Installing a DLT, SDLT, VS, LTO, or DAT Tape Drive Into a Linux Operating System

Purpose

This document provides instructions for integrating a Quantum® DLTtape® drive, a Super DLTtape™ drive, a DLT VS tape drive, an LTO tape drive, or a DAT tape drive into Red Hat® Linux, Novell® SuSE® Linux, or other Linux operating system.

Scope

This document is intended for users who have a general understanding of Linux operating systems.

The instructions and examples provided in this document refer specifically to Red Hat and Novell SuSE Linux systems. The instructions may differ slightly if you are running a Linux system other than Red Hat or Novell SuSE. If these instructions are not adequate, refer to the Linux user guide for your system.

This document pertains to the following products:

- **Tape drives:** DLT 2000, DLT 2000XT, DLT 4000, DLT 7000, DLT 8000, SDLT 220, SDLT 320, SDLT 600, DLT S4, DLT1, VS80, VS160, DLT-V4, LTO-1, LTO-2, LTO-3, and DAT 72.

> **Note:** The Super DLT1 tape drive is currently known as the SDLT 220 tape drive.

- **Mini-libraries:** DLT 2500, DLT 2500XT, DLT 2700, DLT 2700XT, DLT 4500, and DLT 4700.

Conventions Used in This Guide

This document uses the following conventions when providing examples of **st** commands and codes:

Item	Example	Meaning
"n" in parentheses in the **st** file name	(n)st0*	The example applies either to auto-rewind or to no-rewind devices.
"n" absent from the **st** file name	st0*	The device is auto-rewind.
"n" present in the **st** file name	nst0*	The device is no-rewind.

Item	Example	Meaning
"0" in an **st** file name	st0	Represents the device number. In actual practice, this numeric character may be any number from 0 to 31 depending on the node. See Identifying Device Nodes on page 14 for more information.
* (asterisk)	mt-st*	The asterisk is a placeholder representing the rest of the file name.

Reference Documents

As you install and work with your Linux system, you should have the manual for your tape drive available to refer to. Following is a list of all the product manuals for Quantum tape drives.

Product Manual Title	Document Number
DLT 2000/DLT 2500/DLT 2700 Cartridge Tape Subsystem Product Manual	81-109132-03
DLT 2000XT/DLT 2500XT/DLT 2700XT Product Manual	81-109253-03
DLT 4000 Product Manual	81-60043-04
DLT 7000 Product Manual	81-60000-06
DLT 8000 Product Manual	81-60118-06
SDLT 220 and SDLT 320 Product Manual	81-85002-01
SDLT 600 Product Manual	81-81184-03
DLT1 Tape Drive Installation and Operations Guide	000826-01
VS80 Tape Drive Installation and Operations Guide	001596-01
VS160 Tape Drive Installation and Operations Guide	81-81191-01
DLT-V4 Product Manual	81-81422-023
LTO-1/LTO-2 Tape Drive User's Guide	50001007
LTO-3 Tape Drive User's Guide	50002764
DDS-4/DAT 72 User's Guide	50000711

If you do not have the correct product manual, you can view and download it from the Quantum Web site at:

http://www.quantum.com/ServiceandSupport/SoftwareandDocumentationDownloads/Index.aspx

Setting Up Communication with the Tape Drives

This section describes how to install the tape drive and how to configure the Red Hat Linux and Novell SuSE Linux operating systems to recognize and communicate with the tape drive. The Red Hat or Novell SuSE operating system includes a driver to communicate efficiently with SCSI devices, such as a DLT or SDLT tape drive.

Before Installing the Tape Drive

Before you install the tape drive, follow these steps:

1. If you have a DLT 2000, DLT 2500, or DLT 2700 product, make sure it uses version V10 (or higher) controller firmware. If it does not, go to http://www.quantum.com/am/service_support/downloads/default.htm to download V10.

   (The suggested method to determine what version of firmware you are running is to view the SCSI HBA Bios at boot-up.)

   Note: The DLT 2000XT, DLT 2500XT, DLT 2700XT, DLT 4000, DLT 7000, DLT 8000, SDLT 220, SDLT 320, SDLT 600, DLT S4, DLT1, VS80, VS160, DLT-V4, LTO-1, LTO-2, LTO-3, and DAT 72 tape drives do not have a firmware revision restriction.

2. Make sure that you have the appropriate SCSI interface and cable for your tape drive:

If you have this type of tape drive SCSI connection...	You need this type of SCSI interface...
Single-ended (SE)	SE or LVD
Low-voltage differential (LVD)	LVD
High-voltage differential (HVD)	HVD

3. Obtain the appropriate manual for your tape drive. The product manual provides detailed hardware installation instructions, including switch and jumper settings and information about SCSI bus termination. See Reference Documents on page 6.

Installing the Tape Drive

To install the tape drive, follow these steps:

1. Shut down your workstation or server and remove AC power from the system.
2. Follow the instructions in your tape drive's product manual to install the tape drive and set the SCSI ID.

Verifying the st Module is Loaded

st is the tape device driver for Linux. Typically, **st** is loaded into the kernel as a module to support SCSI tape devices. You must verify that **st** is loaded to assure that the kernel supports SCSI tape devices.

To verify the **st** module is loaded, follow these steps:

1. Execute the following command as superuser:

   # modinfo st

   If **st** is loaded, the output will look similar to one of the following:

   - Red Hat Enterprise Linux 3.0 (RHEL3):

     ```
     filename:    /lib/modules/2.4.21-4.ELsmp/kernel/drivers/scsi/st.odescription:
     "SCSI Tape Driver"
     author:     "Kai Makisara"
     license:    "GPL"
     parm:       buffer_kbs int, description "Default driver buffer size (KB; 32)"
     parm:        max_buffers int, description "Maximum number of buffer allocated
                  at initialisation (4)"
     parm:       max_sg_segs int, description "Maximum number of scatter/gather
                  segments to use (32)"
     parm:      blocking_open int, description "Block in open if not ready an no
                  O_NONBLOCK (0)"
     ```

   - Novell SuSE Linux 9 (SuSE9):

     ```
     parm:          try_wdio:Try direct write i/o when possible
     parm:          try_rdio:Try direct read i/o when possible
     parm:          try_direct_io:Try direct I/O between user buffer and tape drive (1)
     parm:          max_sg_segs:Maximum number of scatter/gather segments to
                      use (256)
     parm:         buffer_kbs:Default driver buffer size for fixed block mode (KB; 32)
     license:       GPL
     description:    SCSI Tape Driver
     author:        Kai Makisara
     depends:        scsi_mod
     supported:      yes
     vermagic:       2.6.5-7.79-smp SMP 586 REGPARM gcc-3.3
     ```

   If the **st** module is not loaded, modinfo will report that the module name is not found. You need to load the **st** module by using the #insmod command. Consult your Linux manuals for instructions.

2. Reboot the server. This allows the **st** driver to attach tape device nodes (/dev/st#).

Verifying Communication with the Tape Drives

You must ensure the **st** driver sees the all the tape devices. If you have added a tape device, you must verify that the **st** driver sees the new device.

You do this by displaying and reviewing the kernel initialization information which contains **st** driver initialization and attachment of SCSI tape devices.

Displaying the Kernel Initialization Information

You can display the kernel initialization information by using any of the following three methods:

Method	Instructions
View **st** information during boot-up.	At boot-up, Linux displays kernel initialization information, including the **st** driver initialization and attachment of the SCSI tape devices. The information scrolls by quickly; if you miss it, try one of the other two methods.
Read the Kernel Message Buffer Log.	The kernel message buffer log contains the most recent kernel logs. Look in /var/log/dmesg. **Remember:** The kernel message buffer is limited in size; therefore, when the buffer becomes full, old logs are discarded.
Execute the dmesg command to view the Kernel Message Buffer Log.	Executing the **dmesg** command is another way to open the kernel message buffer log. Execute the following command: # dmesg | less See "Read the Kernel Message Buffer Log" above for more information about the log.

Reviewing the Kernel Initialization Information

All three of the methods show you the same information. The information looks similar to one of the following:

- RHEL3 output looks similar to the following:

```
Attached scsi tape st0 at scsi2, channel 0, id 4, lun 0
st0: Block limits 4 - 16777212 bytes.
st: Version 20030406, bufsize 32768, max init. bufs 4, s/g segs 16
```

- SuSE9 output looks similar to the following:

```
Attached scsi tape st0 at scsi0, channel 0, id 3, lun 0
st0: try direct i/o: yes (alignment 512 B), max page reachable by HBA 1048575
st: Version 20040318, fixed bufsize 32768, s/g segs 256
```

Identifying the Tape Devices

You can identify the tape device by looking at the values listed for "channel" (PCI Bus), "id" (SCSI ID), and "lun" (lun is always "0" for standalone configurations).

If you have more than one tape device, you should see similar lines of output for each device.

What To Do if Device Information is Missing

If you do not see the information for every attached tape device, then the **st** driver is not communicating with the missing device(s). Try the following solutions:

- Verify that the connector cable length does not exceed the specifications listed in your product manual.
- Make sure the SCSI bus is terminated properly (see your product manual for instructions).
- Ensure there are enough **st** tape device nodes for all your attached tape devices. If not, you will need to create more. See Creating Device Nodes on page 15 and Creating No-Rewind Device Nodes on page 16.

Obtaining Device Information Using /proc/scsi/scsi

To obtain information about a specific tape drive, view the /proc/scsi/scsi file.

What is the /proc File System?

The /proc file system is a map to the running kernel process. It displays a list of connected SCSI devices. It is not a disk-based file system and is dynamic to reflect the current boot-up information.

Viewing /proc Files

The recommended method to view /proc files is to use the cat command piped (|) with command more or less. Execute the following command:

```
# cat /proc/scsi/scsi | less
```

> **Caution:** Do NOT execute the cat command on the /proc/kcore file. This unique file contains a running image of the kernel's memory at that particular moment. Executing the cat command on this file will render your terminal unusable.

The output will look something like the following:

```
Attached devices:
Host: scsi0 Channel: 00 Id: 03 Lun: 00
  Vendor: QUANTUM  Model: SDLT320          Rev: 5252
  Type:   Sequential-Access                ANSI SCSI revision: 02
Host: scsi1 Channel: 00 Id: 03 Lun: 00
```

```
  Vendor: SEAGATE  Model: ST336607LC       Rev: DS04
  Type:   Direct-Access                    ANSI SCSI revision: 03
Host: scsi1 Channel: 00 Id: 06 Lun: 00
  Vendor: DELL     Model: 1x6 U2W SCSI BP  Rev: 5.39
  Type:   Processor                        ANSI SCSI revision: 02
Host: scsi2 Channel: 00 Id: 04 Lun: 00
  Vendor: NEC      Model: CD-ROM DRIVE:466 Rev: 1.06
  Type:   CD-ROM                           ANSI SCSI revision: 02
Host: scsi3 Channel: 00 Id: 04 Lun: 00
  Vendor: QUANTUM  Model: SDLT600          Rev: 1A1A
  Type:   Sequential-Access                ANSI SCSI revision: 04
```

Accessing Devices Using /dev/st* Device Nodes

You can access most devices by using a corresponding special device file stored in the /dev directory.

Each tape device corresponds to eight device nodes (four auto-rewind nodes and four no-rewind nodes).

Generating a List of Device Files

You can generate a list of the /dev directory device file names by executing the following commands:

For this type of device...	Execute this command...
Auto-rewind	#ls -ld /dev/st0*
No-rewind	#ls -ld /dev/nst0*

Reviewing the List of Device Files

The device files display in blocks of four listings (one for each mode). Each listing contains the device node file name and file attributes.

You should see a block of four listings for each device. The list will look similar to the following:

For this type of device...	The list of /dev device files will look similar to...
Auto-rewind	crw-rw---- 1 root disk 9, 0 Sep 15 2003 st0 crw-rw---- 1 root disk 9, 96 Sep 15 2003 st0a crw-rw---- 1 root disk 9, 32 Sep 15 2003 st0l crw-rw---- 1 root disk 9, 64 Sep 15 2003 st0m
No-rewind	crw-rw---- 1 root disk 9, 128 Sep 15 2003 nst0 crw-rw---- 1 root disk 9, 224 Sep 15 2003 nst0a crw-rw---- 1 root disk 9, 160 Sep 15 2003 nst0l crw-rw---- 1 root disk 9, 192 Sep 15 2003 nst0m

If one or more device node listings are not present, see What To Do if a Device Node is Missing on page 12.

For an explanation of what each part of the listing means, see Interpreting the Device Node File Listing on page 13.

What To Do if a Device Node is Missing

If any of the **st** device node listings are not present in the /dev directory, you need to create them.

Use the mknod commands described in Creating Auto-Rewind Device Nodes on page 16 and Creating No-Rewind Device Nodes on page 16.

Refer to the following documentation for detailed instructions:

- Your Linux documentation
- st(4) man page
- mknod(1) man page

Interpreting the Device Node File Listing

The following table explains each part of the displayed device node listing shown in Reviewing the List of Device Files on page 11.

Column	Text Format	Description
First column	crw-rw----	These ten characters describe access type and permissions. **The first character** indicates type of access device as follows: • "c" signifies that (n)st0* is a character (sequential access) device. • "b" signifies that (n)st0* is a block (random access) device. **The next nine characters** indicate permissions for: • the owner (characters 1 - 3) • the group (characters 4 - 6) • global users (characters 7 - 9) **Permissions are defined as follows:** • The first character of each set identifies read permissions. An "r" indicates read permission is granted; a hyphen indicates read permission is denied. • The second character of each set identifies write permissions. A "w" indicates write permission is granted; a hyphen indicates write permission is denied. • The third character of each set identifies execute permissions. An "x" indicates execute permission is granted; a hyphen indicates execute permission is denied. In the example shown, the device is sequential access. The owner and group both have read/write permission but do not have execute permission; global users have no permissions.
Second column	1	Numeric character, not applicable.
Third column	root	Identifies the owner of the device nodes. This is always root (root is the name of the superuser account).
Fourth column	disk	Identifies the group associated with these device nodes. This is always disk.
Fifth column	9, *n*	Identifies the major and minor numbers for that node. **Major Number:** The first number is the major number. The major number indexes a particular device driver in the kernel. To utilize the **st** module, the major number is always 9. **Minor Number:** The second number is the minor number. The minor number serves as a device driver parameter defining various characteristics such as compression, block size, and density. For a complete definition of all the major and minor numbers, see device.txt at /usr/src/linux-<kernel revision>/Documentation/device.txt.

Column	Text Format	Description
Sixth column	Mmm DD YYYY	Date. Not applicable.
Seventh column, section heading	(n)st***na*** ***n*** represents a numeric character; ***a*** represents an alpha character. In the example: (n)st0*	Device Node Description. This is also the actual file name. **(n)st** identifies the device node as either auto-rewind or no-rewind as follows: • **st** identifies the device node as auto-rewind. See Creating Auto-Rewind Device Nodes on page 16 for more information. • **nst** identifies the device node as no-rewind. See Creating No-Rewind Device Nodes on page 16 for more information. **n (the numeric character)** identifies the tape device. "0" identifies the first device; "1" identifies the second device; "2" identifies the third device, and so forth. See Identifying Device Nodes for more information. **a (the alpha character)** identifies the mode. (No alpha character means mode 1.) See Identifying Device Modes for a list of modes.

Identifying Device Nodes

Linux supports up to 32 tape devices [(n)st0* through (n)st31*). Device nodes are numbered consecutively beginning with 0.

Each operating system presets a certain number of device nodes as follows. You may add more manually if needed (up to 32 total). The following table shows two examples:

Operating System	Preset Device Nodes
RHEL3	(n)st0* through (n)st31*
SuSE9	(n)st0* through (n)st7*

Identifying Device Modes

Each node has four modes. Both the auto-rewind and no-rewind functions use the same modes.

Mode	Auto-Rewind Node (example)[a]	No-Rewind Node (example)	Alpha Character Identification
1	st0	nst0	no character
2	st0l	nst0l	lowercase l
3	st0m	nst0m	lowercase m
4	st0a	nst0a	lowercase a

a. The "0" in this table is an example representing the device. In actual practice, this numeric character may be any number from 0 to 31 depending on the node.

Creating Device Nodes

This section shows each **stinit** mode and its corresponding device node identification parameters.

The next two sections — Creating Auto-Rewind Device Nodes and Creating No-Rewind Device Nodes — show examples of the codes you use to create device nodes.

The following table shows the numbering you use to identify the first tape device in each mode. To identify a second tape device, increment the tape device number and minor number by one, and so on for each successive tape device. You can have up to 32 tape devices per mode.

Mode	Tape Device Number for first tape device (increment by one for each successive device)	Alpha Character Mode Identifier	Minor Number of first tape device (increment by one for each successive device)
Mode 1 (Auto-Rewind)	0	none	0
Mode 2 (Auto-Rewind)	0	lowercase l	32
Mode 3 (Auto-Rewind)	0	lowercase m	64
Mode 4 (Auto-Rewind)	0	lowercase a	96
Mode 1 (No-Rewind)	0	none	128
Mode 2 (No-Rewind)	0	lowercase l	160
Mode 3 (No-Rewind)	0	lowercase m	192
Mode 4 (No-Rewind)	0	lowercase a	224

Creating Auto-Rewind Device Nodes

Use the mknod commands as shown in the following table to create auto-rewind device nodes. The table shows only the first two tape devices. You can have up to 32 tape devices per mode. For instructions on numbering more than two, see Creating Device Nodes.

Mode	SCSI Tape Device	Command
1	First	# mknod -m 666 /dev/st**0** c 9 **0**
1	Second	# mknod -m 666 /dev/st**1** c 9 **1**
2	First	# mknod -m 666 /dev/st**0l** c 9 **32**
2	Second	# mknod -m 666 /dev/st**1l** c 9 **33**
3	First	# mknod -m 666 /dev/st**0m** c 9 **64**
3	Second	# mknod -m 666 /dev/st**1m** c 9 **65**
4	First	# mknod -m 666 /dev/st**0a** c 9 **96**
4	Second	# mknod -m 666 /dev/st**1a** c 9 **97**

Creating No-Rewind Device Nodes

Use the mknod command as shown in the following table to create no-rewind device nodes. The table shows only the first two tape devices. You can have up to 32 tape devices per mode. For instructions on numbering more than two, see Creating Device Nodes.

Mode	SCSI Tape Device	Command
1	First	# mknod -m 666 /dev/nst**0** c 9 **128**
1	Second	# mknod -m 666 /dev/nst**1** c 9 **129**
2	First	# mknod -m 666 /dev/nst**0l** c 9 **160**
2	Second	# mknod -m 666 /dev/nst**1l** c 9 **161**
3	First	# mknod -m 666 /dev/nst**0m** c 9 **192**
3	Second	# mknod -m 666 /dev/nst**1m** c 9 **193**
4	First	# mknod -m 666 /dev/nst**0a** c 9 **224**
4	Second	# mknod -m 666 /dev/nst**1a** c 9 **225**

mt-st Linux RPM Package

The **mt-st** Linux Red Hat Package Manager (RPM) package consists of the following tools:

- **mt** tape device manager (see mtx Tape Library Tool on page 20)
- **stinit** tape configuration utility (see mt Tape Device Tool)

In order to use the package, you must ensure that **mt-st** is installed. See the following chapters:

- Verifying mt-st is Installed and
- Installing mt-st

Verifying mt-st is Installed

To perform the actions described from this point forward in this guide, you must ensure **mt-st** is installed.

To check whether **mt-st** is installed, execute the following command:

```
# rpm -qa | grep mt-st
```

This command returns the name of the package followed by a version number (for example, RHEL3 returns mt-st-0.7-11).

If no information is returned, **mt-st** is not installed on your system. To install **mt-st**, see Installing mt-st.

Installing mt-st

mt-st is usually included on your Linux installation CD-ROM.

To install **mt-st**, execute the following command:

```
# rpm -ivh mt-st*
```

where the asterisk represents the remaining portion of the file name (in this case, it is the version number).

mt Tape Device Tool

mt is a tape device management tool that enables you to set **st** driver flags, position loaded media, and secure-erase media.

If **mt-st** is installed, then **mt** is available for use. See Verifying mt-st is Installed for instructions on checking whether **mt-st** is installed.

stinit and stinit.def

stinit automatically initializes SCSI tape drive modes at system startup or reboot by sending ioctl commands to the drive. The commands are defined in the stinit.def definitions text file. The text file is indexed using the inquiry data returned by the drive (manufacturer, device, and revision). See Tape Device and Parameter Definitions for more information about the stinit.def file.

After a new installation of the Linux operating system or a new installation of **mt-st**, an stinit.def file may not exist. You can create an stinit.def file by using the following sample file:

```
/usr/share/doc/mt-st-<version>/stinit.def.examples
```

If you modify stinit.def, you can re-initialize the SCSI tape drive modes by rebooting the server or executing the following command:

```
# stinit or # stinit -f <pathname>/stinit.def
```

where *pathname* is the path where stinit.def file is stored.

By default, **stinit** searches your present working directory to find the stinit.def file. If **stinit** cannot find stinit.def in the working directory, it searches /etc/stinit.def.

For more information on **stinit**, see the stinit(8) man page.

Tape Device and Parameter Definitions

The stinit.def file contains definitions of tape devices and their corresponding initialization parameters.

Some of the parameter conventions are listed in the following table:

Item	Definition
{ }	Parameter definitions are delimited by { }.
name = *value*	Definitions consist of pairs where name = *value*. The *value* is either a numeric parameter, a string not containing blanks, or a string enclosed within quotation marks. If = *value* is omitted, a value of "1" is assigned.
#	If the # character appears in an input line, **stinit** deletes from the buffer everything following the # character up to the next carriage return. This enables you to make comments in the stinit.def file.

The following example shows a single entry of a tape device in the stinit.def file:

```
# The XY dat
manufacturer=XY-COMPANY model = "UVW DRIVE" {
scsi2logical=1 # Common definitions for all modes
can-bsr can-partitions auto-lock
# Definition of modes
```

```
mode1 blocksize=0 compression=1
mode2 blocksize=1024 compression=1
mode3 blocksize=0 compression 0
mode4 blocksize = 1024 compression 0 }
```

Identifying Tape DevicesandParameters

You identify tape devices and parameters using keywords. The keywords correspond to the data returned by the tape device in response to a SCSI INQUIRY command. The matches are case-sensitive and performed up to the length defined in the configuration file. Partial matches are permitted.

> **Note:** You may abbreviate some keywords. Some keywords show a portion of the word enclosed within square brackets [ ]. The portion enclosed within the brackets is not required when entering commands. For example, the keyword block[size] means you can signify block size by entering either blocksize or block.

Common Tape Device Keywords

The following table lists the most common tape device keywords.

Keyword	Description
manufacturer=	Specifies the string that must match the vendor identification returned by the tape device; for example, QUANTUM.
model=	Specifies the string that must match the product identification returned by the tape device; for example, SDLT600.
revision=	Specifies the string that must match the product revision level returned by the tape device; for example, 1E1E, which represents V30.

Common Parameter Keywords

Following are some common parameter keywords. For a more thorough description of the keywords used for tape devices and parameters, see the following references:

- stinit(8) man page
- st(4) man page
- /usr/src/linux-<kernel revision>/drivers/scsi/README.st.
- Appendix C – Sample stinit.def Definitions File on page 26

Keyword	Description
block[size]=*value*	The tape block size can be set to value bytes. Quantum recommends using the default block[size] = 0, signifying variable block mode.
comp[ression]= *value*	Compression of the data by the drive is enabled if *value* does not equal zero. Note that the tape driver cannot enable compression for all drives that can compress data. Some drives define compression using density codes. Quantum does not use density codes and requires compression to be enabled. The compression default setting is determined by stinit.def. If stinit.def is not found, compression defaults to ON.

All the matching initializations are collected in the order they are defined in the stinit.def file. This means that you can define global parameters that apply to all devices by placing them before all tape device definitions in stinit.def. For an example, see # Global Keywords and Values in Appendix C – Sample stinit.def Definitions File.

mtx Tape Library Tool

mtx is a tape library media management tool. This section shows you how to determine if **mtx** is installed and how to install it.

The use of **mtx** is beyond the scope of this document.

Verifying mtx is Installed

To verify **mtx** is installed, execute the following command:

```
# rpm -qa mtx
```

This command returns the name of the package followed by a version number (for example, RHEL2.1 returns mtx-1.2.13-1).

If no information is returned, mtx is not installed on the system. To install mtx, see Installing mtx.

> **Note:** **mtx rpm** is not included in the Red Hat Enterprise Linux 3.0. If you are running RHEL 3.0, you will not be able to use the **mtx** module.

Installing mtx

mtx is typically included on your Linux installation CD-ROMs (except for RHEL3.0).

To install **mtx**, execute the following command:

```
# rpm -ivh mtx*
```

where the asterisk represents the remaining portion of the file name (in this case, it is the version number).

For a detailed description of these access modes, refer to the appropriate product manual for your DLTtape mini-library.

Appendix A – Tape Drive and Cartridge Compatibility

This section provides information about tape cartridge and tape drive compatibility. Use these tables to determine which cartridges to use in your tape drive.

Tape Drive and Cartridge Compatibility – DLT

	Drive Type				
Tape Cartridge	**DLT 2000 DLT 2500 DLT 2700**	**DLT 2000XT DLT 2500XT DLT 2700XT**	**DLT 4000 DLT 4500 DLT 4700**	**DLT 7000**	**DLT 8000**
DLTtape III (CompacTape™ III)	10/20 GB	10/20 GB (read only)	10/20 GB (read only)	10/20 GB (read only)	10/20 GB (read only)
DLTtape IIIXT (CompacTape III XT)	N/A	15/30 GB	15/30 GB (read only)	15/30 GB (read only)	15/30 GB (read only)
DLTtape IV (CompacTape IV)	N/A	N/A	20/40 GB	20/40 GB (read only) 35/70 GB	20/40 GB (read only) 35/70 GB (read only) 40/80 GB
Cleaning Tape III	20 uses	20 uses	20 uses	20 uses	20 uses
Cleaning Tape IV	N/A	N/A	N/A	20 uses	20 uses

Tape Drive and Cartridge Compatibility – SDLT

	Drive Type		
Tape Cartridge	**SDLT 220**	**SDLT 320**	**SDLT 600**
DLTtape IV (CompacTape IV)	20/40 GB (read only) 35/70 GB (read only) 40/80 GB (read only)	20/40 GB (read only) 35/70 GB (read only) 40/80 GB (read only)	N/A
Super DLTtape I	110/220 GB	110/220 GB 160/320 GB	110/220 GB (read only) 160/320 GB (read only)
Super DLTtape II	N/A	N/A	300/600 GB
DLTtape VS1	N/A	N/A	80/160 GB (read only)
Cleaning Tape III	N/A	N/A	N/A
SDLT Cleaning Tape	20 uses	20 uses	20 uses

Tape Drive and Cartridge Compatibility – VS

	Drive Type			
Tape Cartridge	**DLT1**	**VS80**	**VS160**	**DLT-V4**
DLTtape IV (CompacTape IV)	40/80 GB	40/80 GB	40/80 GB (read only)	40/80 GB (read only)
DLTtape VS1	N/A	N/A	80/160 GB	80/160 GB (read only) 160/320 GB
DLTtape IV Cleaning Tape	20 uses	20 uses	N/A	N/A
VS160 Cleaning Tape	N/A	N/A	20 uses	20 uses

Tape Drive and Cartridge Compatibility – LTO

	Drive Type		
Tape Cartridge	**LTO-1**	**LTO-2**	**LTO-3**
Ultrium 1	100/200	100/200 (read only)	100/200 (read only)
Ultrium 2	N/A	200/400	200/400 (read only)
Ultrium 3	N/A	N/A	400/800
Ultrium Cleaning Tape	20 uses	20 uses	20 uses

Tape Drive and Cartridge Compatibility – DDS/DAT

	Drive Type		
Tape Cartridge	**DDS3**	**DDS4/SP40**	**DAT 72**
DDS2	4/8	4/8	N/A
DDS3	12/24	12/24	12/24
DDS4	N/A	20/40	20/40
DAT 72	N/A	N/A	36/72
Cleaning Tape	40 uses	40 uses	40 uses

Appendix B – Linux Reference Documentation

For further information about the topics discussed in this guide, refer to the following documentation:

- stinit(8) man page
- /usr/share/doc/mt-st-<version>/stinit.def.examples
- st(4) man page
- dmesg(8) man page
- /usr/src/linux-<kernel revision>/Documentation/devices.txt
- /usr/src/linux-<kernel revision>/drivers/scsi/README.st
- mt(1) man page
- mtx(1) man page

Appendix C – Sample stinit.def Definitions File

This section provides a sample stinit.def definitions file. You must set up your stinit.def file if you have not done so already.

Recommendations

This sample contains Quantum's recommendations for setting up your stinit.def file.

You may use the information provided here (you can copy and paste the information directly into your stinit.def file or enter it manually). You may modify this information as needed. You may also create your own stinit.def file.

Caution: Your system configuration may not be compatible with this particular stinit.def file. Refer to your system documentation before implementing any stinit.def file.

Sample File

```
# Red Hat Linux 9.0, EL 2.1, EL 3.0
# Novell SuSE Linux 8.0, 9.0
#
# Quantum Corporation
# Jeff Willener
#
# 8-29-2003
# Initial Release
#
# 12-20-2004
# Add SDLT600 Support
#
# 6-17-2005
# Add LTO, LTO2, LTO3 support
#
# 11-1-2005
# Add DAT72, DLT-V4 support
#
# 3-7-2010
# Added S4 Support
#
# See also: man page stinit(8),
#                   /usr/share/doc/mt-st-*/stinit.def.examples,
#                   man page st(4),
#                   man page mt(1),
#                   man page stinit(8)

# Supported Quantum Devices:
# QUANTUM DLTS4
# Density Codes:
# 0x4B 800.0 GB (DLTtape S4)
# Alternate Density Codes Not Supported
```

```
# QUANTUM  SDLT600
#          Density Codes:
#          0x4A     320.0 GB                    (Super DLTtape 2)
#          Alternate Density Codes Not Supported
# QUANTUM  SDLT320
#          Density Codes:
#          0x49     160.0 GB                    (Super DLTtape 1)
#          0x92     160.0 GB compression off    (Super DLTtape 1)
#          0x93     320.0 GB compression on     (Super DLTtape 1)
# QUANTUM  SDLT220
#          0x48     100.0 GB                    (Super DLTtape 1)
#          0x90     100.0 GB compression off    (Super DLTtape 1)
#          0x91     200.0 GB compression on     (Super DLTtape 1)
# QUANTUM  DLT8000
#          0x41     40.0  GB                    (DLTtape IV)
#          0x88     40.0  GB compression off    (DLTtape IV)
#          0x89     80.0  GB compression on     (DLTtape IV)
# CERTANCE ULTRIUM 3
#          0x44     400.0 GB
# CERTANCE ULTRIUM 2
#          0x42     200.0 GB
# CERTANCE ULTRIUM
#          0x40     100.0 GB
# CERTANCE DAT72
#          0x47     36.0 GB                     (DAT 72)
# QUANTUM  DLT-V4
#          0x51     160.0 GB                    (DLTtape VS1)
# QUANTUM  DLT VS160
#          0x50     80.0  GB                    (DLTtape VS1)
#          0x98     80.0  GB compression off    (DLTtape VS1)
#          0x99     160.0 GB compression on     (DLTtape VS1)
# BNCHMARK DLT1
#          0x40     40.0  GB                    (DLTtape IV)
#          0x86     40.0  GB compression off    (DLTtape IV)
#          0x87     80.0  GB compression on     (DLTtape IV)

# /usr/src/linux-<kernel revision>/Documentation/devices.txt
# 9 --> Kernel Device Index Number
# char --> Character Device
# 0-255 --> SCSI Tape Device Node
#
# 9 char SCSI tape devices
# 0 = /dev/st0 First SCSI tape, mode 0
# 1 = /dev/st1 Second SCSI tape, mode 0
# ...
# 32 = /dev/st0l First SCSI tape, mode 1
# 33 = /dev/st1l Second SCSI tape, mode 1
#...
# 64 = /dev/st0m First SCSI tape, mode 2
# 65 = /dev/st1m Second SCSI tape, mode 2
#...
# 96 = /dev/st0a First SCSI tape, mode 3
# 97 = /dev/st1a Second SCSI tape, mode 3
#...
# 128 = /dev/nst0 First SCSI tape, mode 0, no rewind
# 129 = /dev/nst1 Second SCSI tape, mode 0, no rewind
#...
```

```
# 160 = /dev/nst0l First SCSI tape, mode 1, no rewind
# 161 = /dev/nst1l Second SCSI tape, mode 1, no rewind
#...
# 192 = /dev/nst0m First SCSI tape, mode 2, no rewind
# 193 = /dev/nst1m Second SCSI tape, mode 2, no rewind
#...
# 224 = /dev/nst0a First SCSI tape, mode 3, no rewind
# 225 = /dev/nst1a Second SCSI tape, mode 3, no rewind

# Global Keywords and Values
drive-buffering=1
#scsi2logical=1
no-wait=0
buffering=0
async-writes=0
read-ahead=1
two-fms=0
auto-lock=0
fast-eom=1
can-bsr=1
noblklimits=0
# can-partitions=0
# QUANTUM DLTS4
manufacturer=QUANTUM model="DLTS4" {
timeout=3600 # 1 hour timeout
long-timeout=14400# 4 hour long timeout
can-partitions=0
mode1 blocksize=0 density=0x4B compression=1 # DLTS4 density, compression on
mode2 blocksize=0 density=0x4B compression=0 # DLTS4 density, compression off
mode3 blocksize=0 density=0x4A compression=1 # SDLT600 density, compression on
mode4 blocksize=0 density=0x49 compression=1 # SDLT320 density, compression on
}
# QUANTUM SDLT600
manufacturer=QUANTUM model="SDLT600" {
timeout=3600      # 1 hour timeout
long-timeout=14400# 4 hour long timeout
can-partitions=0
mode1 blocksize=0 density=0x4A compression=1 # SDLT600 density, compression on
mode2 blocksize=0 density=0x4A compression=0 # SDLT600 density, compression off
mode3 blocksize=0 density=0x49 compression=1 # SDLT320 density, compression on
mode4 blocksize=0 density=0x48 compression=1 # SDLT220 density, compression on
}

# QUANTUM SDLT320
manufacturer=QUANTUM model="SDLT320" {
timeout=3600      # 1 hour timeout
long-timeout=14400# 4 hour long timeout
can-partitions=0
mode1 blocksize=0 density=0x49 compression=1 # SDLT320 density, compression on
mode2 blocksize=0 density=0x49 compression=0 # SDLT320 density, compression off
mode3 blocksize=0 density=0x48 compression=1 # SDLT220 density, compression on
mode4 blocksize=0 density=0x48 compression=0 # SDLT220 density, compression off
}

# QUANTUM SDLT220
manufacturer=QUANTUM model="SuperDLT1" {
timeout=3600
```

```
long-timeout=14400
can-partitions=0
mode1 blocksize=0 density=0x48 compression=1 # SDLT220 density, compression on
mode2 blocksize=0 density=0x48 compression=0 # SDLT220 density, compression off
mode3 blocksize=0 density=0x41 compression=1 # DLT8000 density, compression on
mode4 blocksize=0 density=0x41 compression=0 # DLT8000 density, compression off
}

# QUANTUM DLT8000
manufacturer=QUANTUM model="DLT8000" {
timeout=3600
long-timeout=14400
can-partitions=0
mode1 blocksize=0 density=0x41 compression=1 # DLT8000 density, compression on
mode2 blocksize=0 density=0x41 compression=0 # DLT8000 density, compression off
mode3 blocksize=0 density=0x1B compression=1 # DLT7000 density, compression on
mode4 blocksize=0 density=0x1B compression=0 # DLT7000 density, compression off
}

# CERTANCE ULTRIUM 3
manufacturer=CERTANCE model="ULTRIUM 3" {
timeout=800
long-timeout=14400
can-partitions=0
mode1 blocksize=0 density=0x44 compression=1 # ULTRIUM 3 density, compression on
mode2 blocksize=0 density=0x44 compression=0 # ULTRIUM 3 density, compression off
mode3 blocksize=0 density=0x42 compression=1 # ULTRIUM 2 density, compression on
mode4 blocksize=0 density=0x40 compression=1 # ULTRIUM   density, compression on
}

# CERTANCE ULTRIUM 2
manufacturer=CERTANCE model="ULTRIUM 2" {
timeout=800
long-timeout=14400
can-partitions=0
mode1 blocksize=0 density=0x42 compression=1 # ULTRIUM 2 density, compression on
mode2 blocksize=0 density=0x42 compression=0 # ULTRIUM 2 density, compression off
mode3 blocksize=0 density=0x40 compression=1 # ULTRIUM   density, compression on
mode4 blocksize=0 density=0x40 compression=0 # ULTRIUM   density, compression off
}

# CERTANCE ULTRIUM
manufacturer=SEAGATE model="ULTRIUM06242-XXX" {
timeout=800
long-timeout=14400
can-partitions=0
mode1 blocksize=0 density=0x40 compression=1 # ULTRIUM   density, compression on
mode2 blocksize=0 density=0x40 compression=0 # ULTRIUM   density, compression off
mode3 blocksize=0 density=0x40 compression=1 # ULTRIUM   density, compression on
mode4 blocksize=0 density=0x40 compression=1 # ULTRIUM   density, compression on
}

# CERTANCE DAT72
manufacturer=SEAGATE model="DAT    DAT72" {
timeout=600
long-timeout=-10800
can-partitions=0
```

```
mode1 blocksize=512 density=0x47 compression=1 # DAT72 density, compression on
mode2 blocksize=512 density=0x47 compression=0 # DAT72 density, compression off
mode3 blocksize=512 density=0x26 compression=1 # DDS4  density, compression on
mode4 blocksize=512 density=0x25 compression=1 # DDS3  density, compression on
}

# QUANTUM DLT-V4
manufacturer=QUANTUM model="DLT-V4" {
timeout=3600
long-timeout=14400
can-partitions=0
mode1 blocksize=0 density=0x51 compression=1 # DLT-V4 density, compression on
mode2 blocksize=0 density=0x51 compression=0 # DLT-V4 density, compression off
mode3 blocksize=0 density=0x50 compression=1 # VS160  density, compression on
mode4 blocksize=0 density=0x40 compression=1 # VS80   density, compression on

# QUANTUM DLT VS160
manufacturer=QUANTUM model="DLT VS160" {
timeout=3600
long-timeout=14400
can-partitions=0
mode1 blocksize=0 density=0x50 compression=1 # VS160 density, compression on
mode2 blocksize=0 density=0x50 compression=0 # VS160 density, compression off
mode3 blocksize=0 density=0x40 compression=1 # VS80  density, compression on
mode4 blocksize=0 density=0x40 compression=0 # VS80  density, compression off
}

# BNCHMARK VS80
manufacturer=BNCHMARK model="DLT1" {
timeout=180
long-timeout=14400
can-partitions=0
mode1 blocksize=0 density=0x40 compression=1 # VS80 density, compression on
mode2 blocksize=0 density=0x40 compression=0 # VS80 density, compression off
mode3 blocksize=0 density=0x41 compression=1 # DLT8000 density, compression on
mode4 blocksize=0 density=0x41 compression=0 # DLT8000 density, compression off
}
```

Appendix D – Troubleshooting

This section covers common errors.

Problem	Explanation	Fix
You receive a non-recoverable error that looks similar to this after performing the tar command: tar: /dev/st0: Wrote only 0 of 10240 bytes tar: Error is not recoverable: exiting now	The tape device has a blocksize set to a parameter value other than 0.	Use **mt** to set the blocksize to 0 so the tape drive can operate in variable blocksize mode. Execute this command: # mt -f /dev/st0 setblk 0 **Note:** st0 is used as the device node for this example only; be sure to use the correct mode0 node.
No information is returned when you execute the command # rpm -qa | grep mt-st to verify **mt-st** is installed.	**mt-st** is not installed.	Install **mt-st** (see Installing mt-st on page 17).
No information is returned when you execute the command # rpm -qa mtx to verify **mtx** is installed.	**mtx** is not installed.	Install **mtx** (see Installing mtx on page 21).
modinfo st reports the **st** module name not found.	**st** module is not loaded.	Load the **st** module. Consult your Linux manuals for instructions.
The **st** device nodes are not present in the /dev directory.		Create the device nodes. Use the mknod commands described in Creating Auto-Rewind Device Nodes on page 16 and Creating No-Rewind Device Nodes on page 16 Refer to the following documentation for more detailed instructions: • Your Linux documentation • st(4) man page • mknod(1) man page
The backup application does not work.	Your backup application may not be compatible with the stinit.def file you are using.	Check your backup application documentation for information on what type of stinit.def files you can use; then rewrite the stinit.def file.

Problem	Explanation	Fix
The **st** driver does not see all attached tape devices.	The connector cable may be too long.	Verify that the connector cable length does not exceed the specifications listed in your product manual.
	Your SCSI bus may not be terminated properly.	Ensure the SCSI bus is terminated properly (see your product manual for instructions).
	You don't have enough **st** tape device nodes.	Verify that you have enough **st** tape device nodes for all your attached tape devices. If you need to create more, see instructions in the following sections: • Identifying Device Modes on page 15 • Creating Device Nodes on page 15 • Creating No-Rewind Device Nodes on page 16