01-6,0    VERITAS Media Changer
```

The `rmt1` device file has been created.

4. If the device files do not exist on a FCP controller, use the following command to create them, where *device* is the controller number from step 1 on page 41. For example, fscsi0.

   ```
   /usr/sbin/cfgmgr -l device
   ```

5. Ensure that the device being used is configured for variable-mode and extended file marks as required by Media Manager by using the `chdev` command as follows, where *Dev* is the logical identifier for the drive (for example: `rmt0` or `rmt1`).

   ```
   /usr/sbin/chdev -l Dev -a block_size=0
   /usr/sbin/chdev -l Dev -a extfm=yes
   ```

No Rewind Device File Example

Assume the device files for the desired SCSI 8-mm tape drive (controller 1, SCSI ID 5) do not exist.

1. Determine the logical identifier for the SCSI controller as follows:

   ```
   /usr/sbin/lsdev -C -c adapter | grep SCSI
   ```

   The following output shows that `scsi0` is the logical name for SCSI controller 1:

   ```
   scsi0  Available 00-01    SCSI I/O Controller
   ```

2. Check if the device files exist for any device at SCSI ID 5.

   ```
   /usr/sbin/lsdev -C -s scsi
   ```

   The following output shows that some device files exist for tape and disk, but not for the 8-mm tape drive at controller 1 (scsi0) and SCSI ID 5 (5,0):

   ```
   hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
   hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
   rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
   ```


3. Create the desired device files by using the following command:

```
mkdev -c tape -t ost -s scsi -p scsi0 -w 5,0
```

4. Display the device files by issuing the following `lsdev` command:

```
/usr/sbin/lsdev -C -s scsi

hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
rmt1    Available 00-01-00-5,0 Other SCSI Tape Drive
```

5. To ensure that the tape device is configured for variable-mode and extended file marks, use the following commands:

```
chdev -l rmt1 -a block_size=0
chdev -l rmt1 -a extfm=yes
```

   Enter the following device file path to configure the 8-mm drive connected to controller 1 and SCSI ID 5:

```
/dev/rmt1.1
```

Using Multiple Tape Densities

After creating the necessary device files for your tape drives you may want to use non-default densities on drives that support them (for example, Exabyte 8500C tape drives).

There are two configurable densities available for all tape drives, although not all tape drives support multiple densities. The default density for both density setting 1 and density setting 2 is 0, which means maximum density.

To modify either of the density settings, you can use `smit(1)` or commands similar to the following:

```
chdev -l tapedev -a density_set_1=Density
chdev -l tapedev -a density_set_2=Density
```

Where:

tapedev is the logical identifier for the drive, such as `rmt0` or `rmt1`.

Density is the decimal number representing the desired density.

To use density setting 1, use the following no rewind on close device file:

```
/dev/rmt*.1
```

To use density setting 2, use the following no rewind on close device file:

```
/dev/rmt*.5
```

Adding HP 4-mm Drives and HP C1560B DAT Autoloaders

To support HP (Hewlett-Packard) 4-mm DAT tape drives and HP C1560B DAT Autoloaders use the tape drive switch settings as shown in the following table. Other combinations may work, but these are the settings that were functional during testing with an HP 35480 tape drive and HP C1560B DAT Autoloader.

In the table, 1 = On and 0 = Off.

Switch	Setting
1	1
2	1
3	1
4	1
5	1
6	1
7	0
8	0

Adding Sony AIT Drives

Read this section if you plan to use Sony AIT tape drives in your configuration.

No Rewind Device Files

When adding tape drives to a Media Manager configuration, you need only specify a no rewind on close device path. To display the no rewind device files that are configured on your system, use the `lsdev` command as follows:

```
/usr/sbin/lsdev -C -s scsi

rmt6 Available 00-03-01-6,0 Other SCSI Tape Drive
```


Using the `lsdev` output, if the drive is SCSI ID 6 of adapter 3, the device path you use follows:

```
/dev/rmt0.1
```

Dip Switch Settings

Sony AIT drives have 8 dip switches located on the bottom of the drive. It is important to set these switches correctly, even if it means taking the drives out of robots and checking them.

Some robots (for example, Spectra Logic) provide a way to set the drive switches from the robot itself. For Spectra Logic robots, it doesn't matter what the drive switches are. The Treefrog (215) robot has a dial in the back to set the appropriate OS. The Bullfrog (10000) robot has a means of setting the OS through the touchscreen.

Depending on the version of the AIT drive, drives are shipped from Sony with one of two switch settings, as shown in the following tables (in the tables, 1 = On and 0 = Off).

Note Robot vendors and hardware resellers may change the default drive switch settings.

Switch	Setting
1	0
2	0
3	0
4	0
5	0
6	0
7	1
8	1

Switch	Setting
1	0
2	0
3	0
4	0
5	1
6	0
7	1
8	0

Switches 1 thru 4 are for setting the OS type. Switches 5 thru 8 can be usually be left set at the default position.

Sony documentation (*UNIX Configuration Guide, V2.xx*) states that switches 1 thru 4 can be left set at the default position. However, the drive firmware must be at (or above) one of the following levels:

- 0404 (SDX-300C)
- 0700 (SDX-400C)
- 0107 (SDX-500C)

If the drive has older firmware, update the firmware or use the following switch settings for switches 1 thru 4:

Switch	Setting
1	1
2	0
3	0
4	0

You can use the following command to determine the correct dip switch settings without removing the drives and checking them:

```
/usr/openv/volmgr/bin/scsi_command -d /dev/rmt0.1 -ait
```

The output is as follows:

```
Physical AIT drive switch setting = 0x1 (IBM RS6000 - AIX - disconnect
enabled)
Logical AIT drive switch setting = 0xff (Not set, physical setting in
effect)
```

This example was an AIT drive in a ADIC Grau library. The drive was removed and set to the AIX switch settings.

Configuring Optical Disk Drives

When adding optical disk drives to a Media Manager configuration, you specify only a character device path. Optical disk character device files are located in the `/dev` directory and have the following format, where *ID* is the logical identifier assigned to the device by the system.

`/dev/rhdisk`*ID*

Note To use Hewlett-Packard optical disk drives, the system must recognize the optical drives as disk drives at system boot time. If you are adding Hewlett-Packard 1.2 gigabyte or equivalent model magneto-optical disk drives to an AIX system, the system may not recognize them as disk drives, and thus cannot use them. See "Setting the HP Optical Drive Type in Nonvolatile Memory" on page 51 for information on correcting this condition.

Creating Device Files

Perform the following steps to check for and create the necessary device files:

1. Display which SCSI controllers are physically available on your machine by using the following `lsdev` command:

   ```
   /usr/sbin/lsdev -C -c adapter | grep SCSI
   ```

   This sample output shows that SCSI controller 1 (`00-01`) has been assigned the logical identifier `scsi0`.

   ```
   scsi0  Available 00-01    SCSI I/O Controller
   ```

2. Display the SCSI device files that have already been created by using the following `lsdev` command:

   ```
   /usr/sbin/lsdev -C -s scsi
   ```

   The following example output shows that two disk drives and one tape drive exist:

   - `hdisk0` is a disk drive at controller 1 (00-01) and SCSI ID 0 (0,0)
   - `hdisk1` is a disk drive at controller 1 (00-01) and SCSI ID 1 (1,0)
   - `rmt0` is a tape drive at controller 1 (00-01) and SCSI ID 3 (3,0)

   If the device files for the SCSI optical disk drives exist, they show up in the output as `hdisk0`, `hdisk1`, and so on.

```
hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
```

For `hdisk0`, you would use the following device path:

```
/dev/rhdisk0
```

3. If the device files for the desired optical drive's SCSI ID do not exist, you can create them with the following command:

```
mkdev -c disk -s scsi -t osdisk -p controller -w id,lun
```

Where:

controller is the logical identifier of the device's SCSI adapter, such as `scsi0` or `scsi1`.

id is the SCSI ID of the drive connection.

lun is the logical unit number of the drive connection.

An example for an optical disk drive on controller 1 and SCSI ID 5 follows:

```
mkdev -c disk -t osdisk -s scsi -p scsi0 -w 5,0
```

4. You can display the newly created logical identifier for the device by using the following command:

```
/usr/sbin/lsdev -C -s scsi
```

```
hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
hdisk2  Available 00-01-00-5,0 Other SCSI Disk Drive
ovpass0 Available 00-01-6,0    VERITAS Media Changer
```

The device files for `hdisk2` have been created and you can now use them.

Examples of Optical Disk Device Files

Assume the device files for the desired optical disk drive (controller 1, SCSI ID 5) do not yet exist.

1. Determine the logical identifier for the SCSI controller as follows:

```
/usr/sbin/lsdev -C -c adapter | grep SCSI
```

The output shows that `scsi0` is the logical name for SCSI controller 1.

```
scsi0  Available 00-01    SCSI I/O Controller
```

2. Check to see if the device files exist for `ovpass` at SCSI ID 5.

   ```
   /usr/sbin/lsdev -C -s scsi
   ```

   The output shows that some device files exist for tape and disk, but not for the optical disk drive at controller 1 (`scsi0`) and SCSI ID 5 (5,0).

   ```
   hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
   hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
   rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
   ```

3. Create device files for the optical disk drive on controller 1 at SCSI ID 5 by using the following command:

   ```
   mkdev -c disk -t osdisk -s scsi -p scsi0 -w 5,0
   ```

4. Display the device files by issuing the `lsdev` command.

   ```
   /usr/sbin/lsdev -C -s scsi

   hdisk0  Available 00-01-00-0,0 400 MB SCSI Disk Drive
   hdisk1  Available 00-01-00-1,0 400 MB SCSI Disk Drive
   rmt0    Available 00-01-00-3,0 Other SCSI Tape Drive
   hdisk2  Available 00-01-00-5,0 Other SCSI Disk Drive
   ```

5. Enter the following character device file path to configure the optical disk drive connected to controller 1 and SCSI ID 5:

   ```
   /dev/rhdisk2
   ```

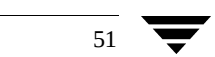
Setting the HP Optical Drive Type in Nonvolatile Memory

To use Hewlett-Packard optical disk drives, the system must recognize the optical drives as disk drives at system boot time. If you are adding HP 1.2 gigabyte or equivalent model magneto-optical disk drives to an AIX system, the system may not recognize them as disk drives and cannot use them.

To detect whether the system recognizes the optical drives, execute the following command after system boot:

```
/usr/sbin/lsdev -C -s scsi
```

If you see the appropriate controller and SCSI ID combination for the optical drive listed as Other SCSI Disk Drive, the system recognizes the drive as a disk drive. If not, use the procedure that follows.



```
hdisk0  Available 00-00-0S-0,0 2.2 GB SCSI Disk Drive
rmt0    Available 00-00-0S-3,0 Other SCSI Tape Drive
omd0    Defined   00-00-0S-6,0 Other SCSI Read/Write Optical Drive
ovpass0 Available 00-00-0S-2,0 VERITAS Media Changer
```

1. Install the ovpass driver if it is not already installed. See "The SCSI Passthru Driver (ovpass)" on page 31.

2. Create the ovpass device file for the optical drive so that the driver can be used to communicate with the optical drive.

   a. Display the SCSI device files that have already been created by using the following command: `/usr/sbin/lsdev -C -s scsi`

      The following example output shows that a disk drive, a tape drive, an optical drive, and SCSI robotic control are configured:

      - `hdisk0` is a disk drive at controller 1 (00) and SCSI ID 0 (0,0)
      - `rmt0` is a tape drive at controller 1 (00) and SCSI ID 3 (3,0)
      - `omd0` is an optical drive at controller 1 (00) and SCSI ID 6 (6,0)
      - `ovpass0` refers to the SCSI robotic control for controller 1 (00) and SCSI ID 2 (2,0)

```
hdisk0 Available 00-00-0S-0,0 2.2 GB SCSI Disk Drive
rmt0   Available 00-00-0S-3,0 Other SCSI Tape Drive
omd0   Defined   00-00-0S-6,0 Other SCSI Read/Write Optical Drive
ovpass0 Available 00-00-0S-2,0 VERITAS Media Changer
```

   b. Create the device files for the optical drive by using the following command:

      `mkdev -c media_changer -s scsi -t ovpass -p` *controller* `-w` *id*`,`*lun*

      Where:

      controller is the logical identifier of the drive's SCSI adapter, such as scsi0 or scsi1.

      id is the SCSI ID of the optical drive (not the robotic connection).

      lun is the logical unit number of the optical drive.

      For example:

      `mkdev -c media_changer -s scsi -t ovpass -p scsi 0 -w 6,0`

      Use the following command to obtain the logical identifier for the optical drive you just created:

```
/usr/sbin/lsdev -C -s scsi
```

c. Verify the temporary `ovpass` device file created in step b.

```
/usr/openv/volmgr/bin/scsi_command -d /dev/ovpass_id -inquiry
```

Where *ovpass_id* is the logical identifier assigned to the temporary device.

For example if the temporary ovpass device was `ovpass2`, enter the following:

```
/usr/openv/volmgr/bin/scsi_command -d /dev/ovpass2 -inquiry
```

The output shows the following:

```
removable device type c_8h_HP
```

3. Use the following command to change the device type (stored in the drive's nonvolatile memory) from optical memory to disk. The format of the command is as follows, where *ovpass_id* is the logical identifier assigned to the device.

```
/usr/openv/volmgr/bin/scsi_command -d /dev/ovpass_id -disk
```

For example:

```
/usr/openv/volmgr/bin/scsi_command -d /dev/ovpass1 -disk
```

4. Remove the ovpass device files and the optical drive that were created by using `rmdev` command as in the following:

```
rmdev -l ovpass_id -d
rmdev -l optical_drive_id -d
```

Where:

ovpass_id is the logical identifier assigned to the device.

optical_drive_id is the optical drive identifier assigned to the optical drive.

For example:

```
rmdev -l ovpass1 -d
rmdev -l omd0 -d
```

5. Reboot the system to allow the drive to be recognized as a disk drive by the kernel's SCSI disk driver during system initialization.

The optical drive should be displayed as: `hdisk`*logical_number*, where *logical_number* is the logical number assigned to the drive by the system.

For example:

```
/usr/sbin/lsdev -C -s scsi
```

The following example output shows a disk drive, tape drive, robotic control, and optical drive:

```
hdisk0  Available 00-00-0S-0,0 2.2 GB SCSI Disk Drive
rmt0    Available 00-00-0S-3,0 Other SCSI Tape Drive
ovpass0 Available 00-00-0S-2,0 VERITAS Media Changer
hdisk1  Available 00-00-0S-6,0 Other SCSI Disk Drive
```

Command Summary

The following is a summary of commands that may be useful when configuring devices. See the procedures in this chapter for examples of their usage.

`/usr/openv/volmgr/bin/driver/install_ovpass`

Installs the ovpass driver for the first time.

`/usr/openv/volmgr/bin/driver/remove_ovpass`

Uninstalls the ovpass driver.

`/usr/openv/volmgr/bin/driver/mkdev_ovpass`

Place this command in the system startup script to ensure that the ovpass driver device files are accessible after each system boot.

`/usr/sbin/lsdev -C | grep I/O`

Displays adapters that are physically available on your server.

`/usr/sbin/lsdev -C -s` *filetype*

Displays the device files that have been created, where *filetype* defines the type of file displayed. `scsi` displays SCSI files and `fcp` displays fibre channel files.

`mkdev -c media_changer -s scsi -t ovpass -p` *controller* `-w` *id*, *lun*

Creates device files for the robotic control SCSI ID.

Where *controller* is the logical identifier of the drive SCSI adaptor (such as `scsi0` or `scsi1`), *id* is the SCSI ID of the robotic connection, and *lun* is the logical unit number of the robotic connection.

`mkdev -c media_changer -s fcp -t ovpass -p` *controller* `-w` *scsi_id*, *lun*

Creates device files for the robotic control fibre channel SCSI ID.

Where *controller* is the logical identifier of the drive SCSI adaptor (such as `scsi0` or `scsi1`), *scsi_id* is the fibre channel SCSI ID of the robotic connection, and *lun* is the logical unit number of the robotic connection.

`mkdev -c disk -s scsi -t osdisk -p` *controller* `-w` *id*, *lun*

Creates device files for optical disk drives.

Where *controller* is the logical identifier of the drive SCSI adaptor (such as `scsi0` or `scsi1`), *id* is the SCSI ID of the robotic connection, and *lun* is the logical unit number of the robotic connection.

`mkdev -c tape -s scsi -t ost -p` *controller* `-w` *id*,*lun*

Creates device files for tapes.

Where *controller* is the logical identifier of the drive SCSI adaptor (such as `scsi0` or `scsi1`), *id* is the SCSI ID of the robotic connection, and *lun* is the logical unit number of the robotic connection.

`/usr/sbin/chdev -l` *dev* `-a block_size=0`

Configures the drive with logical identifier specified by *dev* (for example: `rmt0`) to variable mode.

`/usr/sbin/chdev -l` *dev* `-a extfm=yes`

Configures the drive with logical identifier specified by *dev* (for example: `rmt0`) for extended file marks.

`/usr/openv/volmgr/bin/scsi_command -d /dev/`*ovpass_id* `-disk`

Used for HP optical disk drives to change the device type (stored in the drive's nonvolatile memory) from optical memory to disk.

Where *ovpass_id* is the logical identifier assigned to the device.

`/etc/lsattr -l` *dev* `-E -H`

Displays device information, where *dev* is the name of the device (for example, `rmt1`).

`/usr/sbin/cfgmgr -l` *device*

Creates device files on a FCP controller, where *device* is the controller number (for example, fscsi0).

`/usr/bin/odmget -q "name=`*rmtX*`" CuAt`

Displays the device attributes for the device (*rmtX*). This command can be used to determine SCSI target and lun pairs when configuring fibre channel devices.

Where *rmtX* is the name of the tape device (for example: rmt0 or rmt1).

HP9000 Running HP-UX 11.0/11.11 4

This chapter shows how to configure devices for use with Media Manager on an HP9000 system. Configure drives and robots using one of the available Media Manager administrative interfaces.

The major topics included are as follows:

- Before You Start
- Configuring Robotic Controls
- Configuring Tape Drives
- Configuring Optical Disk Drives
- Command Summary

Before You Start

If You Are Using NetBackup BusinesServer

Portions of this chapter include configuration topics and examples for peripherals that are not supported in NetBackup BusinesServer. It is important to refer to the VERITAS support web site to determine which Media Manager robot types, robots, and drives are supported for NetBackup BusinesServer, before using this chapter.

Topics Not Applicable to NetBackup BusinesServer

- “Enabling SCSI Reserve/Release” on page 69.
- “Configuring Optical Disk Drives” on page 75.

Configuring Robotic Controls

Robots are controlled through a SCSI or a network connection.

- SCSI control is covered in the following sections.
- Configuration of network controlled robotic libraries (for example, ACS robots) is discussed in the appendices of the UNIX Media Manager System Administrator's Guide.

Configuring SCSI Robotic Controls

Read this topic if you plan to use a robotic storage device that is controlled through a SCSI robotic connection. Information on supported SCSI robots (vendor models) can be found on the VERITAS support web site.

Determining Which Passthru Driver to Configure

When communicating with SCSI-controlled robotic peripherals, Media Manager robotic software uses the `spt` or `sctl` SCSI passthru driver. The driver that is used depends on the type of SCSI interface on the system.

The two types of SCSI interfaces are as follows:

- Interfaces that use the scsi1/scsi3 bus-adapter driver require the `spt` passthru driver. The 28655A SCSI interface is in this category.

  See "Configuring Device Files for the spt Passthru Driver" on page 59.

- Interfaces that use the c700 or c720 bus-adapter driver require the `sctl` passthru driver. For example, the following interfaces are in this category:

  - The GSC built-in SCSI interface.
  - Add-on cards for HP9000-700.
  - Some add-on cards for HP9000-800 D, K, T, and V series systems.

  When attaching an autochanger (robotic library) device to a GSC interface and using the `sctl` driver, the `schgr` device driver must also be installed. Without this driver installed, the system will not bind the driver to the device. See the `autochanger(7)` man page.

  See "Configuring Device Files for the sctl Passthru Driver" on page 61.

Examples

To determine the type of interface on your system, use the `ioscan -f` command as shown in the following examples.

Example 1: 28655A SCSI Interface (spt driver)

```
ioscan -f

Class    I  H/W Path Driver S/W State   H/W Type     Description
=============================================================
bc       0           root    CLAIMED    BUS_NEXUS
bc       1 56        bc      CLAIMED    BUS_NEXUS    Bus Converter
ext_bus  0 56/52     scsi1   CLAIMED    INTERFACE    HP 28655A - SCSI
                                                        Interface
target   0 56/52.2   target  CLAIMED    DEVICE
tape     0 56/52.2.0 tape2   CLAIMED    DEVICE       HP  HPC1533A
```

In this case, the `ext_bus` entry (which designates the bus adapter) specifies a `scsi1` driver. You would configure the `spt` passthru driver for the SCSI robotic controls on this system (see “Configuring Device Files for the spt Passthru Driver” on page 59).

Example 2: Built-in SCSI interface (sctl driver)

```
ioscan -f

Class      I  H/W Path     Driver   S/W State  H/W Type  Description
=================================================================
ext_bus    2  10/12/5      c700     CLAIMED    INTERFACE Built-in SCSI
target    11  10/12/5.0    tgt      CLAIMED    DEVICE
tape       0  10/12/5.0.0  stape    CLAIMED    DEVICE    HP C1533A
target    12  10/12/5.2    tgt      CLAIMED    DEVICE
disk       6  10/12/5.2.0  sdisk    CLAIMED    DEVICE    TOSHIBA
                                                           CD-ROM
```

In this case, the `ext_bus` entry specifies a `c700` driver. You would configure the `sctl` passthru driver for the SCSI robotic controls on this system (see “Configuring Device Files for the sctl Passthru Driver” on page 61).

Configuring Device Files for the spt Passthru Driver

Use the following procedure to configure these types of device files. See “Determining Which Passthru Driver to Configure” on page 58 to determine if you require these files.

Note The HP-UX kernel has to be reconfigured to use the `spt` SCSI passthru driver. Refer to the HP-UX `scsi_pt (7)` man page.

The device files for the `spt` driver have the following format:

`/dev/spt/c`*CONTROLLER*`t`*TARGET*`l`*UNIT*

Where:

CONTROLLER is the Instance number of the controlling bus. The Instance value is displayed in the output of `ioscan -f` in the column labeled I of the controller's entry (`ext_bus` in the column labeled Class).

TARGET is the SCSI ID of the robotic control.

UNIT is the SCSI logical unit number (LUN) of the robot. This is usually 0.

You must create the device files for the spt driver manually, as they are not created automatically when the system boots. The following steps describe how to create these device files. These steps are also documented in the `scsi_pt(7)` man page.

1. Install and configure the driver as described in the man page.
2. Determine the character major number of the spt driver using `lsdev -d spt`.
3. Use the following commands to create the device file for the SCSI robotic control:

```
mkdir /dev/spt
mknod /dev/spt/Name c Major 0xIITL00
```

Where:

Name is the device name as defined in the format:
c*CONTROLLER*t*TARGET*l*UNIT*

Major is the character major number (from the `lsdev` command).

II is two hexadecimal digits identifying the controlling bus interface card by its Instance number.

T is one hexadecimal digit representing the SCSI ID of robotic control.

L is one hexadecimal digit representing the SCSI LUN of the robotic control.

Example of a Device File

If the robotic control for an HP Optical Disk Library (ODL) is on a secondary SCSI bus at SCSI ID 3, LUN 0, use the following steps to create the device file.

1. Use the `ioscan -f` command to get information on the SCSI bus and the robotic control.

```
Class    I  H/W Path  Driver  S/W State H/W Type       Description
=============================================================
bc       0            root    CLAIMED   BUS_NEXUS
bc       1  56        bc      CLAIMED   BUS_NEXUS      Bus Converter
ext_bus  1  56/16     scsi1   CLAIMED   INTERFACE      HP 28655A -
                                                        SCSI Interface
target   4  56/16.3   target  CLAIMED   DEVICE
spt      0  56/16.3.0 spt     CLAIMED   DEVICE         HP    C1700T
```

   The Instance number for the robot's SCSI bus is 1. It also confirms that the spt driver is attached to the optical robotic control at H/W Path `56/16.3.0`.

2. Use `lsdev` to get the character major number for the spt driver.

   ```
   lsdev -d spt
   ```

   This output from this command shows that the character major number for the spt driver is 137.

   ```
   Character      Block        Driver           Class
   137            -1           spt              spt
   ```

3. Create the `/dev/spt` directory, if it has not already been created.

   ```
   mkdir /dev/spt
   ```

4. Create the device file as follows:

   ```
   mknod /dev/spt/c1t3l0 c 137 0x013000
   ```

   This command creates the `/dev/spt/c1t3l0` device file. Specify this file as the robot control path when configuring your device under Media Manager.

Configuring Device Files for the sctl Passthru Driver

Use the following procedure to configure these types of device files. See "Determining Which Passthru Driver to Configure" on page 58 to determine if you require these files.

Note You do not have to reconfigure the HP-UX kernel to use the `sctl` passthru driver on HP9000-700 systems, since the generic SCSI driver is part of basic HP-UX.

If the devices do not exist, you can create device files by using the `mknod` command as follows. See the `scsi_ctl(7)` man page.

```
mkdir /dev/sctl
cd /dev/sctl
```

```
/etc/mknod cCONTROLLERtTARGETlLUN c 203 0xIITL00
```

Where:

CONTROLLER is the Instance number of the controlling bus. The Instance value is displayed in `ioscan -f` output under column I of the controller entry (ext_bus in the Class column).

TARGET is the SCSI ID of the robotic control.

LUN is the SCSI logical unit number.

II are two hexadecimal digits that identify the controlling bus interface card by its Instance number (same as controller).

T is one hexadecimal digit representing the SCSI ID.

L is one hexadecimal digit representing the SCSI LUN.

Using ioscan With sctl Robots

If the robotic control has its own SCSI ID, it has an entry similar to the following:

```
Class     I  H/W Path   Driver   S/W State H/W Type  Description
===================================================================
unknown  -1  2/0/1.1.0  unknown  UNCLAIMED UNKNOWN   LAGO SYSLS-340L
```

The `Class`, `I`, and `Driver` columns may also have invalid information. In these instances the robotics are correct, but `ioscan` returns invalid information.

Examples of Device Files

Example 1

In this example the robotic control for a ADIC Scalar 100 library is on a SCSI bus with a instance number of 7 (ext_bus entry, I column), SCSI ID 2 and LUN 0. The robotic control for a IBM ULT3583-TL library is on the same SCSI bus at SCSI ID 3 and LUN 0.

Use the following procedure to create the robotic device files:

1. Use the `ioscan -f` command to get information on the SCSI bus and the robotic control.

```
Class     I  H/W Path     Driver  S/W State H/W Type  Description
===================================================================
ext_bus   7  0/7/0/1      c720    CLAIMED   INTERFACE SCSI C896
                                                     Fast Wide LVD
```

```
target    10  0/7/0/1.0     tgt     CLAIMED   DEVICE
tape      65  0/7/0/1.0.0   stape   CLAIMED   DEVICE    QUANTUM
                                                       SuperDLT1
target    11  0/7/0/1.1     tgt     CLAIMED   DEVICE
tape      66  0/7/0/1.1.0   stape   CLAIMED   DEVICE    QUANTUM
                                                       SuperDLT1
target    12  0/7/0/1.2     tgt     CLAIMED   DEVICE
autoch    14  0/7/0/1.2.0   schgr   CLAIMED   DEVICE    ADIC Scalar 100
target    13  0/7/0/1.3     tgt     CLAIMED   DEVICE
autoch    19  0/7/0/1.3.0   schgr   CLAIMED   DEVICE    IBM ULT3583-TL
target    14  0/7/0/1.4     tgt     CLAIMED   DEVICE
tape      21  0/7/0/1.4.0   atdd    CLAIMED   DEVICE    IBM ULT3580-TD1
target    15  0/7/0/1.5     tgt     CLAIMED   DEVICE
tape      19  0/7/0/1.5.0   atdd    CLAIMED   DEVICE    IBM ULT3580-TD1
```

2. The commands to create the device files follow. See the "Command Summary" on page 77 for the `mknod` command format or see the `scsi_ctl(7)` man page.

   ```
   cd /dev/sctl
   /etc/mknod c7t2l0 c 203 0x072000
   /etc/mknod c7t3l0 c 203 0x073000
   ```

   This creates the following device files, which you specify to Media Manager during configuration of the ADIC robotic control and IBM robotic control respectively:

   ```
   /dev/sctl/c7t2l0
   /dev/sctl/c7t3l0
   ```

Note It is important to also create the passthru driver device files for tape drives. See "Configuring Tape Drives" on page 66.

Example 2

Assume the robotic control for an Exabyte 10i tape stacker (TS8) is on a built-in SCSI bus at SCSI ID 3, LUN 0. Also assume that an `ioscan -f` verifies that the SCSI ID is 3 and shows that the Instance number for the robot's SCSI bus is 1.

The commands to create the device file are

```
cd /dev/sctl
/etc/mknod c1t3l0 c 203 0x013000
```

This creates the following device file, which you specify to Media Manager:

```
/dev/sctl/c1t3l0
```


Example 3

1. Use the `ioscan -f` command to get information on the SCSI bus and the robotic control.

```
Class   I H/W Path          Driver  S/W State H/W Type  Description
===================================================================
ext_bus 3 0/0/0.8.0.0.0     fcpmux  CLAIMED   INTERFACE HP A3308
                                                FCP-SCSI MUX Interface
target  0 0/0/0.8.0.0.0.0   tgt     CLAIMED   DEVICE
tape    0 0/0/0.8.0.0.0.0.0 stape   CLAIMED   DEVICE    QUANTUM
                                                          DLT7000
target  1 0/0/0.8.0.0.0.1   tgt     CLAIMED   DEVICE
autoch  0 0/0/0.8.0.0.0.1.0 schgr   CLAIMED   DEVICE    STK9740
target  2 0/0/0.8.0.0.0.7   tgt     CLAIMED   DEVICE
ctl     3 0/0/0.8.0.0.0.7.0 sctl    CLAIMED   DEVICE    Initiator
```

   With fibre channel and SCSI muxes the hardware paths are a bit longer. If you use the bus H/W Path as a mask and apply it to the other hardware paths for devices on that bus, you are left with *SCSI ID.SCSI LUN* for the device.

   This example has a bus with H/W Path of 0/0/0.8.0.0.0, which has an instance number (I column) of 3. Applying the mask shows a DLT 7000 drive at SCSI ID 0, SCSI LUN 0 and a STK 9740 robot at SCSI ID 1, SCSI LUN 0 also on this bus.

   When configuring the robotic device file for the STK 9740 robot, you would use controller=3, target=1, and lun=0. To enable the passthru path for the Quantum DLT 7000 drive, you would use controller=3, target=0, and lun=0.

2. The commands to create the device file for the robotic control are as follows:

```
cd /dev/sctl
/etc/mknod c3t1l0 c 203 0x031000
```

   These commands create the following device file, which you specify to Media Manager during configuration of the robotic control:

```
/dev/sctl/c3t1l0
```

3. The commands to create the passthru device file for the tape drive follow. See the "Command Summary" on page 77 for the `mknod` command format or see the `scsi_ctl(7)` man page.

```
cd /dev/sctl
/etc/mknod c3t0l0 c 203 0x030000
```

   These commands create the following passthru device file.

```
/dev/sctl/c3t0l0
```

Although the passthru device file is used during NetBackup operation, it is not specified during configuration. During Media Manager tape drive configuration, the following no rewind on close device file path is used:

```
/dev/rmt/c3t0d0BESTnb
```


Configuring Tape Drives

Using Berkeley Style Close

The examples in this section show *Berkeley-style close* for tape drives as indicated by the letter `b` after the compression specification. It is mandatory to specify Berkeley-style close for tape devices that you configure under Media Manager.

The terms *Berkeley-style close* and *AT&T style close* refer to operations where a tape is left logically positioned after a close operation (in relation to a tape mark). One style leaves an application logically positioned before a tape mark and the other leaves it after. Applications must assume where the tape is left after a close in order to establish the correct orientation the next time they do a tape-position or read operation. Some operating systems allow tape devices to be configured with either type of close. NetBackup assumes it is using Berkeley-style close on an HP9000.

The Importance of Using the Passthru Driver for Tape Drives

NetBackup and Storage Migrator can function without a passthru path to tape devices, *but* the following capabilities are not available:

- Locate-block (fast-tape) positioning
- Data protection provided by SCSI reserve / release
- Automatic configuration for tape devices
- Performance optimization for Quantum SDLT drives.

Note Passthru paths are not supported on HP-PB adapters such as HP 28696A - Wide SCSI or HP 28655A - SE SCSI.

Automatic Configuration

Since using the passthru path is so important, NetBackup will automatically create the corresponding passthru path whenever a tape drive is configured with a device file of the format: `/dev/rmt/c`*CONTROLLER*`t`*TARGET*`d`*UNIT*`BESTnb`.

These paths are created in the `/dev/sctl` directory, which will also be created if it does not exist. Passthru paths will also be created automatically whenever the NetBackup device configuration wizard is run. In either case, NetBackup will not modify or delete any existing passthru paths. NetBackup assumes that any existing passthru paths were created correctly.

NetBackup does not detect the type of adapter cards installed in the system, so it will also create passthru paths for tape drives connected to adapter cards that do not support passthru. This is expected and will not cause any problems.

It is normally not necessary to manually create passthru paths for tape drives, but the steps for doing so are provided (see "Enabling Passthru Paths" on page 67).

Enabling Passthru Paths

To enable passthru paths, a device file in the directory `/dev/sctl` must exist for the tape drives. Create the device files as shown in the following example.

Note When using the `mknod` command for tape drives, *TARGET* is the SCSI ID of the tape drive *not* the SCSI ID of the robotic control.

Example:

In this example the robotic control for a ADIC Scalar 100 library is on a SCSI bus with a instance number of 7 (ext_bus entry, I column), SCSI ID 2 and LUN 0, and the robotic control for a IBM ULT3583-TL library is on the same SCSI bus at SCSI ID 3 and LUN 0.

The ADIC library contains 2 Quantum Super DLT drives, one with SCSI ID 0, SCSI LUN 0 and the other with SCSI ID 1, SCSI LUN 0.

The IBM library contains 2 IBM Ultrium LTO drives, one with SCSI ID 4, SCSI LUN 0 and the other with SCSI ID 5, SCSI LUN 0.

Use the following procedure to create the robotic control device files and the passthru driver tape drive device files:

1. Assume the configuration from `ioscan -f` is as follows:

```
Class     I  H/W Path     Driver  S/W State H/W Type   Description
=============================================================
ext_bus   7  0/7/0/       c720    CLAIMED    INTERFACE  SCSI  C896
                                                      Fast Wide LVD
target   10  0/7/0/1.0    tgt     CLAIMED    DEVICE
tape     65  0/7/0/1.0.0  stape   CLAIMED    DEVICE     QUANTUM
                                                      SuperDLT1
target   11  0/7/0/1.1    tgt     CLAIMED    DEVICE
tape     66  0/7/0/1.1.0  stape   CLAIMED    DEVICE     QUANTUM
                                                      SuperDLT1
target   12  0/7/0/1.2    tgt     CLAIMED    DEVICE
autoch   14  0/7/0/1.2.0  schgr   CLAIMED    DEVICE     ADIC Scalar 100
target   13  0/7/0/1.3    tgt     CLAIMED    DEVICE
autoch   19  0/7/0/1.3.0  schgr   CLAIMED    DEVICE     IBM ULT3583-TL
target   14  0/7/0/1.4    tgt     CLAIMED    DEVICE
```


```
tape    21  0/7/0/1.4.0  atdd   CLAIMED    DEVICE     IBM ULT3580-TD1
target  15  0/7/0/1.5    tgt    CLAIMED    DEVICE
tape    19  0/7/0/1.5.0  atdd   CLAIMED    DEVICE     IBM ULT3580-TD1
```

Note Use the IBM `atdd` driver when configuring IBM tape drives on HP-UX. Configure `atdd` and BEST device paths according to IBM driver documentation. Do not configure `atdd` for robotic control of IBM robots. Check the VERITAS support web site for the latest recommended `atdd` driver version from IBM.

2. The commands to create the robotic device files follow. See the "Command Summary" on page 77 for the `mknod` command format or see the `scsi_ctl(7)` man page.

```
cd /dev/sctl
/etc/mknod c7t2l0 c 203 0x072000
/etc/mknod c7t3l0 c 203 0x073000
```

This creates the following device files, which you specify to Media Manager during configuration of the ADIC robotic control and IBM robotic control respectively:

```
/dev/sctl/c7t2l0
/dev/sctl/c7t3l0
```

3. The commands to create the passthru device files for the tape drives are as follows:

```
cd /dev/sctl
/etc/mknod c7t0l0 c 203 0x070000
/etc/mknod c7t1l0 c 203 0x071000
/etc/mknod c7t4l0 c 203 0x074000
/etc/mknod c7t5l0 c 203 0x075000
```

These commands create the following passthru device files. Although the passthru device files for tape drives are used during NetBackup operation, they are not specified during configuration.

```
/dev/sctl/c7t0l0
/dev/sctl/c7t1l0
/dev/sctl/c7t4l0
/dev/sctl/c7t5l0
```

During Media Manager tape drive configuration, the following no rewind on close device files are used. See "No Rewind Device Files" on page 70 for instruction on creating no rewind device files.

```
/dev/rmt/c7t0d0BESTnb
/dev/rmt/c7t1d0BESTnb
/dev/rmt/c7t4d0BESTnb
/dev/rmt/c7t5d0BESTnb
```

Fast-Tape Positioning (locate-block)

Locate-block is supported for most drive types in HP9000 for Fast/Wide GSC SCSI adapters. See the VERITAS support web site for a list of drive types that are supported.

Enabling locate-block

NetBackup and Storage Migrator use the locate-block command by default if a passthru path is configured. See “Enabling Passthru Paths” on page 67.

Disabling locate-block

To disable locate-block positioning, execute the following command:

```
touch /usr/openv/volmgr/database/NO_LOCATEBLOCK
```

With locate-block positioning disabled, NetBackup uses the `forward-space-file/record` method and Storage Migrator skips file marks.

Enabling SCSI Reserve/Release

To enable the VERITAS implementation of SCSI reserve/release to protect data on a tape from corruption by other applications in a shared drive (SSO) configuration, you must

1. Set the kernel parameter `st_ats_enabled` to zero. Using the HP `SAM` utility is the easiest way to set this parameter.

   Leaving this parameter set to `ONE` causes conflicts between the operating system and NetBackup or Storage Migrator’s use of SCSI reserve/release commands.

2. Reboot your system.

Cautions with Using the HP-UX EMS Tape Device Monitor

The Tape Device Monitor (dm_stape) of the Event Monitoring System (EMS) should be configured to *not* run on hosts participating in a tape SAN configuration. Part of the EMS service periodically polls the tape devices to monitor their condition. When this occurs from one server while another server is using the tape device, it may interfere with backup operations causing those operations to time out and abort.

The problem can be avoided in either of the following ways:

- If you want to completely disable EMS you can run the `/etc/opt/resmon/lbin/monconfig` tool and select `(D)isable Monitoring`.
- If the POLL_INTERVAL value in the `/var/stm/config/tools/monitor/dm_stape.cfg` file is set to zero, EMS will still run, but it will not log any events or poll the devices (that is, it will not send any SCSI commands).

No Rewind Device Files

When adding tape drives to the Media Manager configuration, you need only specify a no rewind on close device file path. These device files are found in the `/dev/rmt` directory and have the following format:

`/dev/rmt/c`*CONTROLLER*`t`*TARGET*`d`*UNIT*`BESTnb`

Where:

CONTROLLER is the Instance number of the controlling bus. The Instance value is displayed in `ioscan -f` output under the column I of the controllers entry (ext_bus in the Class column).

TARGET is the SCSI ID of the tape drive.

UNIT is the SCSI logical unit number (LUN) of the drive. This is usually 0.

If the desired tape device file does not exist, you can create device files through `sam`, the system administration manager, or with the following `mksf(1M)` command:

`mksf -C tape -H` *H/W_Path* `-b BEST -u -n`

Where *H/W_Path* is the hardware path of the tape drive as specified by `ioscan`.

No Rewind Device File Example

Assume that the desired 4-mm DDS2 compression tape drive is at SCSI ID 2 and `ioscan -f` shows the following:

```
Class    I  H/W Path Driver  S/W State  H/W Type     Description
=============================================================================
bc       0            root   CLAIMED    BUS_NEXUS
bc       1  56        bc     CLAIMED    BUS_NEXUS    Bus Converter
ext_bus  0  56/52     scsi1  CLAIMED    INTERFACE    HP 28655A-SCSI
                                                          Interface
target   0  56/52.2   target CLAIMED    DEVICE
tape     0  56/52.2.0 tape2  CLAIMED    DEVICE       HP   HPC1533A
.
.
.
```

The Instance number for the controlling bus is 0 and the H/W path for the tape drive is `56/52.2.0`.

The command to create the device file for the drive follows:

```
mksf -C tape -H 56/52.2.0 -b BEST -u -n
```

This creates the following device file, which you specify to Media Manager:

```
/dev/rmt/c0t2d0BESTnb
```

Switch Settings for HP C1533A 4-mm DAT Drives

If you have standalone or robotic 4-mm drives, model HP C1533A, you may have to change the switch settings on the bottom of the drive. This drive comes in the HP C1560B (48AL) DAT Autoloader.

If the C1533A drive or HP C1560B autoloader was purchased from Hewlett Packard, the default switch settings should work. These default settings as documented by Hewlett Packard, are as follows. In the table, 1 = On and 0 = Off

Switch	Setting
1	1
2	1
3	0
4	1
5	1
6	1

Switch	Setting
7	1
8	1

However, if the drive or autoloader was purchased from another vendor and that vendor changed the switch settings, you will have to set the switches as shown.

You may also have to make this change to HP C1533A drives in non-Hewlett Packard 4-mm robots.

Switch Settings for Sony AIT Drives

Sony AIT drives have 8 dip switches located on the bottom of the drive. It is important to set these switches correctly, even if you have to take the drives out of robots to check them.

Some robots (for example, Spectra Logic) provide a way to set the drive switches from the robot itself. For Spectra Logic robots, it doesn't matter what the drive switches are. The Treefrog (215) robot has a dial in the back to set the appropriate OS. The Bullfrog (10000) robot has a means of setting the OS through a touch screen.

Depending on the version of the AIT drive, drives are shipped from Sony with one of the following two settings:

Note Robot vendors and hardware resellers may change the default drive switch settings.

In the following tables, 1 = On and 0 = Off.

Switch	Setting
1	0
2	0
3	0
4	0
5	0
6	0
7	1
8	1

Switch	Setting
1	0
2	0
3	0
4	0
5	1
6	0
7	1
8	0

Switches 1 thru 4 are for setting the OS type. Switches 5 thru 8 can be usually be left set at the default position.

Sony documentation (*UNIX Configuration Guide, V2.xx*) states that switches 1 thru 4 can be left set at the default position. However, the drive firmware must be at (or above) one of the following levels:

- 0404 (SDX-300C)
- 0700 (SDX-400C)
- 0107 (SDX-500C)

Configuring Optical Disk Drives

When adding optical disk drives to the Media Manager configuration, you need only specify a character device path. Optical disk character device files are found in the `/dev/rdsk` directory and have the following format:

`/dev/rdsk/c`*BI*`t`*TARGET*`d0`

Where:

BI is the bus Instance number of the controlling bus. The Instance value is displayed in `ioscan` output under the column I of the ext_bus entries.

TARGET is the SCSI ID of the drive. This ID is in the third position of the H/W Path as displayed by `ioscan`. For example, in 56/52.5.0 the SCSI ID is 5.

You can determine the bus Instance using `ioscan -C ext_bus -f`. The output is similar to the following example:

```
Class    I  H/W Path Driver S/W State H/W Type   Description
=============================================================
ext_bus 0  56/52    scsi1  CLAIMED   INTERFACE  HP 28655A- SCSI
                                                           Interface
ext_bus 1  56/53    lpr2   CLAIMED   INTERFACE  HP 28655A- Parallel
                                                           Interface
```

You can determine the configured drives using `ioscan -C disk -f`. The output is

```
Class    I  H/W Path  Driver S/W State H/W Type  Description
============================================================
disk     1  56/52.1.0 disc3  CLAIMED   DEVICE    HP C1716T
disk     2  56/52.2.0 disc3  CLAIMED   DEVICE    HP C1716T
disk     3  56/52.5.0 disc3  CLAIMED   DEVICE    HP C2490AM
disk     4  56/52.6.0 disc3  CLAIMED   DEVICE    HP C2490AM
```

Example of an Optical Disk Device File

Assume you are using the two optical disk drives at SCSI IDs 1 and 2 as shown in the previous `disk ioscan` example. These drives are on bus `56/52`, which as shown in the previous `ext_bus ioscan`, is bus Instance 0.

The character device file paths that you specify to Media Manager follow:

For target 1:

`/dev/rdsk/c0t1d0`

For target 2:

```
/dev/rdsk/c0t2d0
```

Using the Configure Storage Devices Wizard with Optical Disk Drives

The Media Manager device configuration wizard can discover and automatically configure optical disk drives that have a `/dev/sctl` SCSI passthru device file.

Note Optical disk drives are not supported in an SSO configuration.

To use this wizard, follow these steps:

1. Create the passthru device files.

   See "Configuring Device Files for the sctl Passthru Driver" on page 61 for instructions.

2. Start the wizard from the list of wizards displayed in the right pane of the **Media and Device Management** window of the NetBackup Administration Console.

   Select **Configure Storage Devices**.

   See your NetBackup Media Manager system administrator's guide for more information on using this wizard.

Command Summary

The following is a summary of commands that may be useful when configuring devices. See the procedures in this chapter for examples of usage.

`ioscan -C` *TYPE* `-f`

Shows information about the physical interfaces. Numeric information is displayed in decimal. *TYPE* is the type of interface as follows:

`spt` specifies SCSI robotic controls.

`tape` specifies tape drives.

`disk` specifies optical disks.

`ext_bus` specifies SCSI controllers.

`mknod /dev/spt/`*Name* `c` *Major* `0x`*IITL*`00`

Creates device files for SCSI robotic or tape drive controls.

Name is the device name as defined in the format: c*CONTROLLER*t*TARGET*l*UNIT*

Major is the character major number (from the `lsdev` command).

II are the two hexadecimal digits identifying the controlling bus interface card by its Instance number. The Instance value is displayed in the `ioscan` output under the I column of the proper ext_bus entry.

T is one hexadecimal digit for the SCSI ID of the robotic control.

L is one hexadecimal digit for the SCSI LUN of the robotic control.

`lsdev -d spt`

Displays information about the SCSI robotic control drivers.

`mksf -C tape -H` *H/W_Path* `-b BEST -u -n`

Creates device files for tape drives. Where *H/W_Path* is the hardware path of the tape drive, as specified by `ioscan`.

IRIX 6.5.10-6.5.14 5

This chapter provides information for configuring devices for use with Media Manager on an SGI platform running IRIX. You configure drives and robots using one of the available Media Manager administrative interfaces.

The topics included in this chapter are as follows:

- Before You Start
- Using SCIP Controllers
- Using the mediad Command
- Configuring Robotic Controls
- Configuring Tape Drives
- Configuring Optical Disk Drives
- Command Summary

Before You Start

Observe the following points when performing the configurations described in this chapter:

- The SCSI ID number *must be* unique for the SCSI bus it is on, and *must be* any value other than zero.
- Typical device path names used when configuring drives and robots are described. Instructions for changing and rebuilding the kernel are also included. Depending on the type and number of devices you are adding, you may have to enter information in kernel source files and then reconfigure the kernel.
- The SGI IRIX version of Media Manager has been tested using SCSI peripherals (tape drives, optical disk drives, and robotic control) attached to the built-in SCSI controllers, sometimes referred to as on-board SCSI or Integral SCSI controllers.

When referring to these SCSI controllers, this guide uses the term *integral SCSI controller*. Communication with tape drives attached to integral SCSI controllers is done through the `tps(7M)`, `tpsc(7M)`, and `ts(7M)` tape drivers. Communication with disk drives (including optical disk drives) attached to integral SCSI controllers is done through the `dks(7M)` disk driver.

Using SCIP Controllers

If your IRIX system has SCIP fast-wide-differential controllers, a change to the `/var/sysgen/master.d/scip` file may be required to avoid SCSI timeouts.

You should change the following:

```
uint            scip_mintimeout = 0
```

To the following:

```
uint            scip_mintimeout = 180
```

This value was tested with a Quantum DLT4700 and corrected driver errors. In general, it is better to try a peripheral first without modifying this file. If errors occur, then change the timeout and retry. You may have to contact SGI for further information.

After making this change, you must generate a new kernel and reboot the system as follows:

1. Run the following kernel auto-configuration script:

   ```
   /etc/autoconfig
   ```

2. Reboot the system to utilize the newly built kernel.

Using the mediad Command

Do *not* use the IRIX `mediad` command to monitor devices configured under Media Manager. If you do, Media Manager will not be able to access the devices and you will see a message similar to the following in the system log:

```
Apr 12 10:30:55 3D:boris mediad: Could not access
device /dev/rmt/tps0d4nr, Device busy
```

If you see this type of message and you are using `mediad`, then disable `mediad` as described in the `mediad(1M)` man page.

For example, assume you encounter this problem with a tape device whose device file is `/dev/rmt/tps0d4`. Instruct `mediad` to not monitor this tape device by editing the `/etc/config/mediad.config` file. `mediad` monitors this file so your change should be immediate.

In this example, you would add the following line to `mediad.config`:

```
ignore device /dev/rmt/tps0d4
```


Configuring Robotic Controls

Robots are controlled through a SCSI or a network connection.

- SCSI control is covered in the following section.
- Configuration for network controlled robotic libraries is explained in the appendices of the UNIX Media Manager system administrator's guide.

Configuring SCSI Robotic Controls

Read this topic if you plan to use a robotic device that is controlled through a SCSI robotic connection. Information on supported SCSI robots (vendor models) can be found on the VERITAS support web site.

When communicating with SCSI-controlled robotic peripherals on an SGI platform, Media Manager robotic software utilizes `ds(7M)`, the generic (user mode) SCSI driver. Since this driver is part of basic IRIX, you do not have to reconfigure the kernel and reboot the system to use this driver.

Examples of SCSI Robotic Control Device Files

Note Note that the second-to-last character in the following example paths is the letter l, rather than the number 1, and represents (l)ogical unit.

Example 1

If the robotics control is not for a DLT2700, DLT4700, HP C1560B, or other LUN 1 peripheral and is on SCSI bus (adapter) 0 at SCSI ID 5, the device file you specify is

```
/dev/scsi/sc0d5l0
```

Example 2

If the robotics control is not for a DLT2700, DLT4700, HP C1560B, or other LUN 1 peripheral and is on SCSI bus (adapter) 1 at SCSI ID 3, the device file you specify is

```
/dev/scsi/sc1d3l0
```

Example 3

If a DLT2700, DLT4700, HP C1560B, or other LUN 1 peripheral robotics control is on SCSI bus (adapter) 1 at SCSI ID 4 with logical unit number 1, the device file you specify is

```
/dev/scsi/sc1d4l1
```

Configuring Tape Drives

Read the following topics if you plan to use tape drives:

Fast-Tape Positioning (locate-block)

For most drive types, Media Manager supports the SCSI `locate-block` command for positioning a tape to a specific block. This improves tape-positioning greatly over the alternative method.

NetBackup and Storage Migrator use the `locate-block` command by default unless you disable the command by executing the following:

```
touch /usr/openv/volmgr/database/NO_LOCATEBLOCK
```

With `locate-block` positioning disabled, NetBackup uses the `forward-space-file/record` method and Storage Migrator skips file marks.

No Rewind Device Files

When adding tape drives to a Media Manager configuration, you need only specify a no rewind on close device path. In a typical configuration, most of the desired tape device files exist and you just have to locate them in the `/dev` directory.

No rewind on close device files that connect to the integral SCSI controllers have the following format:

`/dev/rmt/tps`*CONTROLLER*`d`*TARGET*`nrvc`

Where:

CONTROLLER is the SCSI bus (adapter) number.

TARGET is the SCSI ID.

The `v` specifies a variable mode device.

The `c` specifies tape compression.

Some device types (like Exabyte) also have suffixes on device files that designate their particular drive type. For example

`/dev/rmt/tps`*CONTROLLER*`d`*TARGET*`nrv.8500c (EXB8500C)`

Examples of No Rewind Device Files

Example 1

If the desired HP 4-mm (DAT) drive is on SCSI bus 1 at SCSI ID 4, specify the following device path for that drive:

```
/dev/rmt/tps1d4nrvc
```

Example 2

If the desired Exabyte 8500C or 8505 tape drive is on SCSI bus 0 at SCSI ID 3, specify the following device path for that drive:

```
/dev/rmt/tps0d3nrv.8500c
```

Example 3

If the desired DLT2000 or DLT4000 tape drive is on SCSI bus 0 at SCSI ID 5, specify the following device path for the drive:

```
/dev/rmt/tps0d5nrvc
```

Example 4

If the desired DLT7000 tape drive is on SCSI bus 0 at SCSI ID 5, specify the following device path:

```
/dev/rmt/tps0d5nrv.7000c
```

Example 5

If the desired Exabyte 8900 (Mammoth) is on SCSI bus 1 at SCSI ID 5, specify the following device file path for the drive:

```
/dev/rmt/tps1d5nrvc
```

Since this drive writes in only one format, you can ignore the other device files that are created for this drive.

Configuring Nonstandard Tape Drives

For the IRIX system to recognize these drives, specific entries are needed in the kernel. See “Changing the /var/sysgen/master.d/scsi File” on page 92 for instructions on adding these entries.

Adding HP 4-mm DAT Drives and HP C1560B DAT Autoloaders

Ensure that the hardware switch settings on HP35480A 4-mm (DAT) drives are as shown in the following tables (in the tables, 1 = On and 0 = Off).

Note Other combinations may work, but these are the settings that were functional during testing by VERITAS with an HP35480A drive and HP C1560B Autoloader.

Switch	Setting
1	1
2	1
3	1
4	1
5	1
6	1
7	0
8	0

Ensure that the hardware (tape drive) switch settings on the HP C1533A 4-mm (DAT) drives are as follows:

Switch	Setting
1	1
2	1
3	0
4	1
5	1
6	1

Switch	Setting
7	0
8	0

Adding DAT drives (except the HP C1560B DAT Autoloader)

```
/* HP DAT drives. Any product number that starts with HP354.*/
{ DATTAPE, TPDAT, 2, 5, "HP", "HP354", 0, 0, {0, 0, 0, 0},
MTCAN_BSF|MTCAN_BSR|MTCAN_APPEND|MTCAN_SETMK|MTCAN_PART
                                                    |MTCAN_PREV|
MTCAN_SYNC|MTCAN_SPEOD|MTCAN_CHKRDY|MTCAN_VAR|MTCAN_SETSZ|
MTCAN_SILI|MTCAN_SEEK|MTCAN_CHTYPEANY,
/* minimum delay on i/o is 4 minutes, because when a retry is
* performed, the drive retries a number of times, and then
* rewinds to BOT, repositions, and tries again. */
40, 4*60, 4*60, 5*60, 512, 128*512, 0, (u_char*)0, 3 * 3600,
(0), 0, 0, 0,
},
```

Adding the HP C1560B DAT Autoloader

```
/* HP DAT drives. Any product number that starts with HP1533. */
{ DATTAPE, TPDAT, 2, 5, "HP", "C1533", 0, 0, {0, 0, 0, 0},
MTCAN_BSF|MTCAN_BSR|MTCAN_APPEND|MTCAN_SETMK|MTCAN_PART
                                                    |MTCAN_PREV|
MTCAN_SYNC|MTCAN_SPEOD|MTCAN_CHKRDY|MTCAN_VAR|MTCAN_SETSZ|
MTCAN_SILI|MTCAN_SEEK|MTCAN_CHTYPEANY,
/* minimum delay on i/o is 4 minutes, because when a retry is
* performed, the drive retries a number of times, and then
* rewinds to BOT, repositions, and tries again. */
40, 4*60, 4*60, 5*60, 512, 128*512, 0, (u_char*)0, 3 * 3600,
(0), 0, 0, 0,
},
```

Adding Sony DTF Drives

For the IRIX system to recognize these drives, the following specific entries are needed in the kernel.

```
/* SONY GY-2120 drive */
{ SONYGY, TPGY2120, 4, 7, "SONY", "GY-2120", 0, 0, {0, 0, 0, 0},
MTCAN_BSF | MTCAN_BSR | MTCANT_RET | MTCAN_CHKRDY | MTCAN_PREV |
MTCAN_SEEK | MTCAN_APPEND | MTCAN_SILI | MTCAN_VAR | MTCAN_SETSZ |
MTCAN_CHTYPEANY | MTCAN_COMPRESS,
20, 100*60, 10*60, 9*60, 9*60, 16384, 256*1024,
tpsc_default_dens_count, tpsc_default_hwg_dens_names,
tpsc_default_alias_dens_names,
{0}, 0, 0, 0,
0, (u_char *)0
},

/* SONY GY-8240 drive */
{ SONYGY, TPGY2120, 4, 7, "SONY", "GY-8240", 0, 0, {0, 0, 0, 0},
MTCAN_BSF | MTCAN_BSR | MTCANT_RET | MTCAN_CHKRDY | MTCAN_PREV |
MTCAN_SEEK | MTCAN_APPEND | MTCAN_SILI | MTCAN_VAR | MTCAN_SETSZ |
MTCAN_CHTYPEANY | MTCAN_COMPRESS,
20, 100*60, 10*60, 9*60, 9*60, 16384, 256*1024,
tpsc_default_dens_count, tpsc_default_hwg_dens_names,
tpsc_default_alias_dens_names,
{0}, 0, 0, 0, 0, (u_char *)0
},
```

Adding Sony AIT-2 Drives

For the IRIX system to recognize these drives, the following specific entry is needed in the kernel.

```
/* SONY AIT-2 drive */
{ SONYAIT, TP8MM_AIT, 4, 8, "SONY", "SDX-500C" /*AIT2*/, 0, 0, {0},
MTCAN_BSF | MTCAN_BSR | MTCAN_APPEND | MTCAN_SETMK | MTCAN_PART |
MTCAN_PREV | MTCAN_SYNC | MTCAN_SPEOD | MTCAN_CHKRDY | MTCAN_VAR |
MTCAN_SETSZ |MTCANT_IMM | MTCAN_SILI | MTCAN_SEEK | MTCAN_CHTYPEANY
| MTCAN_COMPRESS,
40, 4*60, 4*60, 5*60, 3*3600, 512, 512*512,
tpsc_default_dens_count, tpsc_default_hwg_dens_names,
tpsc_default_alias_dens_names,
{0}, 0, 0, 0,  0, (u_char *)0
},
```


Checking Switch Settings

Sony AIT drives have 8 dip switches located on the bottom of the drive. It is important to set these switches correctly, even if it means taking the drives out of robots and checking them.

Some robots (for example, Spectra Logic) provide a way to set the drive switches from the robot itself. For Spectra Logic robots, it doesn't matter what the drive switches are. The Treefrog (215) robot has a dial in the back to set the appropriate OS. The Bullfrog (10000) robot has a means of setting the OS through a touch screen.

Depending on the version of the AIT drive, drives are shipped from Sony with one of the settings shown in the following tables (in the tables, 1 = On and 0 = Off).

Note Robot vendors and hardware resellers may change the default drive switch settings.

Switch	Setting
1	0
2	0
3	0
4	0
5	0
6	0
7	1
8	1

Switch	Setting
1	0
2	0
3	0

Switch	Setting
4	0
5	1
6	0
7	1
8	0

Switches 1 thru 4 are for setting the OS type. Switches 5 thru 8 can be usually be left set at the default position.

Sony documentation (*UNIX Configuration Guide, V2.xx*) states that switches 1 thru 4 can be left set at the default position. However, the drive firmware must be at (or above) one of the following levels:

- 0700 (SDX-400C)
- 0107 (SDX-500C)

Use the following settings for switches 1 thru 4, if either of the following is true:

- The drive has older firmware and you choose not to update the firmware.
- The drive is a SDX-300C drive.

Switch	Setting
1	1
2	1
3	0
4	0

Adding Quantum DLT 7000 Drives

For the IRIX system to recognize these drives, specific entries are needed in the kernel.

```
/* DEC DLT7000 drive */
{ DECDLT, TPDLT, 7, 7, "QUANTUM", "DLT7000", 0, 0,
{0, 0, 0, 0}, MTCAN_BSF|MTCAN_BSR|MTCAN_APPEND|MTCAN_LEOD|
```


```
MTCAN_CHKRDY|MTCAN_VAR|MTCAN_SETSZ|MTCAN_SILI|MTCAN_SEEK|
MTCAN_SYNC|MTCAN_CHTYPEANY|MTCAN_COMPRESS,
20, 8*60, 20*60, 5*60, 4096, 64*1024, 0, (u_char*)0 },
```

Adding Quantum DLT8000 Drives or Stackers

For the system to recognize these drives, specific entries are needed in the kernel.

The section used to define arrays for density counts and density names must contain the following entry:

```
#define tpsc_dlt8000_dens_count 2
char *tpsc_dlt8000_hwg_dens_names[] = { "8000", "8000_compress" };
char *tpsc_dlt8000_alias_dens_names[] = { ".8000", ".8000c" };
```

The `struct tpsc_types tpsc_types[]` array must contain the following entry:

```
/* DEC THZxx DLT drive */
{ DECDLT, TPDLT, 0, 7, "QUANTUM", "DLT8000", 0, 0,
{0 /*8000*/, 0 /*8000c*/ },
MTCAN_BSF|MTCAN_BSR|MTCAN_APPEND|MTCAN_SPEOD |
MTCAN_CHKRDY|MTCAN_VAR| MTCAN_SETSZ|MTCAN_SILI|MTCAN_SEEK|
MTCAN_SYNC|MTCAN_CHTYPEANY|MTCAN_COMPRESS|MTCAN_SETDEN,
20, 8*60, 20*60, 5*60, 3*3600, 4096, 64*1024,
tpsc_dlt8000_dens_count, tpsc_dlt8000_hwg_dens_names,
tpsc_dlt8000_alias_dens_names,
{0}, 0, 0, 0,
0, (u_char *)0 },
```

Adding Quantum SDLT220 Drives

For the IRIX system to recognize these drives, specific entries are needed in the kernel.

```
/* This is the config without compression */
{ DECDLT, TPDLT, 7, 9, "QUANTUM", "SuperDLT1", 0 , 0,
{0, 0, 0, 0 }, MTCAN_BSF | MTCAN_BSR | MTCAN_APPEND | MTCAN_SPEOD
| MTCAN_CHKRDY | MTCAN_VAR | MTCAN_SETSZ | MTCAN_SILI | MTCAN_SEEK
| MTCAN_SYNC | MTCAN_CHTYPEANY,
20, 8*60, 20*60, 5*60, 3*3600, 16384, 64*1024,
tpsc_default_dens_count, tpsc_default_hwg_dens_names,
tpsc_default_alias_dens_names,
{0}, 0, 0, 0, 0, (u_char *)0 },
```

Adding Exabyte Mammoth2 Compression Drives

For the IRIX system to recognize these drives, specific entries are needed in the kernel.

```
/* EXABYTE Mammoth2 */
{ EXABYTE8900, TP8MM_8900, 7, 8, "EXABYTE", "Mammoth2", 0,0,
{0, 0, 0, 0},
MTCAN_BSF|MTCAN_BSR|MTCAN_PREV|MTCAN_CHKRDY|MTCAN_VAR|MTCAN_SETSZ|
MTCAN_SILI|MTCAN_CHTYPEANY|MTCAN_SPEOD|MTCAN_SYNC|
MTCAN_SEEK|MTCAN_COMPRESS,
80, 4*60, 25*60, 5*60, 5*60, 1024, 128*1024,
tpsc_default_dens_count, tpsc_default_hwg_dens_names,
tpsc_default_alias_dens_names,
{0}, 0, 0, 0,
0, (u_char *)0 },
```

Adding IBM 3590E Drives

For the IRIX system to recognize these drives, specific entries are needed in the kernel.

```
/* IBM NTP drive */
{ IBMNTP, TPNTP, 3, 8, "IBM", "03590E1A", 0, 0, {0, 0, 0, 0},
MTCAN_BSF | MTCAN_BSR | MTCAN_SPEOD | MTCANT_RET | MTCAN_CHKRDY |
MTCAN_PREV | MTCAN_SEEK | MTCAN_APPEND | MTCAN_SILI | MTCAN_VAR |
MTCAN_SETSZ | MTCAN_CHTYPEANY | MTCAN_COMPRESS | MTCANT_LOAD |
MTCAN_LDREW ,
20, 14*60, 67*60, 14*60, 67*60, 16384, 512*1024,
tpsc_default_dens_count, tpsc_default_hwg_dens_names,
tpsc_default_alias_dens_names,
"\300\0\0\0\30\0", 6, 0, 0,
0, (u_char *)0 },

/* IBM NTPSTACKER drive */
{ IBMNTP, TPNTPSTACKER, 3, 8, "IBM", "03590E11", 0, 0, {0, 0, 0,
0},
MTCAN_BSF | MTCAN_BSR | MTCAN_SPEOD | MTCANT_RET | MTCAN_CHKRDY |
MTCAN_PREV | MTCAN_SEEK | MTCAN_APPEND | MTCAN_SILI | MTCAN_VAR |
MTCAN_SETSZ | MTCAN_CHTYPEANY | MTCAN_COMPRESS | MTCANT_LOAD |
MTCAN_LDREW ,
20, 14*60, 67*60, 14*60, 67*60, 16384, 512*1024,
tpsc_default_dens_count, tpsc_default_hwg_dens_names,
tpsc_default_alias_dens_names,
"\300\0\0\0\30\0", 6, 0, 0,
0, (u_char *)0 },
```


Adding STK 9840 or T9940A FC Drives

For the IRIX system to recognize these drives, the following specific entry is needed in the kernel.

```
/* STK 9940 drive */
{ STK9840, TPSTK9840, 3, 4, "STK", "T9940A", 0, 0, {0, 0, 0, 0},
MTCAN_BSF | MTCAN_BSR | MTCANT_RET | MTCAN_CHKRDY | MTCAN_PREV |
MTCAN_SPEOD | MTCAN_SEEK | MTCAN_APPEND | MTCAN_SILI | MTCAN_VAR |
MTCAN_SETSZ | MTCAN_CHTYPEANY | MTCAN_COMPRESS,
20, 8*60, 10*60, 3*60, 3*60, 16384, 256*1024,
tpsc_default_dens_count, tpsc_default_hwg_dens_names,
tpsc_default_alias_dens_names,
{0}, 0, 0, 0,
0, (u_char *)0 },
```

Changing the /var/sysgen/master.d/scsi File

The IRIX tape driver (`tpsc`) provides support for all devices currently supported by VERITAS and coexists with the tape support (`ts`) system. You can choose which driver interface (`ts` or `tpsc`) to use for a particular device type.

The tape support system consists of a tape support driver, personality daemons, and a daemon to manage the personality daemons from SGI or the tape device vendor. For information on how to configure the tape support system, see the `ts` and `ts.config` man pages.

Adding Drive Support

To add support for a new standalone or robotic drive for either driver interface, the appropriate array must contain entries for the drives, as follows:

- For the `ts` driver interface, put entries in `struct ts_types ts_types[]`.
- For the `tpsc` driver interface, put entries in `struct tpsc_types tpsc_types[]`.

These entries are required for the system to recognize the drives. You can find this array in `/var/sysgen/master.d/scsi`.

1. No further changes are necessary if the following are both true:
   - The appropriate entries for the drive are present in this file.
   - You have previously rebuilt the kernel and modified `MAKEDEV` as explained in "Reconfiguring the Kernel and Modifying the MAKEDEV Script".
2. If the entries are *not* in the file, add them as follows:

a. Save a copy of `/var/sysgen/master.d/scsi`.

b. Add the entries. An easy way to make this addition is to copy the entries from the `MediaMgr_DeviceConfig_Guide.txt` file. See "Considerations When Using This Guide" on page 1.

c. Reconfigure the kernel as explained in "Reconfiguring the Kernel and Modifying the MAKEDEV Script".

Reconfiguring the Kernel and Modifying the MAKEDEV Script

If you made any changes to the `/var/sysgen/master.d/scsi` file, then you must reconfigure the kernel and modify the `MAKEDEV` script as follows:

1. Run the following kernel auto-configuration script:

   ```
   /etc/autoconfig
   ```

2. If you are using the tape support (`ts`) system, run `chkconfig -f ts on`.

3. Reboot the system to utilize the newly-built kernel.

Configuring Optical Disk Drives

When adding optical disk drives to a Media Manager configuration, you must specify the following device paths:

- Character device path (disk partition `s7`)
- Volume header disk device path (disk partition `vh`)

In a typical SGI IRIX configuration, most of the desired optical disk device files already exist and you just have to locate them in the `/dev` directory.

Character disk device files have the following format:

`/dev/rdsk/dks`*CONTROLLER*`d`*TARGET*`s7`

Volume disk device files have the following format:

`/dev/rdsk/dks`*CONTROLLER*`d`*TARGET*`vh`

Where:

CONTROLLER is the SCSI bus (adapter) number.

TARGET is the SCSI ID.

`s7` is the desired character device partition.

`vh` is the desired volume header partition.

Examples of Optical Disk Device Files

If the desired optical disk drive is on SCSI bus 1 at SCSI ID 3, you specify the following paths:

```
/dev/rdsk/dks1d3vh  (volume header)
/dev/rdsk/dks1d3s7  (character device)
```

Command Summary

The following is a summary of commands that may be useful when configuring devices. See the procedures in this chapter for examples of their usage.

`MAKEDEV` *Type*

If the device files you need do not exist, you can execute this command from the `/dev` directory to create them.

Type indicates the type of device file, as follows:

`tps` creates all the tape device file combinations for `tps` (the SCSI tape driver for Integral SCSI controllers)

`scsi` creates all the device files for the generic SCSI driver.

`dks` creates all the device files for `dks` (the SCSI disk driver for integral SCSI controllers).

`/etc/autoconfig`

Runs the kernel auto-configuration script.

`/sbin/hinv -v`

Shows the system configuration, including devices configured on SCSI controllers. This command also can be used to verify that a tape drive is recognized by the system.

`/sbin/uname -aR`

Shows what operating system is currently running.

`mt status`

Prints status information for the tape unit.

`chkconfig -f ts on`

Enables the (ts) tape support system.

Compaq Alpha Running TRU64 UNIX 4.0F/4.0G 6

This chapter explains how to configure devices for use with Media Manager on a Compaq Alpha platform. You configure drives and robots using one of the available Media Manager administrative interfaces.

The main topics included in this chapter are

- Configuring Robotic Controls
- Configuring Tape Drives
- Command Summary

Configuring Robotic Controls

Robots are controlled through a SCSI or a network connection.

- SCSI robotic control is covered in the following section.
- Configuration for network controlled robotic libraries is discussed in the appendices of the Media Manager system administrator's guide.

Configuring SCSI Robotic Controls

Read this topic if you plan to use a robotic storage device that is controlled through a SCSI robotic connection. Information on supported SCSI robots (vendor models) can be found on the VERITAS support web site.

When communicating with SCSI-controlled robotic peripherals, Media Manager robotic software utilizes the generic (user mode) SCSI passthru driver. The TRU64 UNIX kernel does not have to be reconfigured to use this driver, since this driver is part of basic TRU64 UNIX.

Creating SCSI Robotic Control Device Files

Media Manager requires that a special file exist in the `/dev` directory for SCSI controlled robotics. Create the device files using the following command:

```
cd /dev
/sbin/mknod ROBTYPEcBUStTARGETlLUN c 38 MINOR
```

Where:

ROBTYPE is the robot type in lower case (for example, tsd).

BUS is the bus (adapter) number.

TARGET is the SCSI ID.

LUN is the logical unit number.

MINOR = `(bus * 256) + (target * 16) + lun`

Examples of SCSI Robotic Control Device Files

Example 1

If the robotics control for an Exabyte 10i (TS8) is connected to bus 0 at SCSI ID 5, lun 0, the commands to create the device file are as follows:

```
cd /dev
/sbin/mknod ts8c0t5l0 c 38 80
```

This creates the following device file, which you specify to Media Manager:

```
/dev/ts8c0t5l0
```

Example 2

If the robotics control for a Quantum DLT2700 (TSD) is connected on bus 1 at SCSI ID 3, lun 1, the commands to create the device file would be

```
cd /dev
/sbin/mknod tsdc1t3l1 c 38 305
```

This creates the following device file, which you specify to Media Manager:

```
/dev/tsdc1t3l1
```

The `lsdev` command can be used to determine what devices are physically connected to the system. An example for determining connected autochangers (robotic libraries) follows. This example shows that there is only one autochanger connected to this system.

```
/usr/openv/volmgr/bin/lsdev changer
Bus 0 Scsi Id 5 Lun 0, Changer: EXABYTE EXB-10i 3.0
```


Configuring Tape Drives

Fast-Tape Positioning (locate-block)

For most drive types, Media Manager supports the SCSI `locate-block` command for positioning a tape to a specific block.

NetBackup uses the `locate-block` command by default unless you disable it by executing the following:

```
touch /usr/openv/volmgr/database/NO_LOCATEBLOCK
```

With `locate-block` positioning disabled, NetBackup uses the `forward-space-file/record` method of tape positioning

Adding Standard Tape Drives

When adding tape drives to a Media Manager configuration, you need to specify a no rewind on close device path.

Note These are LUN 0 tape drives.

These device files are located in the `/dev` directory and have the following format:

`/dev/nrmt`*LTUDensity*

Where:

LTU is the logical tape unit. When the first `MAKEDEV` of a tape drive is done, *LTU* is 0. The next time, *LTU* is 1, and so on.

Values for *Density* can be `l`, `m`, `h`, or `a`. Typically, `h` (for high) is used.

Creating No Rewind Device Files

If the desired tape device file does not exist, create device files using the `MAKEDEV` command as follows:

```
cd /dev
./MAKEDEV tzN
```

Where *N* is (*bus* * 8) + *SCSI ID*

Media Manager provides the `lsdev` command that you can use to determine the devices that are physically connected to the system. This command is located in `/usr/openv/volmgr/bin`.

An example of using `lsdev` to determine connected tape drives follows:

```
lsdev tape

Bus 0 Scsi Id 3 Lun 0, Tape (rmt2): EXABYTE EXB-8500-85Qanx005E0
Bus 0 Scsi Id 4 Lun 0, Tape (rmt0): EXABYTE EXB-850085QANXRC05E0
```

You can also use the following form of the command:

```
lsdev logical_tape_devs

rmt2 is defined on bus 0, scsi id 3
rmt0 is defined on bus 0, scsi id 4
```

If the device files do not exist for a connected tape drive, the command shows (----) instead of `rmt`*LTU*, for example:

```
lsdev tape
```

The output shows that the device files for the tape drive on bus 0, SCSI ID 4 do not exist.

```
Bus 0 Scsi Id 3 Lun 0, Tape (rmt2): EXABYTE EXB-8500-85Qanx005E0
Bus 0 Scsi Id 4 Lun 0, Tape (----): EXABYTE EXB-850085QANXRC05E0
```

To create device files, use the `MAKEDEV` command.

```
cd /dev
./MAKEDEV tz4
```

The output is as follows:

```
MAKEDEV: special file(s) for tz4:
rmt0l
rmt0h
rmt0m
rmt0a
nrmt0l
nrmt0h
nrmt0m
nrmt0a
```

Note Only the no rewind device files are needed for configuration (the last four in the list).

Configuring Fibre Channel Tape Drives

When adding tape drives to a Media Manager configuration, you need only specify a no rewind on close device path. These device files are located in the `/dev` directory, and have the following format:

`/dev/nrmt`*LTUDensity*

Where:

LTU is the logical tape unit.

Values for *Density* can be `l`, `m`, `h`, or `a`. Typically, `h` (for high) is used.

If the desired tape device file does not exist, you can create device files using the `mknod` command. Most fibre channel tape drives have a LUN other than 0.

The commands in the example use the following format:

`mknod /dev/nrmt`*LTUDensity* `c 9` *calc*

Where:

calc = (*LUN* * 64) + (*target_ID* * 1024) + (*bus_number* * 16384) + (*den* * 2) + *rewind*

den = 0 for low, 1 for high, 2 for medium, or 3 for auxiliary density.

rewind = 0 for rewind and 1 for no rewind (use 1 for no rewind on close device files).

Fibre Channel Example

The following example uses the formula to add a SCSI tape device with LUN 3, target ID 4, and bus number 2:

1. Perform the following calculation for the no rewind device files, depending on the density of the device:

```
low  density: (3 * 64) + (4 * 1024) + (2 * 16384) + (0 * 2) + 1 = 37057
high density: (3 * 64) + (4 * 1024) + (2 * 16384) + (1 * 2) + 1 = 37059
med  density: (3 * 64) + (4 * 1024) + (2 * 16384) + (2 * 2) + 1 = 37061
aux  density: (3 * 64) + (4 * 1024) + (2 * 16384) + (3 * 2) + 1 = 37063
```

2. Create the no rewind device files. *LTU* must be a unique number.

```
# mknod /dev/nrmtLTUl c 9 37057
# mknod /dev/nrmtLTUh c 9 37059
# mknod /dev/nrmtLTUm c 9 37061
# mknod /dev/nrmtLTUa c 9 37063
```

Examples of No Rewind Device Files

Example 1

If the desired Exabyte 8500 tape drive is on bus 0 at SCSI ID 4, the commands to create the device files follow:

```
cd /dev
./MAKEDEV tz4
```

This creates the following device file, which you specify to Media Manager (this example assumes *LTU* is 0):

```
/dev/nrmt0h
```

Example 2

If the desired DLT4000 tape drive is on bus 1 at SCSI ID 3, the commands to create the device files are as follows:

```
cd /dev
./MAKEDEV tz11
```

This creates the following device file, which you specify to Media Manager (this example assumes *LTU* is 1):

```
/dev/nrmt1h
```

Adding Nonstandard Tape Drives

VERITAS has tested several tape drives on TRU64 UNIX, including EXABYTE 8-mm drives, HP 4-mm DAT drives, and Quantum DLT drives.

Normally using tape drives from these vendors does not require kernel reconfiguration because the default definitions are sufficient. If a drive vendor recommends kernel reconfiguration, the file that contains the tape drive definitions is `/usr/sys/data/cam_data.c`.

If this file is modified

- Care should be taken to ensure tape drives are configured in variable (rather than fixed) mode.
- Refer to the `doconfig` (8) command for information on rebuilding a new kernel.

Switch Settings for HP C1533A 4mm DAT Drives

If you have standalone or robotic 4MM drives that are model HP C1533A, you may have to change the switch settings on the bottom of the drive. This drive comes in the HP C1560B (48AL) DAT Autoloader.

If the drive or autoloader was purchased from Hewlett Packard, the default switch settings should work. However, if the device was purchased from some other vendor, that vendor may have changed the default switch settings. The same thing may apply to other vendor's 4MM robots if they contain HP C1533A drives.

If this situation exits, set the switch settings as shown in the following table (the documented default). In the table, 1 = On and 0 = Off.

Switch	Setting
1	1
2	1
3	0
4	1
5	1
6	1
7	1
8	1

Switch Settings for Sony AIT Drives

Sony AIT drives have 8 dip switches located on the bottom of the drive. It is important to set these switches correctly, even if it means taking the drives out of robots and checking them.

Some robots (for example, Spectra Logic) provide a way to set the drive switches from the robot itself. For Spectra Logic robots, it doesn't matter what the drive switches are. The Treefrog (215) robot has a dial in the back to set the appropriate OS. The Bullfrog (10000) robot has a means of setting the OS through a touch screen.

Depending on the version of the AIT drive, drives are shipped from Sony with one of the settings shown in the following tables (in the tables, 1 = On and 0 = Off).

Note Robot vendors and hardware resellers may change the default drive switch settings.

Switch	Setting
1	0
2	0
3	0
4	0
5	0
6	0
7	1
8	1

Switch	Setting
1	0
2	0
3	0
4	0
5	1
6	0
7	1
8	0

Switches 1 thru 4 are for setting the OS type. Switches 5 thru 8 can be usually be left set at the default position.

Sony documentation (*UNIX Configuration Guide, V2.xx*) states that switches 1 thru 4 can be left set at the default position. However, the drive firmware must be at (or above) one of the following levels:

- 0404 (SDX-300C)
- 0700 (SDX-400C)
- 0107 (SDX-500C)

If the drive has older firmware, update the firmware or use the following settings for switches 1 thru 4:

Switch	Setting
1	0
2	0
3	1
4	1

Command Summary

The following is a summary of commands that may be useful when configuring devices. See the procedures in this chapter for usage examples.

`/sbin/mknod` *ROBTYPE*`c`*BUS*`t`*TARGET*`l`*LUN* `c 38` *MINOR*

Execute this command from the `/dev` directory to create the special device file for SCSI controlled robotics.

Where:

ROBTYPE is the robot type in lower case (for example, ts8).

BUS is the bus (adapter) number.

TARGET is the SCSI ID.

LUN is the logical unit number.

MINOR = `(bus * 256) + (target * 16) + lun`

`/sbin/mknod /dev/nrmt`*LTUDensity* `c 9` *calc*

Execute this command to can create tape device files.

Where:

LTU is the logical tape unit and values for *Density* can be `l`, `m`, `h`, or `a`.

calc = (*LUN* * 64) + (*target_ID* * 1024) + (*bus_number* * 16384) + (*den* * 2) + *rewind*

den = 0 for low, 1 for high, 2 for medium, or 3 for auxiliary density.

rewind = 0 for rewind and 1 for no rewind.

`./MAKEDEV ace0`

Creates device files for the serial ports. Normally, these files exist after the system is installed. Execute this command from the `/dev` directory.

`./MAKEDEV tz`*N*

Where *N* is `(`*bus* `* 8)` + *SCSI ID.*

Creates device files for tape drives. Execute this command from the `/dev` directory.

`/usr/openv/volmgr/bin/lsdev tape`

Displays tape devices that are physically connected to the system.

`scu sh edt`

Displays the CAM equipment data table (EDT).

`scu sc edt`

Scans for devices and places them in the CAM equipment data table (EDT).

Compaq Alpha Running TRU64 UNIX 5.0a/5.1/5.1a 7

This chapter explains how to configure devices for use with Media Manager on a Compaq Alpha platform. You configure drives and robots using one of the available Media Manager administrative interfaces.

The main topics included in this chapter are

- Configuring Robotic Controls
- Configuring Tape Drives
- Command Summary

Configuring Robotic Controls

Robots are controlled through a SCSI or a network connection.

- SCSI robotic control is covered in the following section.
- Configuration for network controlled robotic libraries is discussed in the appendices of the Media Manager system administrator's guide.

Configuring SCSI Robotic Controls

Read this topic if you plan to use a robotic storage device that is controlled through a SCSI robotic connection. Information on supported SCSI robots (vendor models) can be found on the VERITAS support web site.

When communicating with SCSI-controlled robotic peripherals, Media Manager robotic software utilizes the SCSI passthru capability of the media changer driver. The TRU64 UNIX kernel does not have to be reconfigured to use this driver since this driver is part of basic TRU64 UNIX.

Creating SCSI Robotic Control Device Files

Media Manager requires that device files from the `/dev/changer` directory are used to configure SCSI robotic control. These files are created by the operating system at boot time.

To display devices that are available on the system, use the following command:

```
# hwmgr -view devices

HWID: Device Name         Mfg      Model             Location
------------------------------------------------------------
 3:  /dev/kevm
34:  /dev/disk/floppy0c            3.5in floppy      fdi0-unit-0
55:  /dev/disk/cdrom0c   COMPAQ   CDR-8435          bus-1-targ-0-lun-0
56:  /dev/disk/dsk0c     DEC      RZ2DD-KS (C) DEC  bus-2-targ-0-lun-0
57:  /dev/disk/dsk1c     DEC      RZ2DD-KS (C) DEC  bus-2-targ-1-lun-0
58:  /dev/disk/dsk2c     DEC      RZ2DD-KS (C) DEC  bus-2-targ-2-lun-0
70:  /dev/changer/mc0              C6280-4000        bus-2-targ-3-lun-0
71:  /dev/ntape/tape6    Quantum  DLT4000           bus-2-targ-4-lun-0
```

Example of SCSI Robotic Control Device File

The previous output the from `hwmgr` command shows an HP C6280 robot connected on bus 2 at SCSI ID 3, lun 0. The corresponding device file is `/dev/changer/mc0`. This file should be used to configure the device in Media Manager.

Configuring Tape Drives

Fast-Tape Positioning (locate-block)

For most drive types, Media Manager supports the SCSI `locate-block` command for positioning a tape to a specific block.

NetBackup uses the `locate-block` command by default unless you disable it by executing the following:

```
touch /usr/openv/volmgr/database/NO_LOCATEBLOCK
```

With `locate-block` positioning disabled, NetBackup uses the `forward-space-file/record` method of tape positioning.

Adding Standard Tape Drives

Media Manager requires that no rewind on close device files are used to configure tape drives. These files are created by the operating system at boot time. The device files are located in the `/dev/ntape` directory and have the following format:

`/dev/ntape/tape`*`ID`*`_d`*`DENSITY`*

Where:

ID is the identification number of the tape device.

Values for *DENSITY* can be 0 - 7. Typically, 1 (for high density) is used.

To display devices that are available on the system, use the following command:

```
# hwmgr -view devices

HWID: Device Name         Mfg      Model            Location
------------------------------------------------------------
 3:  /dev/kevm
34:  /dev/disk/floppy0c            3.5in floppy     fdi0-unit-0
55:  /dev/disk/cdrom0c   COMPAQ   CDR-8435         bus-1-targ-0-lun-0
56:  /dev/disk/dsk0c     DEC      RZ2DD-KS (C) DEC bus-2-targ-0-lun-0
57:  /dev/disk/dsk1c     DEC      RZ2DD-KS (C) DEC bus-2-targ-1-lun-0
58:  /dev/disk/dsk2c     DEC      RZ2DD-KS (C) DEC bus-2-targ-2-lun-0
70:  /dev/changer/mc0              C6280-4000       bus-2-targ-3-lun-0
71:  /dev/ntape/tape6    Quantum  DLT4000          bus-2-targ-4-lun-0
```

No Rewind Device File Example

The previous output the from `hwmgr` command shows a Quantum DLT 4000 tape drive connected on bus 2 at SCSI ID 4, lun 0. The corresponding device file is `/dev/ntape/tape6`.

After adding a density suffix of `_d1` (for high density) the device file is `/dev/ntape/tape6_d1`. This file should be used to configure the device in Media Manager.

Adding Nonstandard Tape Drives

This topic applies to the following drive types:

- HP LTO
- Seagate LTO
- STK 9840

Some types of tape drives require changes to the kernel before you can use them on Tru64. To make changes to the kernel do the following:

1. Add the appropriate device-specific entries to the `/etc/ddr.dbase` file (see “Device-Specific Entry for HP Ultrium 230e” on page 113 and the other entries that follow).

2. Compile the `/etc/ddr.dbase` file using the `ddr_config` (Dynamic Device Recognition) utility to create the `ddr.db` file.

Refer to the following man pages for more information on Dynamic Device Recognition (DDR):

- `ddr.dbase(4)`
- `ddr_config(8)`

Device-Specific Entry for HP Ultrium 230e

```
SCSIDEVICE
Type = tape
Name = "HP      " "Ultrium"

PARAMETERS:
MaxTransferSize = 0xffffff
ReadyTimeSeconds = 0x2d
InquiryLength = 0x20
```


```
DENSITY:
DensityNumber = 0
OneFileMarkOnClose = yes
DensityCode = 0x00
Blocking = 0
CompressionCode = 0x0
Buffered = 0x1

DENSITY:
DensityNumber = 1
OneFileMarkOnClose = yes
DensityCode = 0x00
Blocking = 0
CompressionCode = 0x1
Buffered = 0x1
```

Device-Specific Entry for Seagate Viper 200 - LTO

```
SCSIDEVICE
Type = tape
Name = "SEAGATE" "ULTRIUM"

PARAMETERS:
TypeSubClass          = tk
TagQueueDepth         = 0
MaxTransferSize       = 0x0ffffff # (16MB - 1)
ReadyTimeSeconds      = 180 # seconds
CMD_PreventAllow      = supported
CMD_ExtReserveRelease = supported
BlockSize             = 0
PwrMgmt_Capable       = false

DENSITY:
DensityNumber   = 0,2,3,4,5,6,7
DensityCode     = default
CompressionCode = 0x0
Buffered        = 0x1

DENSITY:
DensityNumber   = 1
DensityCode     = default
CompressionCode = 0x1
Buffered        = 0x1
```

Device-Specific Entry for STK 9840

```
SCSIDEVICE
Type = tape
Stype = 2
Name = "STK" "9840"

PARAMETERS:
TypeSubClass          = 3480
BlockSize             = 0
MaxTransferSize       = 0x40000     # 256k
SyncTransfers         = enabled
WideTransfers         = enabled
Disconnects           = enabled
CmdReordering         = disabled
TaggedQueuing         = disabled
TagQueueDepth         = 0
WCE_Capable           = false
PwrMgmt_Capable       = false
LongTimeoutRetry      = disabled
ReadyTimeSeconds      = 240
DisperseQueue         = false
CMD_PreventAllow      = supported
CMD_ExtReserveRelease = supported

DENSITY:
#
# /dev/tape/tapeX_d0, _d4
#
DensityNumber = 0,4
DensityCode = 0x42
CompressionCode = 0
Buffered = 0x1
#
DENSITY:
#
# /dev/tape/tapeX_d1, _d5
#
DensityNumber = 1,5
DensityCode = 0x42
CompressionCode = 1
#
DENSITY:
#
# /dev/tape/tapeX_d2, _d6
#
```


```
DensityNumber = 2,6
DensityCode = 0x43
CompressionCode = 0
Buffered = 0x1
```

Switch Settings for HP C1533A 4mm DAT Drives

If you have standalone or robotic 4MM drives that are model HP C1533A, you may have to change the switch settings on the bottom of the drive. This drive comes in the HP C1560B (48AL) DAT Autoloader.

If the drive or autoloader was purchased from Hewlett Packard, the default switch settings should work. However, if the device was purchased from another vendor, that vendor may have changed the default switch settings. The same thing may apply to other vendor's 4MM robots if they contain HP C1533A drives.

If this situation exits, set the switch settings as shown in the following table (the documented default). In the table, 1 = On and 0 = Off.

Switch	Setting
1	1
2	1
3	0
4	1
5	1
6	1
7	1
8	1

Switch Settings for Sony AIT Drives

Sony AIT drives have 8 dip switches located on the bottom of the drive. It is important to set these switches correctly, even if it means taking the drives out of robots and checking them.

Some robots (for example, Spectra Logic) provide a way to set the drive switches from the robot itself. For Spectra Logic robots, it doesn't matter what the drive switches are. The Treefrog (215) robot has a dial in the back to set the appropriate OS. The Bullfrog (10000) robot has a means of setting the OS through a touch screen.

Depending on the version of the AIT drive, drives are shipped from Sony with one of two settings as shown in the following tables (in the tables, 1 = On and 0 = Off).

Note Robot vendors and hardware resellers may change the default drive switch settings.

Switch	Setting
1	0
2	0
3	0
4	0
5	0
6	0
7	1
8	1

Switch	Setting
1	0
2	0
3	0
4	0
5	1

Switch	Setting
6	0
7	1
8	0

Switches 1 thru 4 are for setting the OS type. Switches 5 thru 8 can be usually be left set at the default position.

Sony documentation (*UNIX Configuration Guide, V2.xx*) states that switches 1 thru 4 can be left set at the default position. However, the drive firmware must be at (or above) one of the following levels:

- 0404 (SDX-300C)
- 0700 (SDX-400C)
- 0107 (SDX-500C)

If the drive has older firmware, update the firmware or use the following settings for switches 1 thru 4:

Switch	Setting
1	0
2	0
3	1
4	1

Command Summary

The following is a summary of commands that may be useful when configuring devices. See the procedures in this chapter for usage examples.

```
hwmgr -view devices
```

Displays all devices on the system.

Intel Hosts Running Red Hat Linux 6.2/7.0/7.1 8

This chapter explains how to configure devices for use with Media Manager on a Intel host platform running Red Hat Linux. You can configure robots and drives for Media Manager using one of the available Media Manager administrative interfaces.

The main topics included in this chapter are

- Before You Start
- Configuring Robotic Controls
- Configuring Tape Drives
- Verifying The Device Configuration
- Utilities to Test SCSI Devices
- Command Summary

Before You Start

Observe the following important points when performing the tasks described in this chapter:

- Verify that the st (tape) device driver is installed or loaded in the kernel. This driver allows the use of SCSI tape drives.
- Verify that the sg device driver is installed or loaded in the kernel. This driver allows passthru commands to SCSI tape drives and control of robotic devices.

  Use the following command to display and verify that these modules are loaded in the kernel: `/sbin/lsmod`. Example output from the `lsmod` command follows:

```
Module          Size           Used by
sg             14844             0
st             24556             0
```

  The standard Red Hat Linux 6.2 distribution has these modules available for loading. When running Red Hat Linux, these modules are dynamically loaded as needed.

If you see problems which leave NetBackup Media Manager processes hanging because the modules are unloaded, VERITAS suggests you install these modules in the kernel by building a custom kernel. This is documented in the Red Hat Linux Reference Guide.

To load these modules if they are not in the kernel, use the following commands:

```
/sbin/insmod st
/sbin/insmod sg
```

To install the st and sg modules in the kernel, use the standard system utilities to modify the configuration file in `/usr/src/linux` and then rebuild the kernel. If the `/linux` directory is not there, install the kernel source code.

- During NetBackup installation the binary `/usr/openv/volmgr/bin/make_scsi_dev` is run and does the following:
  - Creates the directories `/dev/sg` and `/dev/st`.
  - Obtains the device file output generated by the sg and st drivers.
  - Creates device files of the format required for Media Manager and places them in these directories. See "Configuring SCSI Robotic Control Device Paths" on page 123 and "Adding Standard Tape Drives" on page 124.
- Verify that a SCSI low-level driver is installed for each HBA in your system. Refer to your HBA vendor documentation for instructions.

If You Are Using NetBackup BusinesServer

Portions of this chapter may include configuration topics and examples for peripherals that are not supported in NetBackup BusinesServer. It is important to refer to the VERITAS support web site to determine which Media Manager robot types, robots, and drives are supported for NetBackup BusinesServer, before using this chapter.

Configuring Robotic Controls

To use robotics, the following drivers must be configured in the kernel or loaded as modules:

- Standard SCSI driver.
- SCSI-adaptor driver.
- Linux SCSI generic (sg) driver.

Information on supported SCSI robots (vendor models) can be found on the VERITAS support web site.

Configuring SCSI Robotic Control Device Paths

The binary `/usr/openv/volmgr/bin/make_scsi_dev` creates device files in the `/dev/sg` directory. The names of these files in this directory have the following format:

h*HOST*c*CHANNEL*t*TARGET*l*LUN*

Where:

HOST is the host bus adaptor.

CHANNEL is channel.

TARGET is the target ID.

LUN is the logical unit number.

Use the files in the `/dev/sg` directory for the robotic path when using Media Manager interfaces to configure robots.

There must be a `/dev/sg`*N* entry (where *N* is a decimal number from 0 to 255) for each device. These entries are needed by `make_scsi_dev`, which is run during the NetBackup installation.

If you have devices that are not being discovered by Media Manager device discovery, it may be because the Linux default number of these entries is not sufficient. You may need to create additional entries.

Create device entries as follows, where *N* is a decimal number from 0 to 255. After creating all entries necessary, rerun `make_scsi_dev`.

`mknod /dev/sg`*N* `c 21` *N*

Examples of SCSI Robotic Control Device Files

```
h10c0t1l0
h10c0t2l0
h10c0t3l0
h25c0t0l0
h25c0t1l0
```

Configuring Tape Drives

To use SCSI tape drives, the following drivers must be configured in the kernel or loaded as modules:

- Standard SCSI driver

- SCSI-adaptor driver
- SCSI tape (st) driver
- Linux SCSI generic (sg) driver

The Importance of Using the Passthru Driver for Tape Drives

NetBackup can function without a passthru path to tape devices, *but* the following capabilities are not available:

- Locate-block (fast-tape) positioning
- Data protection provided by SCSI reserve / release
- Automatic configuration for tape devices
- Performance optimization for Quantum SDLT drives

Enabling Passthru Paths

Several sg device files are provided by default in the typical Linux installation. If you have more devices than the default number provided, you need to create the needed device files.

There must be a `/dev/sg`*N* entry (where *N* is a decimal number from 0 to 255) for each device. These entries are needed by `make_scsi_dev`, which is run during the NetBackup installation.

If you have devices that are not being discovered by Media Manager device discovery, it may be because the Linux default number of these entries is not sufficient. You may need to create additional entries.

Create device entries as follows, where *N* is a decimal number from 0 to 255. After creating all entries necessary, rerun `make_scsi_dev`.

`mknod /dev/sg`*N* `c 21` *N*

Adding Standard Tape Drives

`make_scsi_dev` creates device files in the `/dev/st` directory. The names of the no rewind device files in this directory have the following format:

`nh`*HOST*`c`*CHANNEL*`t`*TARGET*`l`*LUN*

Where:

`n` is the no rewind on close device file.

HOST is the host bus adaptor.

CHANNEL is channel.

TARGET is the target ID.

LUN is the logical unit number.

When adding tape drives to a Media Manager configuration, you need to specify a no rewind on close device path. Use the files in the `/dev/st` directory when configuring tape drives.

Examples of SCSI Tape Device Files

```
nh10c0t2l0
nh10c0t3l0
```

Verifying The Device Configuration

The file `/proc/scsi/scsi` shows all devices recognized by the SCSI driver. To verify that the operating system can see the devices, run the following command from a terminal window to view this file:

```
cat /proc/scsi/scsi
```

The output displayed should be similar to the following example:

```
Attached devices:
Host: scsi0 Channel: 00 Id: 01 Lun: 00
 Vendor: HP       Model: C7200-8000       Rev: 1040
 Type:   Medium Changer                   ANSI SCSI revision: 03
Host: scsi0 Channel: 00 Id: 02 Lun: 00
 Vendor: QUANTUM  Model: DLT8000          Rev: 010F
 Type:   Sequential-Access                ANSI SCSI revision: 02
Host: scsi0 Channel: 00 Id: 03 Lun: 00
 Vendor: QUANTUM  Model: DLT8000          Rev: 010F
 Type:   Sequential-Access                ANSI SCSI revision: 02
```

If the operating system can see your SCSI devices, Media Manager device discovery will also see the devices.

Utilities to Test SCSI Devices

You can manipulate tape devices with the system `mt` command. The man page for `MT(1)` explains how to do this.

Robots can be tested using the `robtest` utility in `/usr/openv/volmgr/bin`.

There is also a set of SCSI utilities available from the Linux SCSI Generic (sg) driver home page.

Command Summary

The following is a summary of commands that may be useful when configuring devices. See the appropriate sections of this chapter for examples of their usage.

`sbin/lsmod`

List loaded modules.

`sbin/insmod`

Install loadable kernel modules.

`/usr/sbin/reboot`

Stop and restart the system.

`/bin/mknod /dev/sg`*N* `c 21` *N*

Create SCSI generic device files. Where *N* is a decimal number from 0 to 255.

`/usr/openv/volmgr/bin/make_scsi_dev`

Create NetBackup device files.

NCR Running SVR4MP-RAS 3.02 9

This chapter explains how to configure devices for use with Media Manager on a NCR system. Configure drives and robots using one of the available Media Manager administrative interfaces.

The main topics covered here are as follows:

- NCR Device Files
- Configuring Robotic Controls
- Configuring Tape Drives

NCR Device Files

You do not need to install a passthru driver or run `mknod` commands to add new device files. (The device files are created automatically when the machine is rebooted after adding a new device.)

After you attach the hardware and boot the machine, locate your device file names in the `/etc/device.tab.rd` text file and use those device file names when configuring Media Manager.

Information about attached devices can be found in this text file, for example

```
------snippet 1 from /etc/device.tab.rd ------
c13t2d0s0:/dev/rmt/c13t2d0s0:::\
        removable="true" \
        id="Quantum  DLT4000         " \
        desc="Tape Drive" \
-----------------------------------------------

------snippet 2 from /etc/device.tab.rd ------
c13t4d0s0:/dev/rchg/c13t4d0s0:::\
        removable="true" \
        id="STK      9714            " \
        desc="Medium Changer Device" \
-----------------------------------------------
```

Configuring Robotic Controls

Robots are controlled through a SCSI or a network connection.

- From the previous example, an example robotic path for SCSI control is `/dev/rchg/c13t4d0s0`.
- Configuration for network controlled robotic libraries is discussed in the appendices of the Media Manager system administrator's guide.

Configuring Tape Drives

To configure a no rewind on close tape device, use the device file with the `nn` suffix. In the following example this device file would be: `/dev/rmt/c13t2d0s0nn`.

The following example list was created using the command `/usr/openv/volmgr/bin/tpconfig -d`:

```
Index  DriveName        DrivePath             Type   Shared     Status
*****  **********       *********             ****   *******    ******

  0    DRIVE2           /dev/rmt/c13t2d0s0nn  dlt    No          UP
         TLD(0) Definition    DRIVE=2
Currently defined robotics are:
  TLD(0) robotic path = /dev/rchg/c13t4d0s0,volume database host = ted
```

Note Currently supported devices is limited to STK9710 and STK9714 robots (SCSI or Automated Cartridge System control) with DLT2000/DLT4000 drives.

Sequent Running DYNIX/ptx 4.4.2/4.4.4-4.4.8/4.5/4.5.2 10

This chapter explains how to configure devices for use with Media Manager on a Sequent system running DYNIX. You configure drives and robots using one of the available Media Manager administrative interfaces.

The main topics covered in this chapter are as follows:

- Configuring Robotic Controls
- Configuring Tape Drives

Configuring Robotic Controls

Robots can be controlled through a SCSI or a network connection.

- Configuring SCSI robotic control is covered in the following section.
- Configuration for network controlled robotic libraries is discussed in the appendixes of the Media Manager system administrator's guide. These appendixes describe specific platform requirements and restrictions.

Configuring SCSI Robotic Controls

Information on supported SCSI robots (vendor models) can be found on the VERITAS support web site. Use the following procedure to configure a pseudo device file for the robot passthru capability:

1. The following display using `lsdev`, lists the devices in a system. This command uses the passthru capability to do an inquiry command. If `lsdev` works it is a good indicator that the robotics will also work.

```
/usr/openv/volmgr/bin/lsdev

Bus 0, target 0, lun 0, Disk: (IBM OEM DFHSS4E         4343)
Bus 0, target 1, lun 0, Disk: (SEAGATE ST15150W        0023)
Bus 0, target 3, lun 0, Tape: (EXABYTE EXB8500C8SQANXRU07J0)
Bus 0, target 4, lun 0, Tape: (TANDBERG TDC 3800        -07:)
Bus 0, target 5, lun 0, Cdrom: (PLEXTOR CD-ROM PX-6XCS  4.05)
Bus 0, target 7, lun 0, Processor: (SEQUENT CSM SCSI Ctlr 0601)
Bus 0, target 8, lun 0, Disk: (HP     C2490A           5083)
Bus 1, target 1, lun 0, Disk: (SEAGATE ST15150W        0023)
Bus 1, target 3, lun 0, Changer: (STK     9730         1102)
Bus 1, target 4, lun 0, Tape: (Quantum DLT4000         CD3C)
Bus 1, target 5, lun 0, Tape: (Quantum DLT4000         CD3C)
```

2. Note the bus, target, and lun of the robotic library you want to control as a TLD robot. In the previous example, it is the STK 9730.

3. Create a pseudo device file, as follows:

a. Create a directory in `/dev`.

```
cd /dev
mkdir dir-name
cd dir-name
```

b. Create a file, *file-name*, in this directory that contains the bus, target, and lun for the robotics. The directory name and file name used in the following example is `veritas/stk9730`, but they can be any names.

To configure the STK 9730 robot, create a file as follows. The `lsdev` display in step 1 shows that the bus is 1, the target is 3, and the lun is 0. These three values are entered in the new file.

```
cat > stk9730
1 3 0
^D
```

4. Use `/dev/`*dir-name*`/`*file-name* as the robotic path when using `tldtest` or when configuring the robot. For example

```
tldtest -r /dev/veritas/stk9730
```

Media Manger uses the file to obtain the path to the device required by the passthru capability (bus, target, and lun).

Configuring Tape Drives

The following table shows the drivers that are used with various drive types:

Drive Type	Sequent Driver
Exabyte 8500, 8500C, 8505, 8505XL, 8900	tx
DLT4000, DLT7000	tl
IBM Magstar (3590)	tc
4mm DAT	td
STK 4490, 4781 (4480), 4791 (Silverton), 4890 (Twin Peaks), 9490 (Timberline), SD-3 (Redwood)	tf

See the Sequent DYNIX `man` pages on the tape drivers for information on which device paths to use for a specific drive. The following table shows example device files to use in Media Manager:

Drive Type	No Rewind Device
Exabyte 8500C	/dev/rmt/tx0x85cn
1/2 Cartridge (3480)	/dev/rmt/tf2n
DLT	/dev/rmt/tl4n
IBM Magstar (3590)	/dev/rmt/tc3n
4mm DAT	/dev/rmt/td6n

To configure psuedo-device files for tape drives to use fast positioning (locate block), perform the following steps:

1. The following output from `lsdev`, lists the devices in an example system. `lsdev` uses the passthru capability to do an inquiry command.

```
/usr/openv/volmgr/bin/lsdev
Bus 0, target 0, lun 0, Disk: (IBM OEM DFHSS4E         4343)
Bus 0, target 1, lun 0, Disk: (SEAGATE ST15150W        0023)
Bus 0, target 3, lun 0, Tape: (EXABYTE EXB8500C8SQANXRU07J0)
Bus 0, target 4, lun 0, Tape: (TANDBERG TDC 3800       -07:)
```

```
Bus 1, target 1, lun 0, Disk: (SEAGATE ST15150W        0023)
Bus 1, target 3, lun 0, Changer: (STK     9730          1102)
Bus 1, target 4, lun 0, Tape: (Quantum DLT4000         CD3C)
Bus 1, target 5, lun 0, Tape: (Quantum DLT4000         CD3C)
```

Note the bus, target, and lun of the tape drives you want to configure (for example, the two Quantum DLT4000s).

2. Use the command `/etc/dumpconf` to determine the tape device name by matching the target (in the UNIT) column and the `scsibus`. The following is an excerpt from `dumpconf`:

```
NAME     CFGTYPE  DEVNUM  UNIT        FLAGS  OnBUS    OnDEVICE
tl0      tl          0    0x00000040  S      scsi     scsibus1
tl1      tl          1    0x00000050  S      scsi     scsibus1
        The tape at target 4 is /dev/rmt/tl0.
        The tape at target 5 is /dev/rmt/tl1.
```

3. Create a device file, as follows:

a. Create a `veritas` directory in `/dev` if it does not exist (the name must be `veritas`).

```
cd /dev
mkdir veritas
cd veritas
```

b. Create a file, *file-name*, in `dev/veritas` that contains the bus, target, and lun for each tape drive. *file-name* must be located in this directory and must match the last element of the path of the tape drive that is configured as the non-rewind device name (using the Media and Device management interface, `tpconfig`, or `xdevadm`).

For example to configure the two DLT drives, use the output from the `tpconfig -d` command.

```
Index   DriveName       DrivePath          Type  Shared     Status
*****   *********       **********         ****  ******     ******
  4   /dev/rmt/tl0     /dev/rmt/tl0n       dlt     no        UP
        TLD(0) Definition       DRIVE=1
  5   /dev/rmt/tl1     /dev/rmt/tl1n       dlt     no        UP
        TLD(0) Definition       DRIVE=2

Currently defined robotics are:
    TLD(0)     robotic path = /dev/veritas/stk9730, volume database
                                                          host = hosta
```

Create files for the two DLT drives as follows. The existence of the files `/dev/veritas/tl0n` and `/dev/veritas/tl1n` with the correct bus, target, and lun is all that's needed to enable locate block. The important thing to remember is that the filename must be the same as the `/dev/rmt` filename for the non-rewind device.

```
cat > tl0n
1 4 0
^D

cat > tl1n
1 5 0
^D
```

Kernel Configuration

Media Manager (the `avrd` daemon) periodically attempts to open configured tape drives that are UP to see if a tape has been loaded. DYNIX logs error messages to the console when a not ready (empty) tape drive is opened.

The following are kernel configuration options you can make to reduce the number of messages that are logged. After making changes to any kernel configuration files you must generate a new kernel for the system. See the `config (1M)` man page.

Turning Off Messages

To turn off messages for drives being scanned, change the following line in `/usr/conf/uts/io/scsitape/scsitape_space.c`.

From

```
int sct_devroute = CE_TRACE | CE_WARN;
```

To

```
int sct_devroute = CE_TRACE;
```

Exabyte Drive Type

If you are using 8mm Exabyte tape drives, you may want to disable the 45 second wait for a drive to become ready. Change the following line in `/usr/conf/uts/io/tx/tx_space.c`.

From

```
int tx_ready_timeout = 45;
```

To

```
int tx_ready_timeout = 0;
```

DLT Drive Type

If you are using DLT tape drives, you may want to disable the 45 second wait for a drive to become ready. Change the following line in `/usr/conf/uts/io/tl/tl_space.c`.

From

```
int tl_ready_timeout = 45;
```

To

```
int tl_ready_timeout = 0;
```

Tape Drive Support

DLT Drive Type

The DLT driver from Sequent should be installed. Refer to the SGI installation guide for instructions for this driver.

IBM Magstar (3590) Drive Type

The IBM Magstar driver from Sequent should be installed. Refer to the SGI installation guide for instructions for this driver.

Command Summary

The following commands display the hardware configuration.

`/etc/dumpconf`

Examines the physical devices configured on the system.

The `-d` option shows the SCSI buses and tape devices on the system.

`/etc/showcfg`

Displays the configuration of the system in a manner similar to the power-up monitor configuration command.

The `-s` option selects an alternate one-line format that gives the quantity of each type of board.

The `-d` option produces a dump of relevant parts of the system configuration description table. The data displayed includes information about the memory available, the boot flags, the boot device, console tty control characters, and the current system bus mode.

Glossary

access control list (ACL)

Security information associated with files on some file systems.

ACS

Automated Cartridge System. ACS can refer to any of the following:

- A type of Media Manager robotic control. This robot type is supported only by NetBackup DataCenter servers.
- The StorageTek (STK) system for robotic control.
- The highest-level component under STK's ACS library software, which refers to a specific standalone robotic library or to multiple libraries connected with a media passthru mechanism.

active job

A job for which NetBackup is currently processing backup or restore data.

activity logs

See "debug logs."

activity monitor

A NetBackup administration utility that displays information about NetBackup jobs and provides limited control over them.

administration client

See "remote administration console."

administrator

A user that is granted special privileges to install, configure, and manage the operation of a system, network, or application.

AIT

Sony Advanced Intelligent Tape, a type of tape drive or media type.

alternate-client restore

See “redirected restore (different client).”

alternate-target restore

See “redirected restore (different target).”

alternate path restore

See “redirected restore (different path).”

alternate read server

A server used to read a backup image which was originally written by a different media server. The media server specified as Alternate Read Server must have access to the media containing the backup image or images it is configured to read.

archive

A special kind of backup where NetBackup backs up the selected files, and if the backup is successful, deletes the files from the local disk. In this manual, references to backups also apply to the backup portion of archive operations except where otherwise noted.

archive bit

A file-status bit that the Microsoft based operating system sets when it writes a file, thereby indicating that the file has changed.

attributes for a policy

Configuration parameters that control the behavior of NetBackup during operations involving this policy.

autochanger

See “robotic library.”

autoloader

See “robotic library.”

automatic backup

A scheduled backup by the master server.

back up

The act of copying and saving files and folders to storage media.

backup

Refers to the process of copying and saving files and directories to storage media. For example, *the backup is complete*. This term can also refer to the collection of data that NetBackup saves for a client during a backup or archive. For example, *duplicate the backup*.

Backup is two words when used as a verb. For example, *back up the file*.

backup, archive, and restore interface

The name of the NetBackup Microsoft Windows and Java based user interfaces for clients. On servers these interfaces can be started through the NetBackup Administration Console.

backup window

The period of time during which backups can begin.

block size

The number of bytes in each block of data written on the media during a backup.

bp

A backup, archive, and restore utility for users on NetBackup UNIX clients. It has a character-based, menu interface that can be run from terminals that do not have X Windows capabilities.

bpadm

An administrator utility that runs on NetBackup UNIX servers. It has a character-based, menu interface that can be run from terminals that do not have X Windows capabilities.

bp.conf file

A NetBackup configuration file on UNIX servers and also on UNIX, Macintosh, and OS/2 clients.

bp.ini file

NetBackup initialization file for Novell NetWare target clients.

bpcd

NetBackup Client service on Windows and the NetBackup Client daemon on UNIX.

bprd

NetBackup Request Manager service on Windows and NetBackup Request daemon on UNIX.

cancel a job

Terminating a job and removing it from the job queue.

carousel

See “robotic library.”

catalogs

Internal NetBackup and Media Manager databases. These catalogs contain information about configuration, media, devices, status, errors, and the files and directories in the stored backup images.

CDF

Context-dependent file, which is a type of directory structure on a Hewlett-Packard system.

changer

See “robotic library.”

class

See “policy.”

client

The system with the files to back up, archive, or restore.

client-user interface

See “user interface.”

cluster

See master and media server cluster.

command lines

Commands that users can execute either from the system prompt or in scripts.

compression

The process of compacting data to enable more efficient transmission and storage.

configuration

The parameters that govern the behavior of an application. This term can also refer to the manner in which a network or system is laid out or connected (for example, a network configuration).

consolidated eject

A process of ejecting media for more than one Vault session at a time. A Consolidated Eject can be performed for one or more logical vaults at one time.

consolidated report

A process of generating reports for more than one Vault session at a time. A Consolidated Report can be performed for one or more logical vaults at one time. Consolidated reports are organized by report title, not by vault.

cpio

A UNIX command that can be used for copying files to or from a cpio archive on disk or tape.

ctime

The time that a UNIX inode was changed.

cumulative-incremental backup

A backup that is scheduled by the administrator on the master server and backs up files that have changed since the last successful full backup. All files are backed up if no prior backup has been done. Also see “differential-incremental backup.”

daemon

A program on a UNIX system that runs in the background and performs some task (for example, starting other programs when they are needed). Daemons are generally referred to as services or processes on Windows server systems.

database-agent clients

Clients with additional NetBackup software that is designed to back up relational databases.

database-extension clients

See “database-agent clients.”

debug logs

Logs that can be optionally enabled for specific NetBackup and Media Manager programs and processes and then used to investigate problems.

destination storage unit

A storage unit to which Vault sends the data from a duplication operation. If the duplicated backup images are to be vaulted, then the destination storage unit must correspond to the robotic volume group.

device delays

Delays caused by the device that are beyond the control of the storage application. An example is the time required to position tape under the read and write heads.

device host

A host (that has Media Manager installed) where a drive or robotic control is attached or is defined.

device monitor

A Media Manager administration utility that provides monitoring and manual control of Media Manager storage devices. For example, an administrator or computer room operator can use this utility to manually reset devices or set them to the UP or DOWN state.

DHCP

Dynamic host configuration protocol. This TCP/IP protocol automatically assigns temporary IP addresses to hosts when they connect to the network.

differential-incremental backup

Scheduled by the administrator on the master server and backs up files that have changed since the last successful incremental or full backup. All files are backed up if no prior backup has been done. Also see "cumulative-incremental backup."

directory depth

The number of levels below the current directory level that the NetBackup interfaces show in their directory and file list displays.

directory tree

The hierarchical structure in which files are organized on a disk. Each directory lists the files and directories that are directly below it in the tree. On UNIX, the topmost directory is called the root directory.

disaster recovery

Recovering data from backups after a disk crash or other catastrophe.

disk

Magnetic or optical disk storage media.

disk-image backup

A bit-by-bit rather than a file system backup of a disk drive on a Windows platform.

DLT

Digital-linear tape or tape drive type.

Domain Name Service (DNS)

A program that handles name translation for network communications.

drive cleaning

The use of a special cleaning tape to clean the heads on a drive.

duplicate image

A copy of a backup image.

eject

Move media out of a robotic library.

encryption

Provides additional security by encrypting backup data on the client. This capability is available only with the NetBackup Encryption option.

entry and exit ports

See “media access port.”

exclude list

A list that designates files or directories to exclude from automatic backups.

expiration (image)

The date and time when NetBackup stops tracking a backup image.

expiration (volume)

The date and time when the physical media (tape) is considered to be no longer usable.

external media ID

This is an identifier written on a media cartridge or canister to help the operator identify the volume before inserting it into a drive or robot. For labeled media, the external media ID should be the same as the media ID recorded on the media.

EVSN

See “external media ID.”

FlashBackup

A special type of raw-partition backup that requires the NetBackup FlashBackup separately-priced option (this option is available only for NetBackup DataCenter).

flush level

Controls how often Netbackup clears its log files on a Novell NetWare or Microsoft Windows client platform.

fragment

A part of a backup or archive image. NetBackup can be configured to divide images into fragments when they exceed a certain size or span tapes.

frequency (backup)

How often NetBackup performs scheduled backups. For example, if the frequency is seven days then backups occur once a week.

FROZEN media state

If a volume is FROZEN, NetBackup keeps it indefinitely and can restore from it but not use it for further backups or archives.

full backup

A backup that copies, to a storage unit, all files and directories that are beneath a specified directory.

FULL media state

If this appears in a report or listing, it indicates the volume is FULL and cannot hold more data or be used for further backups.

global attributes

NetBackup configuration attributes that affect all policies.

GDM Dashboard

The name for the Global Data Manager interface. The Dashboard enables monitoring job and drive activity on multiple master servers, as well as providing alerts to problem conditions.

GDM Managed Server

A NetBackup master server that appears as a managed master server in the left pane of the GDM Dashboard.

GDM Server

A NetBackup master server that has the Global Data Manager license activated. When logging into this host, the user can monitor the activity on multiple master servers using the GDM Dashboard interface. If the host has installed the Advanced Reporter option, the reports show information on multiple master servers.

Global Data Manager (GDM)

A separately-priced option (for UNIX servers) that provides an interface with a tree view where the administrator can view and administer multiple master servers. The server where the option is installed is called a GDM Server.

Global Device Database

A single host that serves as the repository for global device configuration information. When you install NetBackup, by default the master server is configured as the global device database host.

GNU tar

A public domain version of the UNIX tar program.

goodies directory

A directory containing programs, scripts, and other files that are not formally supported.

GUI

Graphical user interface.

hard link

On UNIX, a hard link is a pointer to the inode for the data. On a Windows server, a hard link is a directory entry for a file. Every file can be considered to have at least one hard link. On NTFS volumes each file can have multiple hard links, and a single file can appear in many directories (or even in the same directory with different names).

heap level

A parameter for memory-heap debugging on a Novell NetWare or Windows NetBackup client.

hierarchical storage management

The process of automatically migrating selected files from a managed file system to specified migration levels on secondary storage, while maintaining transparent access to those files.

host

A computer that executes application programs.

host name

Name by which a host computer is identified by programs and other computers in the network.

HSM

See storage migrator.

image

The collection of data that NetBackup saves for an individual client during each backup or archive. The image contains all the files, directories, and catalog information associated with the backup or archive.

import

The process of recreating NetBackup records of images so the images can be restored.

include list

A list that designates files or directories to add back in from the exclude list.

incremental backup

See “cumulative-incremental backup” and “differential-incremental backup.”

inject

Move media into a robotic library.

inport

See “media access port.”

inode

A UNIX data structure that defines the existence of a single file.

install_path

Directory where NetBackup and Media Manager software is installed. The default on Windows servers is `C:\Program Files\VERITAS` and on UNIX it is `/usr/openv`.

jbpSA

The Java-based NetBackup interface for performing user backups, archives, and restores.

jnbSA

The Java-based NetBackup interface for administrators.

job

A parcel of work submitted to a computer. NetBackup jobs are backups, archives, or restores.

kernel

The nucleus of an operating system.

keyword phrase

A textual description of a backup.

kill a job

See “cancel a job.”

label

Identifier of a tape or optical disk volume. A recorded label includes a media ID.

A barcode label allows a barcode scanner to be used for media tracking.

library

See “robotic library.”

link

See “hard link” or “symbolic link.”

LMF - Library Management Facility

A Media Manager designation for a category of robot. For the specific vendor types and models in this category, see the VERITAS support web site.

This robot type is supported only by NetBackup DataCenter servers.

load

(noun) Amount of work that is being performed by a system or the level of traffic on a network. For example, network load affects performance.

(verb) Copy data to internal memory. For example, load the installation program.

(verb) Used to indicate tape drive initialization done when new media is being added.

logs

Files where a computer or application records information about its activities.

mailslot

See “media access port.”

man pages

Online documentation provided with UNIX computer systems and applications.

Master and media server cluster

A NetBackup master server and the remote media servers that it is using for additional storage. It is possible to configure clusters only with NetBackup DataCenter servers. NetBackup BusinesServer supports only a single server, the master.

Master of Masters

A NetBackup host where Global Data Manager software is installed. When logging into this host, the interface has a tree view where the administrator can view and administer multiple master servers.

master server

The NetBackup server that provides administration and control for backups and restores for all clients and servers in a master and media server cluster. NetBackup BusinesServer supports only a single server and it is the master.

media

Physical magnetic tapes, optical disks, or magnetic disks where data are stored.

media access port

A slot or other opening in a robot where you can insert or remove a tape without having to access the interior of the robot. After inserting a tape, you move it to a slot by using an inject command. Prior to removing a tape, you move it to the port by using an eject command. The inject and eject commands are supported through the add and move screens in the Media Manager administration interface.

media host

NetBackup server to which the job (client) is sending the data.

media ID

An identifier that is written on a volume as part of the recorded label.

Media Manager

Software that is part of NetBackup and manages the storage devices and removable media.

Media Manager Host

Host where Media Manager is installed (may have devices attached).

media server

A NetBackup server that provides storage within a master and media server cluster. The master can also be a media server. A media server that is not the master is called a remote media server. NetBackup BusinesServer does not support remote media servers.

menu interface

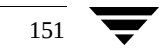
A character-based interface for use on terminals that do not have graphical capabilities.

mount

Make a volume available for reading or writing.

mount point

The point where a file system on a disk logically connects to a system's directory structure so the file system is available to users and applications.



MPX

See "multiplexing."

mtime

The point in time when a UNIX or NTFS file is modified.

multiplexing

The process of sending concurrent-multiple backups from one or more clients to a single storage device and interleaving those images onto the media.

multiplexed group

A set of backups that were multiplexed together in a single multiplexing session.

NDMP

Network data management protocol. NetBackup requires the NetBackup for NDMP separately-priced option to support NDMP.

NetBackup Client service

NetBackup Windows service that runs on clients and servers and listens for connections from NetBackup servers and clients in the network. When a connection is made, this service starts the necessary programs.

NetBackup configuration options

On UNIX servers and on UNIX and Macintosh, clients, these settings are made in the `bp.conf` file. On NetWare target and OS/2 clients, they are in the `bp.ini` file. On Windows servers and Windows clients, these settings are called properties and are made through the Backup, Archive, and Restore interface or the Host Properties dialog in the NetBackup Administration Console.

NetBackup databases

See catalogs.

NetBackup Database Manager service

NetBackup Windows service that runs on the master server and manages the NetBackup internal databases (called catalogs). This service must be running on the master server during all NetBackup administrative operations.

NetBackup Device Manager service

The NetBackup Windows service that runs on a NetBackup server and starts the robotic control processes and controls the reservation and assignment of volumes. This service runs only if the server has devices under Media Manager control. The process is `ltid`.

NetBackup properties

Same as NetBackup configuration options but are called NetBackup properties on Microsoft Windows platforms.

NetBackup Request Manager service

The NetBackup Windows service that runs on the master server and starts the scheduler and receives requests from clients.

NetBackup Volume Manager service

A NetBackup Windows service that runs on a NetBackup server, allows remote administration of Media Manager, and manages volume information. The process is `vmd`.

NIS

Network information service.

NLM

NetWare loadable module.

NFS

Network file system.

nonrobotic

See “standalone.”

ODL

Optical disk library. This robot type is supported only by NetBackup DataCenter servers.

offsite volume group

A volume group in which media will appear after having been ejected from the robot for vaulting. When Vault ejects media it is moved from the robotic volume group to the off-site volume group.

offsite volume pool

A volume pool that contains media that is to be ejected and vaulted. Backup images written to an off-site volume pool by an original NetBackup backup policy or by Vault's duplication feature will be ejected and vaulted. More than one off-site volume pool can be specified for the Eject step of a Vault profile.

original backup

A backup image created by a backup job. A single backup image or all backup images created by an Inline Tape Copy (multiple copy) configuration are considered original backups. A backup image created by a duplication job is not an original backup.

outport

See "media access port."

partitions

The logical partitions into which a magnetic disk is divided.

patch

A program that corrects a problem or adds a feature to an existing release of software.

path length

Number of characters in a pathname.

pathname

The list of directories in the path to a destination directory or file.

PC clients

NetBackup clients that have Microsoft Windows, Macintosh, or IBM OS/2 operating systems.

peername

The name by which a computer identifies itself when establishing connections to other systems.

policy

Defines the backup characteristics for a group of one or more clients that have similar backup requirements.

port

A location used for transferring data in or out of a computer.

Also see “media access port.”

primary copy

The copy of an image that NetBackup uses to satisfy restores. When NetBackup duplicates an image, the original is designated as the primary copy.

privileges

The tasks or functions that a user, system, or application is authorized to perform.

profile

A vault profile is a way to save configuration settings. Specific parameters for duplication, catalog backup, eject, and report or any combination of these steps, are configured within a profile.

progress report

Log where NetBackup records events that occur during user operations.

proxy restore

A proxy restore allows the user to restore files that he has write access to, on a machine other than his desktop. The files must be in a backup of the machine to which they are being restored.

QIC

Quarter-inch-cartridge tape.

queued job

A job that has been added to the list of jobs to be performed.

raw-partition backup

Bit-by-bit backup of a partition of a disk drive on UNIX. On Windows, this is called a disk-image backup.

rbak

The program that Apollo clients use to read data from tape during a restore.

recorded media ID

This is an identifier written as part of the label on a volume and used by Media Manager to ensure that the correct volume is mounted. The recorded media ID should match the external media ID.

redirected restore (different client)

Restoring files to your client when they were originally backed up from a different client. The administrator using the interface on the master server can direct a restore to any client (this variation is called a server directed restore).

redirected restore (different target)

On a Novell NetWare server platform running the NetBackup target version of client software, this operation restores files to a different target than the one from which they were backed up.

redirected restore (different path)

Restores files to a different directory than the one from which they were backed up.

registry

A Microsoft Windows database that has configuration information about hardware and user accounts.

remote administration console

A Windows NetBackup client that has the administration interface software installed and can be used to administer NetBackup servers.

remote media server

A media server that is not the master. Note that only NetBackup DataCenter supports remote media servers. NetBackup BusinesServer supports only a single server, the master.

residence

In Media Manager, information about the location of each volume is stored in a volume database. This residence entry contains information, such as robot number, robot host, robot type, and media type.

resource

A Novell NetWare term that refers to a data set on the target. For example, in DOS, resources are drives, directories, and files. Also see “target service.”

restore

(verb) The act of restoring selected files and directories from a previous backup or archive and returning them to their original directory locations (or to a different directory).

(noun) The process of restoring selected files and directories from a previous backup and returning them to their original directory locations (or to a different directory).

retention level

An index number that corresponds to a user-defined retention period. There are 10 levels from which to choose (0 though 9) and the retention period associated with each is configurable. Also see “retention period.”

retention period

The length of time that NetBackup keeps backup and archive images. The retention period is specified on the schedule.

robotic arm

The component of a robotic library that physically selects the media (tape or optical disk).

robotic library

Refers to a robot and its accompanying software. A robotic library includes a collection of tapes or optical platters used for data storage and retrieval. For example, a Tape Library DLT (TLD) refers to a robot that has TLD robotic control.

robotic volume group

A volume group from which media will be ejected and vaulted. When Vault duplicates backups, they are duplicated to media in the robotic volume group.

root

The highest level directory in a hierarchical directory structure. In MS-DOS, the root directory on a drive is designated by a backslash (for example, the root on drive C is C:\). On UNIX, the root directory is designated by a slash (/).

Also, a UNIX user name having administration capability.

RS-232

An industry-standard interface for serial communications and sometimes used for communicating with storage peripherals.

RSM Interface

Application in Windows 2000 used to manage Removable Storage Manager (RSM) devices.

RSM - Removable Storage Manager

A Media Manager designation for a category of robot. For the specific vendor types and models in this category, see the VERITAS support web site.

Also, a component of the Windows 2000 operating system that manages storage devices.

RVSN

See “recorded media ID.”

schedules

Controls when backups can occur in addition to other aspects of the backup, such as: the type of backup (full, incremental) and how long NetBackup retains the image.

SCSI

Small computer system interface. This is a type of parallel interface that is frequently used for communicating with storage peripherals.

server-directed restore

Using the user interface on the master server to restore files to any client. Only the administrator can perform this operation.

server independent restore

Restoring files by using a NetBackup server other than the one that was used to write the backup. This feature is available only with NetBackup DataCenter.

server list

The list of servers that a NetBackup client or server refers to when establishing or verifying connections to NetBackup servers. On a Windows server and Microsoft Windows clients, you update the list through a dialog box in the interface. On a UNIX server and UNIX and Macintosh clients, the list is in the `bp.conf` file. On NetWare target and OS/2 clients, the list is in the `bp.ini` file.

service

A program on a Windows server system that runs in the background and performs some task (for example, starting other programs when they are needed). Services are generally referred to as daemons on UNIX systems.

session

An instance of NetBackup checking its schedules for backups that are due, adding them to its worklist, and attempting to complete all jobs in the worklist. For user backups and archives, a session usually consists of a single backup or archive.

Session (Vault)

A vault session consists of executing a particular profile or profiles.

shared drives

See “Shared Storage Option (SSO).”

Shared Storage Option (SSO)

A separately priced VERITAS software option that allows tape drives (standalone or in a robotic library) to be dynamically shared among multiple NetBackup and Storage Migrator servers.

This option is supported only on NetBackup DataCenter servers.

SMDR

Storage management data requestor, a Novell NetWare program that provides its services transparently to all SMS modules and lets remote and local modules communicate with one another.

SMS

Novell NetWare storage management services.

source volume group

A volume group from which Vault can select backups to duplicate. This parameter is used to restrict the list of backups from all backups that reside on media in any volume group to backups that reside on media in a single volume group. Where a volume group corresponds to a particular robot, the profile will duplicate only backups on media in that robot. The Source Volume Group is normally only specified if you have multiple robots attached to the same server, for example you want to duplicate backups that reside in robot 0 to media that reside in robot 1.

SSO

See “Shared Storage Option (SSO).”

stacker

Usually a small robotic library that contains one drive only. See “robotic library.”

standalone

A qualifier used with drives and media to indicate they are not associated with a robot. For example, a standalone tape drive is one where you must manually find and insert tapes before using them. A standalone volume is one that is located in a standalone drive or is stored outside of a drive and designated as standalone in the volume configuration.

status code

A numerical code, usually accompanied by a troubleshooting message, that indicates the outcome of an operation.

storage migrator

Refers to the VERITAS Storage Migrator line of hierarchical storage management products for UNIX and Windows. These products make extra room on a disk by transparently moving data to other storage and then transparently retrieving the data when it is needed by a user or application.

Storage Migrator is available only for NetBackup DataCenter servers.

storage unit

Refers to a storage device where NetBackup or Storage Migrator stores files. It can be a set of drives in a robot or consist of one or more single tape drives that connect to the same host.

SUSPENDED media state

If a volume is SUSPENDED, NetBackup can restore from it but cannot use it for backups. NetBackup retains a record of the media ID until the last backup image on the volume expires.

symbolic link

On a UNIX system, this is a pointer to the name of the file that has the source data.

TapeAlert

Allows reactive cleaning for most drive types and is a function of the tape drive.

tape format

The format that an application uses to write data on a tape.

tape marks

A mark that is recorded between backup images on a tape.

tape overhead

The space required for data that is not part of the backup images. For example, tape marks and catalogs of what are on the tape are considered overhead.

tape spanning

Using more than one tape to store a single backup image.

tar

Tape Archive program that NetBackup uses to extract backup images during a restore.

target

See “target service.”

target service

A Novell NetWare service that needs storage management. The SMS views all services (for example, print services, communication services, workstations) as targets.

Target Service Agent

A Target-service agent is a Novell NetWare agent that prepares the target's data for SMS during a backup and for the target during a restore.

TLD - Tape Library DLT

A Media Manager designation for a category of robot. For the specific vendor types and models in this category, see the VERITAS support web site.

TLH - Tape Library Half-inch

A Media Manager designation for a category of robot. For the specific vendor types and models in this category, see the VERITAS support web site.

This robot type is supported only by NetBackup DataCenter servers.

TLM - Tape Library Multimedia

A Media Manager designation for a category of robot. For the specific vendor types and models in this category, see the VERITAS support web site.

This robot type is supported only by NetBackup DataCenter servers.

TL4 - Tape Library 4MM

A Media Manager designation for a category of robot. For the specific vendor types and models in this category, see the VERITAS support web site.

TL8 - Tape Library 8MM

A Media Manager designation for a category of robot. For the specific vendor types and models in this category, see the VERITAS support web site.

timeout period

The period of time that an application has allotted for an event to occur.

TIR

See “true image restore.”

tpconfig

A Media Manager administration utility for configuring devices which is started from the command line. On UNIX, it has a character-based menu interface that can be run from terminals that do not have X Windows capabilities. tpconfig also has a command line interface.

transfer rate

The rate at which computer information is transferred between a source and a destination.

transport

See “robotic arm.”

true image restore

Restores the contents of a directory to what it was at the time of any scheduled full or incremental backup. Previously deleted files are ignored.

TS8 - Tape Stacker 8MM

A Media Manager designation for a category of robot. For the specific vendor types and models in this category, see the VERITAS support web site.

TSA

See “Target Service Agent.”

TSD - Tape Stacker DLT

A Media Manager designation for a category of robot. For the specific vendor types and models in this category, see the VERITAS support web site.

TSH - Tape Stacker Half-inch

A Media Manager designation for a category of robot. For the specific vendor types and models in this category, see the VERITAS support web site.

This robot type is supported only by NetBackup DataCenter servers.

unassigned media

Media that contain no valid images. A piece of unassigned media has an entry in the volumes database but no entries in the images database. Unassigned Media do not have a “time assigned” in the Media section of the GUI.

user interface

The program used to perform user backups, archives, and restores.

user operation

A backup, archive, or restore that is started by a person on a client system.

Vault

Vault is a separately-priced NetBackup option that provides offsite backup management. Vault automatically duplicates specified backup images, and automates the process of offsite media rotation (a critical component of any backup or disaster recovery strategy). Vault manages offsite storage and retrieval of media for original backups, duplicate backups, and catalog backups. Additionally, NetBackup Vault generates reports to track the location and content of each piece of media.

vault

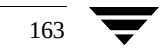
In the context of the NetBackup Vault, a vault is logical entity associated with a particular robot that acts as a designated holding place for backups that will eventually be sent to a physical offsite vault. The term ‘vault’ is used to refer both to the process, and to the physical storage location of a set of tapes offsite.

vault process

Vaulting is the process of choosing backup images to duplicate or eject, optionally duplicating backups, ejecting duplicate or original media, storing it at an offsite location, and later returning expired media to your robot. Vaulting is an integral part of the disaster recovery process.

verbose flag

Configuration file entry that causes a higher level of detail to be written in the logs.



verify

An operation that compares the list of files that are actually on a volume with what NetBackup has recorded as being on it. The data that is on the media is not verified.

vmadm

A Media Manager administrator utility for managing volumes. It runs on UNIX and has a character-based, menu interface that can be run from terminals.

vm.conf

A Media Manager configuration file with entries that include the servers that can manage local devices and default media ID prefixes for media that do not contain barcodes.

volume

Media Manager volumes are logical units of data storage or cleaning capability on media that have been assigned media IDs and other attributes, which are recorded in the Media Manager volume database.

volume configuration

Refers to configuration information that is stored in the Media Manager volume database.

volume database

An internal database where Media Manager keeps information about volumes. All hosts (where Media Manager is installed) have a volume database. However, the database is empty unless the host is designated as a volume database host.

volume database host

The host (where Media Manager is installed) that contains information about the volumes that Media Manager uses in a device. Because NetBackup BusinesServer supports only a single server, the volume database host is always on the same server.

volume group

A set of volumes that are configured within Media Manager to reside at the same physical location (for example, in a specific robot).

volume pool

A set of volumes that are configured within Media Manager to be used by a single application and are protected from access by other applications and users.

wakeup interval

The time interval at which NetBackup checks for backups that are due.

wildcard characters

A character that can be used to represent other characters in searches.

Microsoft Windows

(noun) Describes a line of operating systems developed by Microsoft, Inc.

For more information on the Windows operating systems that NetBackup supports, refer to the VERITAS support web site at `http://www.support.veritas.com`.

Windows

(adjective) Used to describe a specific product or clarify a term. Some examples are: Windows 95, Windows 98, Windows NT, Windows 2000, Windows servers, Windows clients, Windows platforms, Windows hosts, and Windows GUI.

Windows servers

A term that defines the Windows server platforms that NetBackup supports; those platforms are: Windows NT and 2000.

Windows clients

A term that defines the Windows client platforms that NetBackup supports; those platforms are: Windows 95, 98, ME, NT, 2000, XP (for 32- and 64-bit versions), and LE.

Windows Display Console

A NetBackup-Java interface program that runs on Windows 2000, NT, 98, and 95 computers. Users can start this interface on their local system, connect to a UNIX system that has the NetBackup-Java software installed, and then perform any user operations that their permissions allow.

WORM media

Write-once, read-many media for optical disks. NetBackup BusinesServer does not support WORM media.

xbp

The X Windows-based backup, archive, and restore program for users on NetBackup UNIX clients.

Index

T