Quantum.

CLI Reference Guide

Quantum StorNext 3.5.3

Quantum StorNext CLI Reference Guide, 6-67340-01 Rev A, June 2011, Product of USA.

Quantum Corporation provides this publication "as is" without warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability or fitness for a particular purpose. Quantum Corporation may revise this publication from time to time without notice.

Contents

Contents

Introduction

Quantum recommends using the GUI to complete most StorNext tasks, but there might be situations where you prefer or need to use the command line interface (CLI) instead.

This document describes commonly used StorNext File System (SNFS) and StorNext Storage Manager (SNSM) commands. For each command, the following information is provided:

- Description of what the command does and how it is used.

- Command syntax that shows how to compose the command.

- List of the options and arguments of the command.

Note: Some descriptions also include command examples.

This document contains the following sections:

On a Windows Vista system, when you run applications from the command line that require administrative privileges, you can start the CLI application either from an elevated shell environment or a DOS shell.

Note: For Windows Vista users only: If the CLI application is started from an alternative shell (such as those provided by **cygwin**,) you *must* start that shell in an elevated mode. If you do not, CLI commands requiring administrative privileges will fail and you will receive an error message indicating that you do not have sufficient privileges to run the command. To start the shell in elevated mode, right-click the icon for Command Prompt or Cygwin and select Administrative Mode.

Viewing man Pages

This document summarizes information from the **man** pages for the included SNFS and SNSM commands. The **man** pages contain detailed usage information and examples for many StorNext commands.

Use the following procedure to view the **man** page for a command.

1 Source the StorNext profile. Do one of the following:

- If using the **bash** shell, at the command prompt, type:

  source /usr/adic/.profile

- If using the **csh** or **tcsh** shell, at the command prompt, type:

  source /usr/adic/.cshrc

2 View the **man** page for a command. At the command prompt, type:

man <command>

where **<command>** is the command for which you want to view the **man** page.

3 Press **<Spacebar>** to page through the **man** page.

4 When you are finished, type **q** and press **<Enter>** to exit the **man** page.

Note: Paging through **man** pages may work differently depending on the viewer specified by the **$PAGER** environmental variable.

File System Commands

This section describes the following StorNext File System commands:

cvadmin

cvadmin is an interactive utility used for general administration of a file system. With **cvadmin**, you can display file system and client status, activate a file system, view and modify stripe group attributes, administer user and group quotas, and perform other administrative tasks.

Invoke **cvadmin** to start an interactive session, and then use **cvadmin** commands to display and modify the active SNFS configuration. Changes remain in effect as long as the current FSM (file system manager) is running. To make permanent changes, edit the configuration file.

Syntax

cvadmin [-H <FSMHostName>] [-F <FileSystemName>] [-f <filename>]

[-e <command> [-e <command>] ...]

In **cvadmin** interactive mode (**snadmin**), several commands use name as an argument. For example, the **select**, **start**, and **stop** commands use **file_system_name** as an argument and the **up** and **down** commands use the stripe group (**groupname**) as an argument. If the value of **file_system_name** or **groupname** is identical to the **snadmin** command name or command keyword, then use double quotation marks on the command argument. For example, to start a file system named **quotas** where **"quotas"** is the **snadmin** command name, the syntax is:

snadmin (localfs) > start "quotas"

Additionally, **snadmin** command keywords must use double quotation marks if they are used as quota user or group names. For example, **user** is a **snadmin** command keyword, so it would need double quotation marks:

snadmin (localfs) > quotas set group "user" 11G 10G 1440

When creating a group quota, you cannot name the group **"user"**.

Note:	**cvadmin** no longer supports the **enable** or **disable** commands. To enable or disable a file system for reads/writes, modify the configuration file and then restart the FSM.

Options / Arguments

Table 1 lists options and arguments for the **cvadmin** command. Commands that can be used in interactive mode are also listed.

Note:	You must quote commands with more than one word. For example: **cvadmin -e 'start snfs1'**

Table 1 Options and Arguments: cvadmin

Options / Arguments	Description
[-H <FSMHostName>]	Connects to the FSM (file system manager) on the machine **FSMHostName**. (By default **cvadmin** attempts to connect to an FSM on the local machine.)
[-F <FileSystemName>]	Sets **FileSystemName** as the active file system in **cvadmin**.
[-f <filename>]	Reads commands from the specified file.
[-e <command>]	Runs the **cvadmin command** and exits. (Commands are listed below.)
activate <file_system_name>	Activates the file system **file_system_name**.

Options / Arguments	Description
debug [[+ | -] <value>]	Views or sets SNFS debugging flags. Enter **debug** with no value to return the current settings, the location of the FSM log file, and a legend describing what each setting does.
disks [refresh]	Displays SNFS disk volumes visible to this machine. If the **refresh** option is used, the volumes are re-scanned.
down <groupname>	Disables access to the stripe group **groupname**.
fail <file_system_name | index_number>	Initiate failover of file system **file_system_name** or file system **index_number**.
fsmlist [<file_system_name>] [on [<hostname> | <IP_address>]]	Displays the state (running or not running) of FSM processes for the specified file system, hostname, or IP address.
filelocks [yes | no]	Shows file record lock enforcement. (File locks are automatically used on UNIX or Linux.)
help <command>	Displays a usage summary of **command**.
latency-test [<index_number> | all] [seconds]	Runs a network latency test between the FSM and a client or all clients.
paths	Displays SNFS labeled disks visible to the local system.
proxy [long]	Displays distributed LAN servers for the selected file system. With the long argument, the served disks are also displayed.
proxy who <hostname>	Displays all active distributed LAN connections for the specified host.

Options / Arguments	Description
qos	Displays per-stripe group QOS statistics.
repof	Creates a report showing all open files on the file system.
quit	Disconnects **cvadmin** from SNFS and exits.
**quotas [yes	no]**	Enables or disables quota accounting and enforcement. Enter **quotas** with no value to see current settings.
**quotas get [user	group] <name>**	Displays current quota parameters for user or group **name**.
**quotas set [user	group] <name> <hardlim> <softlim> <timelim>**	Sets current quota parameters for user or group **name**. • **hardlim**: The maximum amount of space (in bytes) the specified user or group can use. • **softlim**: The amount of space used (in bytes) before a warning is sent. (Typically this is 80% of the hard limit.) • **timelim**: The amount of time (in minutes) it takes for the soft limit to turn into a hard limit. Setting a parameter to 0 disables quota enforcement.
quotacheck	Recalculates the amount of space consumed by all users and groups in the file system.
repquota	Generates quota reports for all users and groups in the file system.
select <file_system_name>	Selects file system **file_system_name** for viewing or modification.

Options / Arguments	Description
show [<groupname>] **[long]**	Displays all stripe groups or a specific stripe group.
start <file_system_name> **[<hostname> |** **<IP_address>]**	Starts the SNFS manager for file system **file_system_name**.
stat	Displays the status of the currently selected file system.
stop <file_system_name>	Stops the SNFS manager for file system **file_system_name**.
up <groupname>	Enables the stripe group **groupname**.
who	Displays a list of mounted clients on the currently selected file system.

Examples

- Enter **cvadmin** interactive mode (**snadmin**):

    cvadmin

- Show all stripe groups in the file system:

    cvadmin -F <file_system_name> -e show

- Display which FSM debug flags are set:

    cvadmin -F <file_system_name> -e debug

- Display the long version of the strip1 stripe group:

    cvadmin -F <file_system_name> -e 'show stripe1 long'

- Down the stripe group named stripe1:

    cvadmin -F <file_system_name> 'down stripe1'

cvaffinity

The **cvaffinity** command is used to set a session affinity type. When a file system is configured, an affinity (a name describing a special media type or property) may be created in a stripe group. Once a stripe group

affinity is established for a session, all allocations of files associated with the session are made on stripe groups having the specified affinity in the affinity set list. If an affinity does not exist for any of the stripe groups, then the allocation occurs on the non-exclusive data pool. If there is no non-exclusive data pool, then **ENOSPC** (no space) is returned.

Command Syntax

cvaffinity [-k <key>] <filename>

cvaffinity [-s <key>] <filename>

cvaffinity [-l] <filename>

cvaffinity [-d] <filename>

Options / Arguments

Table 2 lists options and arguments for the **cvaffinity** command.

Table 2 Options and Arguments: cvaffinity

Options / Arguments	Description
[-k <key>]	Indicates where the file system should place the data file. If the **key** parameter is defined, then the data file is placed on stripe groups specified to support this parameter.
	If there is no stripe group with the specified key, the file is placed on the non-exclusive data pool. If there is no non-exclusive data pool, then **ENOSPC** (no space) is returned.
[-s <key>]	Associate affinity **key** with file or directory **filename**.
[-l]	Lists the affinity for the file **filename**.
[-d]	Deletes the affinity from the file or directory **filename**.
<filename>	File or directory name.

Examples

- Set the stripe group that supports the specified affinity key, with the mount point of the file system as a reference handle:

    cvaffinity -k 6100_n8 /usr/clips

- Turn off the stripe group affinity, using the mount point of the file system as a reference handle:

    cvaffinity /usr/clips

cvcp

The **cvcp** command provides a high speed, multi-threaded copy mechanism to copy directories and **tar** images on and off SNFS.

The **cvcp** commands works in a number of modes:

- Directory-to-directory copies of regular files
- Single file-to-file copy
- Data stream to a target directory
- Single file or directory copy to an output stream

The **cvcp** utility can be tuned to improve performance and resource utilization by adjusting the **-b**, **-k**, and **-t** options.

- **-b** option specifies the number of copy buffers. This option should be set between one to three times the number of copy threads. Increasing the number of copy buffers increases the amount of work that queues waiting for an available copy thread, but it also consumes more resources.

- **-k** option specifies the size of the copy buffer. This option can be tuned to fit the I/O characteristics of a copy. If files smaller than 4MB are being copied, performance may be improved by reducing this setting to match the copy buffer size to the source file size.

- **-t** option specifies the number of copy threads. This option is useful when copying large directory structures. Single file copies are not affected by the number of copy threads.

Command Syntax

**cvcp [-A] [-b <buffers>] [-d] [-k <buffer_size>] [-l] [-n]
[-p <source_prefix>] [-s] [-t <num_threads>] [-u] [-v] [-x] [-
y] [-z] <source> <destination>**

Options / Arguments

Table 3 lists options and arguments for the **cvcp** command.

Table 3 Options and
Arguments: cvcp

Options / Arguments	Description
<source>	Source file or directory.
<destination>	Destination file or directory.
[-A]	Disables file pre-allocation.
[-b <buffers>]	Sets the number of I/O buffers to **buffers**. Default: two times the number of copy threads started
[-d]	Creates a sub-directory called Source in the Destination directory, then copies the files and sub-directories under Source to the new sub-directory.
[-k <buffer_size>]	Sets the I/O buffer size to **buffer_size** in bytes. Default: 4MB
[-l]	Copies the target of symbolic links rather than copying the link itself.
[-n]	Prevents recursion through subdirectories.
[-p <source_prefix>]	Only copies files that begin with the value of the **source_prefix** parameter.

Options / Arguments	Description
[-s]	Forces allocations to line up on the beginning block modulus of the stripe group.
[-t <num_threads>]	Specifies the number of copy threads. Default: 4
[-u]	Only copies when the source file is newer than the destination file or the destination file does not exist.
[-v]	Displays verbose output about copied files.
[-x]	Retains original permissions from the source file.
[-y]	Retains ownership and group information (if set by the super user).
[-z]	Retains original modification times.

Examples

- Copy the **abc** directory and its sub-directories to **/usr/clips/foo**. This example uses the default number of copy threads and buffers, has a total buffer pool size of 32MB (8 buffers of 4MB each), retains all permissions and ownerships, and shows all files being copied.

    cvcp -vxy abc /usr/clips/foo

- Within the **abc** directory, copy all files that start with **mumblypeg**:

    cvcp -vxy -p mumblypeg abc /usr/clips/foo

cvfsck

The **cvfsck** command is used to check and repair a file system that was corrupted by a system crash, bad disk, or other catastrophic failure.

Command Syntax

cvfsck [-C] [-d] [-e] [-f] [-g] [-G] [-j] [-J] [-K] [-l] [-L] [-M] [-n]
[-r] [-t] [-T] [-v] [-x] [-T <directory>] <FsName> [<FsPath>]

Options / Arguments

Table 4 lists options and arguments for the **cvfsck** command.

Table 4 Options and Arguments: cvfsck

Options / Arguments	Description
[-C]	Reset the inode free list. Use only if **cvfsck** cannot repair free-list inconsistencies.
[-d]	Enables internal debugging.
[-e]	Reports statistics on all file extents.
[-f]	Reports fragmentation.
[-g]	Prints the journal recovery log.
[-G]	Exits immediately after **cvfsk** completes without waiting for a key press (Windows systems only).
[-j]	Executes journal recovery and exits.
[-J]	Dumps the raw journal to **stdout** and exits.
[-K]	Resets the journal and exits. **WARNING:** Resetting the journal may introduce metadata inconsistency. Exercise extreme caution when using this option.
[-l]	Writes information to **syslog**.

Options / Arguments	Description
[-L <inode_number>]	Reassigns orphaned files to the specified directory rather than to the root of the file system. The **inode_number** of the directory must be a 64-bit value. **Note:** Use the **dc pathname** command in **cvfsdb** to find the 64-bit inode number of a directory.
[-M]	Determines if a new metadata dump is needed and prints an explanation. If used, **cvfsck** exits with one of the following statuses: • **0** - Dump not needed. • **1** - Dump needed. • **2** - An error occurred.
[-n]	Does not update metadata (read only).
[-r]	Reports statistics on all files.
[-t]	Reports on allocation covering restricted areas of a stripe group.
[-T <directory>]	Places temporary files in the specified directory.
[-v]	Displays verbose output.
[-x]	Reports statistics for input to a spreadsheet.
<FsName>	Name of the file system to check.
[<FsPath>]	An optional, alternate file system path (other than **/usr/cvfs/data**) to locate file systems.

cvfsid

The **cvfsid** command provides a mechanism to display the SNFS identifier for the executing system. This identifier is submitted to the Quantum Technical Assistance Center, which issues license authorization keys.

Command Syntax

cvfsid [-?] [-h] [-l] [-n]

Options / Arguments

Table 4 lists options and arguments for the **cvfsid** command.

Table 5 Options and Arguments: cvfsid

Options / Arguments	Description
[-?] [-h]	Displays command help.
[-l]	Lists the local host's Authorizing IDs, IP addresses, and MACs (Linux only).
[-n]	Displays the network information in a compact, machine readable form.

Examples

• Display the SNFS identifier for the executing system.

    cvfsid

cvlabel

The **cvlabel** command is used to label SNFS disk devices. One client with visibility to all SAN disk devices must execute **cvlabel -l** or **cvlabel -L** to create a list of disk labels. The **mount_cvfs** (1M) process uses the volume labels to determine which disk drives are used for SNFS stripe group nodes.

Quantum recommends that the **cvlabel** command first be used with the **-l** option, which presents all disk devices found on the system. The next step is to create a label list file with the -c option.

Caution: Use the **cvlabel** command with extreme caution. Incorrect use of this command may destroy any existing file system data for the named file system.

Note: The **cvlabel** command is used only to view or modify disk labels. It is not used to determine the LUNs a file system has access to. To see the LUNs a file system can access, or the paths being used to access LUNs, use the **cvadmin** command with the **disks** or **paths** option. For more information, see cvadmin on page 3.

Command Syntax

cvlabel -l [-s] [-v] [-a] [-g] [-F <filter>]

cvlabel -L [-v] [-a] [-g] [-F <filter>]

cvlabel -c [-T] [-F <filter>]

cvlabel [-v] [-f] [-q <tag_q_depth>] [-r] [-R] <label_list>

cvlabel [-f] -u <VolumeName>

cvlabel [-f] -U <DeviceName>

cvlabel -D <VolumeName>

Options / Arguments

Table 6 lists options and arguments for the **cvlabel** command.

Table 6 Options and Arguments: cvlabel

Options / Arguments	Description
-l	Lists available disk devices on the system (short format).
-L	Lists available disk devices on the system (verbose format).
-u <VolumeName>	Unlabels the volume **VolumeName**.

Options / Arguments	Description
-U <DeviceName>	Unlabels the raw disk device **DeviceName**.
[-s]	Prints the disk device serial number when used with the **-l** option.
[-v]	Reports verbose output about labeling.
[-a]	Prints unusable disk devices when used with the **-l** or **-L** options.
[-g]	Prints GUID information for EFI-labeled disks when used with the **-l** or **-L** options.
[-F <filter>]	Lists only devices whose inquiry string contains the filter string when used with the **-c**, **-l**, or **-L** option.
[-q <tag_q_depth>]	Sets the Command Tag Queue Depth to **tag_q_depth** on IRIX systems. Default: 16
[-f]	Forces labeling (without confirmation) even if a label exists.
-c	Outputs to **stdout** a list of all located devices in a format that can later be used as a **label_list** file.
[-T]	When used with the **-c** option, outputs to **stdout** a list of devices that can be converted from the old VTOC format to the new EFI format.
-D <VolumeName>	Writes label information of volume VolumeName to **stdout**.
[-r]	Forces a disk to be relabeled even if there are no changes to the label information.

Options / Arguments	Description
[-R]	Displays the default RAID strings inquiry table.
[-e]	Changes the label type for disks between 1TB and 2TB from VTOC to EFI.
[-E]	Changes the default label type for all disks to EFI.
<label_list>	File that specifies StorNext disk labels.

Examples

- List all disk devices in a system.

  /usr/cvfs/bin/cvlabel -l

- Label the disk drives.

  /usr/cvfs/bin/cvlabel /tmp/mylabelfile

- Re-label the disk drives using the force (**-f**) and verbose (**-v**) options.

  /usr/cvfs/bin/cvlabel -fv /tmp/mylabelfile

cvmkdir

The **cvmkdir** command is used to create a SNFS directory and attach the affinity parameter (**-k <key>**) to it. If no option is used and there is an existing directory, the **cvmkdir** command output displays the assigned affinity. Once an affinity is assigned to a directory, it cannot be altered. If no option is used and the directory does not exist, the directory will not be created.

Command Syntax

cvmkdir [-k <key>] <dirname>

Options / Arguments

Table 7 lists options and arguments for the **cvmkdir** command.

Options / Arguments	Description
[-k <key>]	Indicates the affinity to associate with the directory. All new sub-directories and files beneath this directory inherit its affinity as they are created. If the affinity is changed or removed, only files or directories created after the change are affected.
<dirname>	Path where the directory is created.

cvmkfile

The **cvmkfile** command is used to create a pre-allocated file on SNFS. This is a useful method when preparing a file for use in a real-time or streaming environment because the entire file is represented in only one file system extent. Additionally, a file can be placed on a specific stripe group by specifying the key parameter (used as an affinity locator).

Command Syntax

cvmkfile [-k <key>] [-p] [-s] [-w] [-z] <size>[k|m|g|t] <filename>

Options / Arguments

Table 8 lists options and arguments for the **cvmkfile** command.

Options / Arguments	Description
[-k <key>]	Indicates where the file system should place the data file. If the **key** parameter is defined, then the data file is place on stripe groups specified to support the **key** parameter.

Options / Arguments	Description
[-p]	Forces the allocation and subsequent expansions to fit as multiples of the **InodeExpandMin** configuration parameter.
[-s]	Forces the allocation to begin on a stripe-aligned block.
[-w]	Sets the file size to the value of the **size** parameter.
[-z]	Causes the file to physically zero out.
<size>[k|m|g|t]	Specifies the file size in bytes. If appended with **k**, **m**, **g**, or **t**, the value represents kilobytes, megabytes, gigabytes, or terabytes. **Caution:** Do not include any spaces between size value and the suffix (**k**, **m**, **g**, or **t**).
<filename>	Name of the file to create.

Examples

- Make a 1GB file with 0 (zero) length. Allocate a stripe group that favors the media type **6100_n8**.

    cvmkfile -k 6100_n8 1g samplefile

cvmkfs

The **cvmkfs** command is used to initialize a file system and (optional) to specify the file system name. If a file system name is not provided, a list of configured file systems is presented. Active file systems may not be re-initialized.

Caution: Use the **cvmkfs** command with extreme caution. Executing this command will destroy any existing file system data for the named file system.

Command Syntax

cvmkfs [-G] [-F] <file_system_name>

Options / Arguments

Table 9 lists options and arguments for the **cvmkfs** command.

Table 9 Options and
Arguments: cvmkfs

Options / Arguments	Description
[-G]	Bypasses **"Press return to continue..."** type prompts.
[-F]	Executes the **cvmkfs** command in non-interactive mode. The user does not receive confirmation prompts. **Warning:** Executing the **cvmkfs** command without prompts results in an irreversible re-initialization of the file system. Use extreme caution when invoking this option.
<file_system_name>	Name of the file system to reinitialize.

Examples

• Re-initialize a StorNext file system.

   /usr/cvfs/bin/cvmkfs <file_system_name>

cvupdatefs

The **cvupdatefs** command is used to update the configuration of a file system. This allows you to add stripe groups or expand current stripe groups to expand the storage space available to a file system.

Note:	All metadata modifications must be made on an inactive (stopped) file system.

Command Syntax

cvupdatefs [-F] [-G] [-h] [-l] [-n] [-v] [-c <pathname>] [<FsName>]

Options / Arguments

Table 10 lists options and arguments for the **cvupdatefs** command.

Table 10 Options and
Arguments: cvupdatefs

Options / Arguments	Description
[-F]	Forces updates without prompting for confirmation.
[-G]	Pauses the program after displaying the exit status (Windows only).
[-h]	Displays command help.
[-l]	Logs when the update finished.
[-n]	Sets metadata to read-only mode.
[-v]	Turns on verbose reporting.
[-c <pathname>]	Use the specified file as the original configuration and run **cvfsck** before applying updates from the new configuration file.
[<FsName>]	Name of the file system to update.

cvversions

The **cvversions** command is used to display different versions (revision, build level and creation date) for the File System Manager and client sub-systems of SNFS. If you contact the Quantum Technical Assistance Center, provide this information.

Command Syntax

cvversions

Options / Arguments

None

Examples

- Display version information for a StorNext file system.

    cvversions

Storage Manager Commands

This section describes the following StorNext Storage Manager commands:

Note: Storage Manager commands are available only on Linux MDC systems.

---

fsaddclass

The **fsaddclass** command is used to create and define a new policy class.

Command Syntax

fsaddclass <class> [-s <softlimit>] [-h <hardlimit>] [-S <stubsize>]
[-t <mediatype>] [-l <securitycode>] [-o <acctnum>]
[-x <maxcopies>] [-d <defaultcopies>] [-m <minstoretime>]
[-c <mintrunctime>] [- a <affinity>] [-i <minreloctime>]
[-R <affinity>] **[-v <drivepool>] [-k <maxversions>] [-f i | p]**
[-r c | s] [-p yes | no] [-z <minsetsize> [MB | GB] -g <maxsetage>]
[-G y | n] [-V y | n] [-D y | n]

Caution: The file system must be mounted and active before running this command or it will fail.

Options / Arguments

[Table 11](#) lists options and arguments for the **fsaddclass** command.

Table 11 Options and Arguments: fsaddclass

Options / Arguments	Description
<class>	The policy class. (Maximum of 16 characters.)
[-s <softlimit>]	Warning limit for the number of media allocated in the policy class. Default: 20,000

Options / Arguments	Description
[-h <hardlimit>]	Maximum number of media allowed in the policy class. Default: 25,000
[-S <stubsize>]	The truncation stub size (in KB). This is the minimum number of bytes to leave on disk when files are truncated. Default: 0
[-t <mediatype>]	Defines the media type to use for the policy class. Valid values: • **AIT** • **AITW** • **LTO** • **LTOW** • **SDISK** • **3590** • **3592** • **9840** • **9940** • **T10K** • **DLT4** Default: **CLASS_DEF_MEDIA_TYPE**
[-l securitycode>]	Security code for the policy class. (Maximum of four characters.) Default: **NONE**
[-o <acctnum>]	Account number for the policy class. (Maximum of five characters.) Default: 12345
[-x <maxcopies>]	Maximum number of copies allowed for any file associated with the policy class. Default: 4

Options / Arguments	Description
[-d <defaultcopies>]	Total number of copies allowed for each file associated with the policy class. This value cannot exceed the **maxcopies** setting. Default: 1
[-m <minstoretime>]	Minimum time (in minutes) that a file must reside unmodified on disk before being considered a candidate for storage on media. Default: 5
[-c <mintrunctime>]	Minimum time (in days) that a file must reside unaccessed on disk before being considered a candidate for truncation. Default: 3
[-a <affinity>]	List of disk affinities that the files in this policy class will traverse in their life cycle. Default: '' (empty list)
[-i <minreloctime>]	Minimum time (in days) that a file must reside unaccessed on disk before being considered a candidate for relocation. Default: 7
[-R <affinity>]	The affinity to retrieve a truncated file to (overrides the default affinity).
[-v <drivepool>]	Media Manager drive pool used to store or retrieve data. Default: **fs_F0drivepool**
[-k <maxversions>]	Maximum number of inactive versions to keep for a file. Minimum: 1. Maximum: 25. Default: 10.

Options / Arguments	Description
[-f i | p]	File retention policy for the policy class. Files can be: • **i** - Truncated immediately • **p** - (Default) Truncated at policy application time
[-r c | s]	Media classification cleanup action. When all files are deleted from a medium, the medium can revert back to: • **c** - Policy class blank pool • **s** - (Default) System blank pool
[-p yes | no]	Specifies whether or not (**yes** or **no**) to allow the policy engine to automatically store files for the policy class. (If set to **no**, use **fsschedule** to schedule when the policy is run.) Default: **yes**
[-z <minsetsize> [MB | GB] -g <maxsetage>]	Sets the minimum set size of the policy's data to store (MB or GB) and the candidate expiration time to **maxsetage** (hours).
[-G y | n]	Specifies whether or not (**y** or **n**) to generate and maintain a checksum for each stored file. Default: **n**
[-V y | n]	Specifies whether or not (**y** or **n**) to verify the checksum of each retrieved file. Default: **n**
[-D y | n]	Specifies whether or not (**y** or **n**) to remove database information when a file is removed. If enabled, the file cannot be recovered with **fsrecover**.

fsaddrelation

The **fsaddrelation** command is used to associate a directory with a policy class.

Command Syntax

fsaddrelation <directory> -c <class>

Options / Arguments

Table 12 lists options and arguments for the **fsaddrelation** command.

Table 12 Options and Arguments: fsaddrelation

Options / Arguments	Description
<directory>	Path name of the directory to associate with the policy class.
-c <class>	Name of the policy class to associate with the directory. The name of the policy class can be up to 16 characters in length and the following special characters can be used: comma (,), period (.), and underscore (_).

Caution: **fsaddrelation** fails if the directory has an affinity associated with it. Use the **cvaffinity** command to remove the affinity before running this command.

Caution: Quantum recommends running this command before adding any files or subdirectories to the directory. If the directory has subordinate directories when you run the **fsaddrelation** command, the subordinate directories retain their current affinity association. In this case, any new files created in those directories can result in allocations to unexpected stripe groups.

> **Note:** After the **fsaddrelation** command completes successfully, if the class has any affinities, the directory has its affinity set to the first affinity in the class affinity list.

Running fsaddrelation under Linux

If the operating system is Linux, do not run the **fsaddrelation** command from the SNFS parent directory or directories of the relation **<directoryName>**. For Linux:

1 Create a subdirectory under the file system mount point where data will be created. For example:

   mkdir /stornext/snfs1/data1

2 Change the permissions of the directory to read, write, execute for world, group, and user. For example:

   chmod 777 /stornext/snfs1/data1

3 Create the association between the policy class and the directory. First, change directory. Type:

   cd /stornext

4 Then run the **fsaddrelation** command. Be sure to provide the full pathname of the relation **<directoryName>**. For example:

   fsaddrelation -c policyclass1 /stornext/snfs1/dir20041021

fschdiat

The **fschdiat** command is used to modify the policy class attributes of a directory.

Command Syntax

fschdiat [-s e | p] [-r e | p] [-t e | p] [-D y | n] [-c <class>] [-S <stubsize>] <directoryname>

Options / Arguments

Table 13 lists options and arguments for the **fschdiat** command.

Options / Arguments	Description
[-s e | p]	Indicates how storage policies (**-s**) operate on the directory: • **e** - Excludes files from truncation when a store policy occurs. • **p** - Stores files by storage policy.
[-r e | p]	Indicates how relocation policies (**-r**) operate on the directory: • **e** - Excludes files from truncation when a relocation policy occurs. • **p** - Stores files by relocation policy.
[-t e | p]	Indicates how truncation policies (**-t**) operate on the directory: • **e** - Excludes files from truncation when a store or cleanup policy application occurs. • **p** - Truncates files by cleanup policy.
[-D y | n]	Indicates if database entries are to be cleaned when the file is removed from the file system. • **y** - Database entries are cleaned and the file is not recoverable using **fsrecover**. • **n** - Database entries are not cleaned and the file is recoverable using **fsrecover**.
[-c <class>]	Specifies the policy class associated with the directory.
[-S <stubsize>]	The truncation stub size (in KB). This is the minimum number of bytes to leave on disk when files are truncated.

Options / Arguments	Description
<directoryname>	The directory to change class attributes for. Separate multiple directory names with a space.

fschfiat

The **fschfiat** command is used to modify the policy class attributes of a file.

Command Syntax

fschfiat [-s e | p] [-r e | p] [-t e | p | i] [-D y | n] [-c <copies>] [-a <class>] [-S stubsize] <filename>

Options / Arguments

Table 14 lists options and arguments for the **fschfiat** command.

Table 14 Options and Arguments: fschfiat

Options / Arguments	Description
**[-s e	p]**	Indicates how storage policies (**-s**) operate on the file: • **e** - Excludes the file from truncation when a store policy occurs. • **p** - Stores the file by storage policy.
**[-r e	p]**	Indicates how relocation policies (**-r**) operate on the file: • **e** - Excludes the file from truncation when a relocation policy occurs. • **p** - Stores the file by relocation policy.

Options / Arguments	Description
[-t e | p | i]	Indicates how truncation policies (**-t**) operate on the file: • **e** - Excludes the file from truncation when a store or cleanup policy application occurs. • **p** - Truncates the file by cleanup policy. • **i** - Truncates the file immediately.
[-D y | n]	Indicates if database entries are to be cleaned when the file is removed from the file system. • **y** - Database entries are cleaned and the file is not recoverable using **fsrecover**. • **n** - Database entries are not cleaned and the file is recoverable using **fsrecover**.
[-c <copies>]	Number of file copies to store (including the primary copy). This value cannot exceed the policy's **maxcopies** setting.
[-a <class>]	Specifies an alternate policy class associated with the file.
[-S <stubsize>]	The truncation stub size (in KB). This is the minimum number of bytes to leave on disk when files are truncated.
<filename>	File to change class attributes for. Separate multiple file names with a space.

fschmedstate

The **fschmedstate** command is used modify the state of media or the policy class to which the media belongs.

Command Syntax

fschmedstate <mediaID> [-c <class>]

fschmedstate <mediaID> [-b]

fschmedstate <mediaID> [-s <state>]

Options / Arguments

Table 15 lists options and arguments for the **fschmedstate** command.

Table 15 Options and
Arguments: fschmedstate

Options / Arguments	Description
<mediaID>	One or more media identifiers. Separate multiple media IDs with a space.
[-c <class>]	Change the policy class name of blank media to **<class>**.
[-b]	Change the policy class of blank media to system blank pool.
[-s <state>]	The media state. Valid values are: • **unsusp** - Resets media error count to 0. • **protect** - Marks media as write protected. • **unprotect** - Marks media as unprotected. • **avail** - Marks media as available. • **unavail** - Marks media as unavailable. • **unmark** - Unmark media that is marked **Error** or **Check Out**.

fschstate

The **fschstate** command is used to modify the state of all storage subsystems and drive components.

Command Syntax

fschstate <componentalias> -s <state>

Options / Arguments

<u>Table 16</u> lists options and arguments for the **fschstate** command.

Options / Arguments	Description
<componentalias>	Component alias name used to identify storage subsystems and drive components. During system configuration, the System Administrator specifies the values of the **componentalias** parameter.
-s <state>	Desired state of drive components or subsystems. Valid values (drive components): • **MAINT** • **ON** • **OFF** Valid values (subsystems): • **ON** • **OFF**

fsclassinfo

The **fsclassinfo** command is used to report policy class and, within each group, its processing parameters and directory paths.

Command Syntax

fsclassinfo [-l] [<class> ...]

Options / Arguments

<u>Table 17</u> lists options and arguments for the **fsclassinfo** command.

Table 17 Options and Arguments: fsclassinfo	Options / Arguments	Description
	[-l]	Display the long report format, including processing parameters, directory paths, and the disk affinity list for the policy class.
	[<class> ...]	The policy class or classes to be listed. Separate multiple policy classes with a space. If no class is specified, a short report for all policy classes is displayed.

fsclassrnm

The **fsclassrnm** command is used to rename an existing policy class. This causes all files and media that were in the old class to be assigned to the new class.

Command Syntax

fsclassrnm <oldclass> <newclass>

Options / Arguments

Table 18 lists options and arguments for the **fsclassrnm** command.

Table 18 Options and Arguments: fsclassrnm	Options / Arguments	Description
	<oldclass>	The name of the policy class to be changed. (Maximum of 16 characters.)
	<newclass>	The name of the new policy class. (Maximum of 16 characters.)

fsclean

The **fsclean** command is used to remove inactive file versions.

Command Syntax

fsclean -m <mediaID> [-t [<endtime>]]

fsclean -s <filesystem> [-t [<endtime>]]

fsclean -c <class> [-t [<endtime>]]

fsclean -t [<endtime>]

fsclean -r [-l]

fsclean -r [<mediaID>] -p <filename>

Table 19 lists options and arguments for the **fsclean** command.

Table 19 Options and
Arguments: fsclean

Options / Arguments	Description
-m <mediaID>	The media identifier(s) to clean. Separate multiple media IDs with a space.
-s <filesystem>	The file system to clean. Separate multiple file systems with a space.
-c <class>	The policy class associated with the data to clean. Separate multiple policy classes with a space.
[-t <endtime>]>	Endtime option. The value of this parameter should be less than or equal to the current time. Format is: **YYYY:MM:DD:hh:mm:ss** The default is the current time (all inactive versions removed for all times).
-r [<mediaID>]	Cleans all Tertiary Manager knowledge of files on media where **rminfo** processing has been done. If a **mediaID** is specified, then processing is restricted to only that media.

Options / Arguments	Description
[-l]	Valid only with **-r** option. Displays a list of all media where **rminfo** processing has been done.
-p <filename>	Valid only with **-r** option. The file **filename** contains a list of files that need to be retrieved so missing copies can be regenerated.

fsdirclass

The **fsdirclass** command is used to display the policy class associated with the specified directory.

Command Syntax

fsdirclass <directory>

Options / Arguments

Table 20 lists options and arguments for the **fsdirclass** command.

Table 20 Options and Arguments: fsdirclass

Options / Arguments	Description
<directory>	The directory path for which the associated policy class is returned. If preceded by a slash (/) provide the full path. Otherwise the path is resolved from the current directory. (The path must be less than 256 characters.)

fsdrvclean

The **fsdrvclean** command is used to clean the specified drive.

Command Syntax

fsdrvclean <drivealias>

Options / Arguments

<u>Table 21</u> lists options and arguments for the **fsdrvclean** command.

Options / Arguments	Description
<drivealias>	The alias of the drive to clean. A drive alias is a variable string of up to 255 characters. If the drive alias contains spaces, use single quotes around the string. (Use the **fsstate** command to list the set of drive aliases.)

fsfilecopy

The **fsfilecopy** command is used to replace the primary copy (or a specified copy) of a file on a piece of media by duplicating it on another piece of media.

Command Syntax

fsfilecopy -r <filename> -d <destinationmediaID> [-c <copynum>]

fsfilecopy -r <filename> [-b] [-t <mediatype>] [-c <copynum>]

Options / Arguments

<u>Table 22</u> lists options and arguments for the **fsfilecopy** command.

Table 22 Options and
Arguments: fsfilecopy

Options / Arguments	Description
-r	Copies the file(s) **filename** to a piece of media and deletes all corresponding file(s) on the original piece of media.
<filename>	The path of the file(s) to replace. If preceded by a slash (/) provide the full path. Otherwise the path is resolved from the current directory. Separate multiple file names with a space.
-d <destinationmediaID>	Specifies the piece of media to which the file(s) are copied.
[-c <copynum>]	Specifies the copy of the file(s) to replace. (If not specified, the primary copy is replaced.)
[-b]	File(s) are copied to a blank piece of media.
[-t <mediatype>]	Defines the type of media to be used for replacement. Valid values: • **AIT** • **AITW** • **LTO** • **LTOW** • **SDISK** • **3590** • **3592** • **9840** • **9940** • **T10K** • **DLT4**

fsfileinfo

The **fsfileinfo** command is used to report the current location of files, whether on disk, media, or not in the system.

Command Syntax

fsfileinfo [-c] <filename>

Options / Arguments

Table 23 lists options and arguments for the **fsfileinfo** command.

Table 23 Options and Arguments: fsfileinfo

Options / Arguments	Description
<filename>	Path and filename of the file(s) to report on. If preceded by a slash (/) provide the full path. Otherwise the path is resolved from the current directory. Separate multiple file names with a space.
[-c]	Displays the checksum value generated for the file when stored.

fsmedcopy

The **fsmedcopy** command is used to duplicate the contents of non-blank media. Media duplication for replacement is intended for:

- Media maintenance - if errors occur frequently when attempting to read or write to the media.

- Media defragmentation - if the media is filled with unusable space due to deleted or modified files.

Command Syntax

fsmedcopy <mediaID> -r [-a] [-d <destinationmediaID>] [-u <runtime>]

fsmedcopy <mediaID> -r [-a] [-b] [-t <mediatype>] [-u <runtime>]

fsmedcopy [<mediaID>] [-f <fill >] [-w <fragmentation>]

Options / Arguments

Table 24 lists options and arguments for the **fsmedcopy** command.

Table 24 Options and
Arguments: fsmedcopy

Options / Arguments	Description
<mediaID>	The media identifier(s) to copy. Separate multiple media IDs with a space.
-r	Initiates copy-replace for the media.
[-a]	Copies all files (active and inactive versions) from the source media.
[-d <destinationmediaID>]	Copies data file(s) to the destination media identifier.
[-b]	Specifies blank media for media destination.
[-f <fill>]	Specifies the fill level threshold (between 0 and 100%). The value of the **fill** parameter is the percentage of the media that has been written, including active and inactive file versions.

Options / Arguments	Description
[-t <mediatype>]	Defines the media type to be used. Valid values: • **AIT** • **AITW** • **LTO** • **LTOW** • **3590** • **3592** • **9840** • **9940** • **T10K** • **DLT4**
[-u <runtime>]	Maximum allowable time (in hours) for the **fsmedcopy** command to finish.
[-w <fragmentation>]	The percentage (0 to 100%) of filled space on the media that is wasted (fragmented) space. The percentage is based on the amount of filled media space, not the total capacity of the medium. If not specified, the **fragmentation** parameter is set to zero (0) percent.

fsmedinfo

The **fsmedinfo** command is used to generate a report on the status of specified media.

Command Syntax

fsmedinfo [-l] <mediaID>

Options / Arguments

Table 25 lists options and arguments for the **fsmedinfo** command.

Table 25 Options and Arguments: fsmedinfo	Options / Arguments	Description
	[-l]	Generates a long format report, including a list of files on the media.
	<mediaID>	The media identifier(s) to report on. Separate multiple media IDs with a space.

fsmedlist

The **fsmedlist** command is used to generate a list of media based on media status and policy class.

Command Syntax

fsmedlist [-c <class>] [-g] [-l [k] [m]]

fsmedlist [-c <class>] [-g] [-l [b] [f] [p] [q] [a] [n] [u] [s]]

fsmedlist [-c <class>] [-g] [-l [d] [h] [t] [z] [o]]

Options / Arguments

Table 26 lists options and arguments for the **fsmedlist** command.

Table 26 Options and Arguments: fsmedlist	Options / Arguments	Description
	[-c <class>]	Policy class(es) for which the report is to be generated. Separate multiple policy classes with a space.
	[-g]	Reports on blank media in the general scratch pool.
	[-l]	Lists media in long report format.
	[-l k]	Lists all media marked for check out.
	[l -m]	Lists all media marked for removal.
	[-l b]	Lists unformatted blank media.

Options / Arguments	Description
[-l f]	Lists formatted blank media.
[-l p]	Lists all write-protected media.
[-l q]	Lists media with mark error.
[-l a]	Lists all available media.
[-l n]	Lists all media unavailable to the StorNext software.
[-l u]	Lists all media marked as unavailable but still located in the StorNext system.
[-l s]	Lists all suspect media.
[-l d]	Lists all media located in a drive.
[-l h]	Lists all media located in its home slot/bin.
[-l t]	Lists all media transitioning between locations.
[-l z]	Lists all media exiting a storage area.
[-l o]	Lists all media checked out of storage areas.

fsmodclass

The **fsmodclass** command is used to modify the processing parameters of a policy class.

Command Syntax

fsmodclass <class> [-s <softlimit>] [-h <hardlimit>] [-S <stubsize>]
[-t <mediatype>] [-l <securitycode>] [-o <acctnum>]
[-x <maxcopies>] [-d <defaultcopies>] [-m <minstoretime>]
[-c <mintrunctime>] [- a <affinity>] [-i <minreloctime>]
[-R <affinity>] [-v <drivepool>] [-k <maxversions>] [-f i | p] [-r c | s] [-p yes | no] [-z <minsetsize> [MB | GB] -g <maxsetage>

] [-G y | n]
[-V y | n] [-D y | n]

Options / Arguments

Table 27 lists options and arguments for the **fsmodclass** command.

Options and Arguments: fsmodclass

Options / Arguments	Description
<class>	The policy class.
[-s <softlimit>]	Warning limit for the number of media allocated in the policy class.
[-h <hardlimit>]	Maximum number of media allocated in the policy class.
[-S <stubsize>]	The truncation stub size (in KB). This is the minimum number of bytes to leave on disk when files are truncated.
[-t <mediatype>]	Defines the media type to use for the policy class. Valid values: • **AIT** • **AITW** • **LTO** • **LTOW** • **SDISK** • **3590** • **3592** • **9840** • **9940** • **T10K** • **DLT4**
[-l <securitycode>]	Security code for the policy class (up to 4 characters in length).

Options / Arguments	Description
[-o <acctnum>]	Account number for the policy class (up to 5 characters in length).
[-x <maxcopies>]	Maximum number of copies allowed for any file associated with the policy class.
[-d <defaultcopies>]	Total number of copies allowed for each file associated with the policy class.
[-m <minstoretime>]	Minimum time (in minutes) that a file must reside unmodified on disk before being considered a candidate for storage on media.
[-c <mintrunctime>]	Minimum time (in days) that a file must reside unaccessed on disk before being considered a candidate for truncation.
[-a <affinity>]	List of disk affinities that the files in the policy class will traverse in their life cycle.
[-i <minreloctime>]	Minimum time (in days) that a file must reside unaccessed on disk before being considered a candidate for relocation.
[-R <affinity>]	The affinity to retrieve a truncated file to (overrides the default affinity).
[-v <drivepool>]	Media Manager drive pool used to store or retrieve data.
[-k <maxversions>]	Maximum number of inactive versions to keep for a file. (The minimum is 1 and the maximum is 25. The default is 10.)

Options / Arguments	Description
[-f i | p]	File retention policy (**-f**) for the policy class. Files can be: • **i** - Truncated immediately • **p** - Truncated at policy application time
[-r c | s]	Media classification cleanup action. When all files are deleted from a medium, the medium can revert (**-r**) back to: • **c** - Policy class blank pool • **s** - System blank pool
[-p yes | no]	Specifies whether or not (**yes** or **no**) to allow the policy engine to automatically store files for the policy class. (If set to **no**, use **fsschedule** to schedule when the policy is run.)
[-z <minsetsize> [MB | GB] -g <maxsetage>]	Sets the minimum set size of the policy's data to store (MB or GB) and the candidate expiration time to **maxsetage** (hours).
[-G y | n]	Specifies whether or not (**y** or **n**) to generate and maintain a checksum for each stored file.
[-V y | n]	Specifies whether or not (**y** or **n**) to verify the checksum of each retrieved file.
[-D y | n]	Specifies whether or not (**y** or **n**) to remove database information when a file is removed. If enabled, the file cannot be recovered with **fsrecover**.

fspolicy

The **fspolicy** command is used to manage files on a file system controlled by the StorNext software. It manages these files by applying a combination of the following:

• The storage, relocation, and truncation parameters defined by the policy class.

• The file system configuration parameters.

A set of file candidates for storing, relocating, and truncating are kept up to date by the StorNext software, which tracks file system activity. When the **fspolicy** command is executed (manually or by the StorNext software), a group of those files meeting the criteria will be stored, relocated, or truncated as needed.

Command Syntax

fspolicy -s [-c <class>] [-v <drivepool>] [-m <minstoretime>]

fspolicy -r [-y <filesystem>] [-e] [-o <goal>] [-m <minreloctime>]
[-z <minsize>] [-a <affinity>]

fspolicy -t [-y <filesystem>] [-e] [-o <goal>] [-m <mintrunctime>]
[-z <minsize>] [-a <affinity>]

fspolicy -s [-e] [-y <filesystem>] [-v <drivepool>]
[-m <minstoretime>]

fspolicy -r [-c <class>] [-m <minreloctime>] [-z <minsize>]

fspolicy -t [-c <class>] [-o <goal>] [-m <mintrunctime>]
[-z <minsize>]

fspolicy -b [-y <filesystem>]

Options / Arguments

Table 28 lists options and arguments for the **fspolicy** command.

Table 28 Options and
Arguments: fspolicy

Options / Arguments	Description
-s	Invokes the storage policy.
-r	Invokes the relocation policy.
-t	Invokes the cleanup (truncation) policy.

Options / Arguments	Description
-b	Scans file system for candidate files to store or truncate and rebuilds the candidate lists.
[-y <filesystem>]	File system to which the policy will be applied.
[-e]	Invokes a high priority for immediate action on file storage in a file system.
[-v <drivepool>]	Specifies the drive pool from which to allocate drives when storing files.
[-m <minstoretime>]	Minimum time (in minutes) that a file must reside on disk before being considered a candidate for storage.
[-o <goal>]	Percentage of used disk space at which a relocation or truncation policy ceases to be applied.
[-m <minreloctime>]	Minimum time (in minutes) that a file must reside on disk before being considered a candidate for relocation.
[-z <minsize>]	Minimum file size (in bytes) to relocate or truncate.
[-a <affinity>]	List of disk affinities that the files in this policy class will traverse in their life cycle.
[-m <mintrunctime>]	Minimum time (in days) that a file must reside unaccessed on disk before being considered a candidate for truncation.
[-c <class>]	Policy class associated with data to be stored or truncated.

fsqueue

The **fsqueue** command is used to view subsystem resource requests.

Command Syntax

fsqueue [-r <requestID>]

fsqueue -m [-r <requestID>]

fsqueue -f [-r <requestID> | <filename>]

fsqueue -a [-v]

Options / Arguments

Table 29 lists options and arguments for the **fsqueue** command.

Table 29 Options and
Arguments: fsqueue

Options / Arguments	Description
[-r <requestID>]	Identifier of the request to be reported.
-m	Reports media movement for a **requestID** or all media in the queue.
-f	Reports files in the queue or specific files for a **requestID** or **filename**.
-a	Displays information only for active data mover hosts.
[-v]	Displays information only for active data mover requests.
<filename>	Path and file name of the file to be reported. Specify the full path unless the file is located in the current working directory. Separate multiple file names with a space.

Quantum StorNext CLI Reference Guide

fsrecover

The **fsrecover** command is used to report on files that were removed from disk and recover files back to disk. The command only recovers the primary file copy, but once recovered all copies are available.

Command Syntax

fsrecover <filename> [-p] [-t <starttime> [<endtime>]]

fsrecover <dirname> -d [-p] [-r] [-a]

fsrecover [<RM_time>::]<filepathname> -u [-v]

fsrecover <dirpathname> -u -d [-r] [-a] [-v]

Options / Arguments

Table 30 lists options and arguments for the **fsrecover** command.

Table 30 Options and Arguments: fsrecover

Options / Arguments	Description
<filename>	Name of the file to report on. The name can be a file name, a partial path name, or a complete path name. The "%" character is a wild card character. Separate multiple file names with a space. **Note:** Files that are recoverable no longer reside on disk, but are still located in the StorNext database.
[-p]	Specifies that the user report displays all entries for which the user has permissions (file entries to which the user has read access and directory entries to which the user has write access). If this option is not specified, the report to the user displays only user-owned entries.

Options / Arguments	Description
[-t <starttime> [<endtime>]]	Indicates a time range to restrict the length of the user report. The format of the start time and end time parameters is: **YYYY:MM:DD:hh:mm:ss**
<dirname>	Name of the directory containing one or more files to recover. The name can be a directory name, partial path, or full path. The "%" character is a wild card. Separate multiple directory names with a space.
-d	Indicates that directory recovery processing is requested.
[-r]	Indicates recursive processing is requested.
[-a]	Specifies active files rather than recoverable files when either reporting or recovering a directory.
[<RM_time>::] <filepathname>	Full path of the file to recover. Optionally, you can have a timestamp of the form **YYYY:MM:DD:hh:mm:ss::/stornext/ snfs2/rel1/sub1/file.a**. If there are two files with the same paths that were removed at different times, use the timestamp to distinguish between them. When you specify the timestamp, put it with the pathname, as in the report. Separate multiple file names with a space.
-u	Indicates that recovery processing is requested.
[-v]	Specifies the verbose mode during a recover sequence. The system reports on the recovered files.

Options / Arguments	Description
<dirpathname>	Full path of the directory containing the file(s) to recover. Separate multiple directory names with a space.

fsretrieve

The **fsretrieve** command is used to retrieve or recover a file from media and place it on disk. By default, the primary copy of a file is retrieved.

Command Syntax

fsretrieve [-a] [-c <copy>] [-x y | n] <filename>

fsretrieve <filename> -n <newfilename> [-b <startbyte> <endbyte>]
[-a]

fsretrieve <filename> -n <newfilename> [-c <copy>] [-a]

fsretrieve -R <directory> [-a]

Options / Arguments

Table 31 lists options and arguments for the **fsretrieve** command.

Table 31 Options and
Arguments: fsretrieve

Options / Arguments	Description
[-a]	Updates the access time of the requested files.
[-c copy]	Retrieves a specific copy of **filename** if one exists.
<filename>	Full path and filename of the file to retrieve. Separate multiple file names with a space.
-n <newfilename>	New path and name into which to retrieve the file.

Options / Arguments	Description
[-b <startbyte> <endbyte>]	Specifies a byte range to retrieve. The value of **startbyte** must be less than the **endbyte** value. Both values must be within the byte range of the file.
-R <directory>	Directory from which to start recursive retrieval.
[-x y | n]	Forces a change in the alternate retrieval location behavior. • **y** - The alternate location is used, bypassing the standard copies. • **n** - The alternative location is ignored.

fsrmcopy

The **fsrmcopy** command is used to remove a copy or copies of a file from disk or media.

Command Syntax

fsrmcopy [-c <copynum> | -a] <filename>

Options / Arguments

Table 32 lists options and arguments for the **fsrmcopy** command.

Table 32 Options and Arguments: fsrmcopy

Options / Arguments	Description
[-c <copynum> | -a]	Deletes either a specific number of file copies (**- c <copynum**>) or all file copies (**-a**) from the media.

Options / Arguments	Description
<filename>	The file(s) to remove from disk or media. If preceded by a slash (/) provide the full path. Otherwise the path is resolved from the current directory. Separate multiple file names with a space.

fsrmdiskcopy

The **fsrmdiskcopy** command removes a file copy or copies from disk after the file is stored to media. All Storage Manager copies required for the file must be made before the disk space can be freed up. If the file is also replicated or deduplicated by SNFS, the required SNFS operations must have been completed.

Command Syntax

fsrmdiskcopy <filename>

Options / Arguments

Table 33 lists options and arguments for the **fsrmdiskcopy** command.

Table 33 Options and Arguments: fsrmdiskcopy

Options / Arguments	Description
<filename>	The file(s) to remove from disk. If preceded by a slash (/) provide the full path. Otherwise the path is resolved from the current directory. Separate multiple file names with a space.

fsrminfo

The **fsrminfo** command is used to remove information about all files on a piece of media from the StorNext database.

Caution: Use the **fsrminfo** command with extreme caution. Executing this command removes all knowledge of the files on the media from StorNext. If files are disk-resident (not truncated), they will be restored after the next scheduled **fsclean -r** command following the **fsrminfo** command that is run. (The **fsclean** command is run by default on a weekly basis.). However, if the files are truncated, the files will be lost and will not be recoverable.

Command Syntax

fsrminfo <mediaID> [-y]

Options / Arguments

Table 34 lists options and arguments for the **fsrminfo** command.

Table 34 Options and Arguments: fsrminfo

Options / Arguments	Description
<mediaID>	The media identifier of the media from which to remove file information. Separate multiple media IDs with a space.
[-y]	Forces the removal of file information without prompting for confirmation.

fsschedlock

Use the **fsschedlock** command to lock and unlock some automated StorNext features. The **fsschedlock** command can be used to lock a feature for a defined period, unlock a currently locked feature, or list features that are currently locked.

Note: Locking a feature only prevents it from being run automatically. It does not prevent the feature from being run manually.

Command Syntax

fsschedlock [-r]

**fsschedlock [-a | -d] [-t <type>] [-f | [-s <time> [-e <time> |
-p <hrs>]]] [<day>]**

Options / Arguments

Table 35 lists options and arguments for the **fsschedlock** command.

Table 35 Options and
Arguments: fsschedlock

Options / Arguments	Description
[-r]	Reports the current lock status for all features.
**[-a	-d]**	Add (**-a**) or delete (**-d**) a lock from the feature(s) specified with the **-t** option.
[-t <type>]	The type of feature for which to add or delete a lock. Valid options are: • **store** • **reloc** • **foreign** • **rebuild** • **lospace** • **mintime** • **clnver** • **clninfo** • **p_backup** • **f_backup** • **spolicy** • **all**
[-f]	Locks the feature for a full day
[-s <time>]	The time to start the lockout in **HHMM** format. (Uses a 24 hour clock. Midnight is **0000**.)

Options / Arguments	Description
[-e \<time> | -p hrs]	Specify one of the following: • The time to end the lockout in **HHMM** format. (Uses a 24 hour clock. Midnight is **0000**.) • The duration of the lockout period (in hours).
[\<day>]	The day(s) of the week to lock the feature. Valid options are: **sun**, **mon**, **tue**, **wed**, **thu**, **fri**, **sat**, or **all**.

fsschedule

The **fsschedule** command is used to report, insert, modify, delete, or reset scheduled features for the Quantum storage system. By default, **fsschedule** generates a report showing when automated features are scheduled as well as the status of the last run for each feature.

Command Syntax

fsschedule [-f \<feature> | -n \<name>] [-l]

**fsschedule -a -n \<name> -f \<feature> -p \<period> [-e \<weekday> |
-y \<monthday>] -t \<runtime> [-w \<window>] [-o \<option>]**

**fsschedule -m -n \<name> [-p \<period> [-e \<weekday> |
-y \<monthday>]] [-t \<runtime>] [-w \<window>] [-o
\<option>]**

fsschedule -d -n \<name>

fsschedule -r -f \<feature>

Options / Arguments

Table 36 lists options and arguments for the **fsschedule** command.

Table 36 Options and
Arguments: fsschedule

Options / Arguments	Description
-f <feature>	The scheduled feature to work with. Valid options are: **clninfo**, **clnver**, **rebuild**, **p_backup**, **f_backup**, **spolicy**.
-n <name>	The schedule name.
[-l]	Lists schedules in the long report format.
-a	Adds a new schedule. (This option requires the **-n**, **-f**, and **-p** options.)
[-p <period> **[-e <weekday> |** **-y <monthday>]]**	The period of the schedule. Valid options are: **daily**, **weekly**, **monthly**. (This options requires the **-t** option.) • For weekly, you must specify **-e weekday** where valid options are: **sun**, **mon**, **tue**, **wed**, **thu**, **fri**, **sat**. • For monthly, you must specify **-y monthday** where **monthday** is a day of the month (**1–31**).
[-t <runtime>]	The start time of the feature in **HHMM** (hours and minutes).
[-w <window>]	The window or runtime offset in **HHMM** (hours and minutes).
[-o <option>]	If **spolicy** is specified for **-f feature**, use **-o option** where **option** is the name of an existing policy class.
-m	Modifies an existing schedule. (This option requires the **-n** option.)
-d	Deletes an existing schedule. (This option requires the **-n** option.)
-r	Resets all schedules of a feature. (This option requires the **-f** option.)

fsstate

The **fsstate** command is used to generate a report showing the state of drive components, storage subsystems, or the StorNext software.

Command Syntax

fsstate <componentalias>

fsstate -f

Options / Arguments

Table 37 lists options and arguments for the **fsstate** command.

Table 37 Options and
Arguments: fsstate

Options / Arguments	Description
[<componentalias>]	Alias used to identify drives and storage subsystems.
[-f]	Generates a report showing the status of the StorNext software. Valid states are: Active, Not Active, Not Available.

fsstore

The **fsstore** command is used to expedite the storage of data to media, instead of allowing automatic data migration by the StorNext software.

Command Syntax

fsstore <filename> [-t <mediatype>] [-c <copies>] [-f i | p] [-v <drivepool>] [-z <minsize>] [-u <runtime>]

Options / Arguments

Table 38 lists options and arguments for the **fsstore** command.

Options / Arguments	Description
\<filename\>	Name of file on disk to store to media. If preceded by a slash (/) provide the full path. Otherwise the path is resolved from the current directory. Separate multiple file names with a space.
[-t \<mediatype\>]	Defines the media type to use for storage. Valid values: • **AIT** • **AITW** • **LTO** • **LTOW** • **SDISK** • **3590** • **3592** • **9840** • **9940** • **T10K** • **DLT4**
[-c \<copies\>]	Number of file copies to store, including the primary copy.
[-f i | p]	File retention policy (**-f**) for the file. Files can be: • **i** - Truncated immediately • **p** - Truncated at policy application time
[-v \<drivepool \>]	Specifies the Media Manager drive pool group used to store the file.
[-z \<minsize\>]	Specifies the minimum file size (in bytes) to store.

Options / Arguments	Description
[-u <runtime>]	Specifies the maximum time (in hours) allowed for the command to finish.

fsusedspace

The **fsusedspace** command is used to report the total amount of primary copy data stored on media in the storage subsystem. This value is displayed in gigabytes (GB).

Command Syntax

fsusedspace

Options / Arguments

None

fsversion

The **fsversion** command is used to generate a report of valid versions of a file, or to change the current version.

Command Syntax

fsversion -h

fsversion [-a] [-v] <name>

fsversion -c <ver> [-f] <name>

Options / Arguments

Table 39 lists options and arguments for the **fsversion** command.

Table 39 Options and Arguments: fsversion

Options / Arguments	Description
-h	Displays command usage.

Options / Arguments	Description
[-a]	Lists all available versions of the file (current and inactive).
[-v]	Verbose listing of the file (version number and modification time).
<name>	File (file name, partial path, or full path) for which the version number will be checked. Separate multiple file names with a space.
-c <ver>	Modifies the current version of the file to the specified inactive version.
[-f]	Forces the change operation if the current version is not stored on tape. **Warning:** This prevents recovery of the data that is currently on disk. Before using this option, store the current version using **fsstore** or make a copy using the operating system.

snbackup

The **snbackup** command is used to generate a backup of elements of a StorNext Storage Manager file system, including configuration information, database information, and file system metadata. These elements can be restored at a later time using the **snrestore** command.

Command Syntax

snbackup [-p] [-s] [-h] [-n]

Options / Arguments

Table 40 lists options and arguments for the **snbackup** command.

Table 40 Options and
Arguments: snbackup

Options / Arguments	Description
[-p]	Creates a partial backup (journal files and file system metadata).
[-s]	Gets status for the currently running backup or results of the previous backup.
[-h]	Displays command usage.
[-n]	Does not create gzipped metadata dump file copies and ignores existing ones.

snbkpreport

The **snbkpreport** command is used to list available StorNext Storage Manager backups which can be used for a restore operation.

Command Syntax

snbkpreport [-h]

Options / Arguments

Table 41 lists options and arguments for the **snbkpreport** command.

Table 41 Options and
Arguments: snbkpreport

Options / Arguments	Description
[-h]	Displays command usage.

sndpscfg

The **sndpscfg** command is used to configure a distributed LAN server on Linux.

Command Syntax

sndpscfg -e

sndpscfg -E <file_system>

sndpscfg -a

Options / Arguments

Table 42 lists options and arguments for the **sndpscfg** command.

Options / Arguments	Description
-e	Allows the default distributed LAN server configuration file to be edited. (The file is created if it does not already exist.)
-E <file_system>	Allows the distributed LAN server configuration file for the specific file system to be edited. (The file is created if it does not already exist.)
-a	Displays a template of a distributed LAN server configuration file on standard output

Examples

• Edit the default distributed LAN server configuration file:

    sndpscfg -e

snpolicy

The **snpolicy** command is used to view and administer StorNext policy and event information. For more information, vew the **snpolicy** man page (see)Viewing man Pages on page 2

snrestore

The **snrestore** command is used to restore elements of a StorNext Storage Manager file system, including configuration information, database information, and file system metadata.

Command Syntax

**snrestore [-p <temp_path>] [-e] [-r <restore_path>]
[-m | -mj [<file_system_name>]] [-d | -dj] [-c] [-h]**

Options / Arguments

Table 43 lists options and arguments for the **snrestore** command.

Options and
Arguments: snrestore

Options / Arguments	Description
[-p <temp_path>]	Specifies a temporary storage directory for restored files. By default, this directory is: **/usr/adic/ TSM/tmp**
[-e]	Extracts files to the temporary storage directory.
[-r <restore_directory_path>]	Specifies the directory for files to restore. No files are restored from media.
**[-m	-mj [<file_system_name>]]**	Specifies the file system metadata restore values. You can specify: • **m** - Restore software configuration information for all file systems or for the specified file system. • **mj** - Restore file system journal information for all file systems or for the optional specified file system.
**[-d	-dj]**	Specifies the database restore values. You can specify: • **d** - Restore database and journal files. • **dj** - Restore only the journal files and apply them to the existing database on the system.

Options / Arguments	Description
[-c]	Specifies the software files to restore. You can specify: • **c** - Restore all software configuration information.
[-h]	Displays command usage.

Examples

• Restore all components from backup located on tape media:

  snrestore

• Restore database from backup located on tape media and use the **/tmp** temporary directory:

  snrestore -d -p /tmp

• Restore database journals from backup located on tape media.

  snrestore -dj

• Restore from a backup file located in the **/backup** directory.

  snrestore -r /backup

• Restore file system data for **snfs1** file system.

  snrestore -m snfs1

vsarchiveqry

The **vsarchiveqry** command is used to query for information about an archive.

Command Syntax

vsarchiveqry <archivename> | -a [-c] [-d] [-m] [-t] [-v] [-I] [-h]
[-H <hostname>] [-P priority] [-R retries] [-T <timeout>]
[-V <number>]

Options / Arguments

[Table 44](#) lists options and arguments for the **vsarchiveqry** command.

Options / Arguments	Description
<archivename> | -a	Specifies the archive (**archivename**) to be queried or that all archives (**-a**) will be queried.
[-c]	Indicates that detailed information on all media classes associated with the specified archive(s) will be reported.
[-d]	Indicates that all drives associated with the specified archive(s) will be reported.
[-m]	Indicates that all media associated with the specified archive(s) will be reported.
[-t]	Indicates that detailed information on all media types associated with the specified archive(s) will be reported.
[-v]	Indicates that all drives, media classes, media, and media types associated with the specified archive(s) will be reported.
[-I]	Indicates that command line options are read from **stdin**.
[-h]	Displays command usage.
[-H <hostname>]	Hostname of the Media Manager server.

Options / Arguments	Description
[-P <priority>]	Execution priority of the entered command. Default: 15 Valid values: 1 to 32
[-R <retries>]	Number of retries that the software attempts if a timeout is received. Default: 3
[-T <timeout>]	Time interval (in seconds) that the software waits for status before returning a timeout. Default: 120
[-V <number>]	RPC program number for the software. Default: 300016

Examples

- Requests that the Media Manager software return the media classes and the media types associated with the **shelf1** archive:

  vsarchiveqry shelf1 -ct

- Requests that the Media Manager software return the drives, media, media classes, and media types associated with the **stage1** archive:

  vsarchiveqry stage1 -v

vsdriveqry

The **vsdriveqry** command queries for information about the specified drive.

Command Syntax

vsdriveqry <driveID> | -a [-I] [-h] [-H <hostname>] [-P <priority>]
[-R <retries>] [-T <timeout>] [-V <number>]

Options / Arguments

Table 45 lists options and arguments for the **vsdriveqry** command.

Options / Arguments	Description
<driveID> | -a	Indicates a specific drive (**driveID**) or all drives (**-a**) to be queried.
[-I]	Indicates that the command line options are to be read from **stdin**.
[-h]	Displays command usage.
[-H <hostname >]	Hostname of the Media Manager server.
[-P <priority>]	Execution priority of the entered command. Default: 15 Valid values: 1 to 32
[-R <retries>]	Number of retries that the software attempts if a timeout is received. Default: 3
[-T <timeout>]	Time interval (in seconds) that the software waits for status before returning a timeout. Default: 120
[-V <number>]	RPC program number for the software. Default: 300016

Examples

- Requests that the Media Manager software return information on every drive known to the Media Manager system:

    vsdriveqry -a

vsdrivevary

The **vsdrivevary** command is used to change (vary) the operational availability state of a drive.

Command Syntax

**vsdrivevary <driveID> -s state [-v] [-I] [-h] [-H <hostname>]
[-P <priority>] [-R <retries>] [-T <timeout>] [-V <number>]**

**vsdrivevary -p <drivepool> -s <state> [-v] [-I] [-h] [-H
<hostname>]
[-P <priority>] [-R <retries>] [-T <timeout>] [-V <number>]**

Options / Arguments

Table 46 lists options and arguments for the **vsdrivevary** command.

Table 46 Options and
Arguments: vsdrivevary

Options / Arguments	Description
<driveID>	Specifies the drive(s) whose state is to be varied.
-s <state>	Specifies the target state of the specified drive.
[-v]	Verbose listing (status information) of varied drives.
[-I]	Indicates that command line options are read from **stdin**.
[-h]	Displays command usage.
[-H <hostname >]	Hostname of the Media Manager server.
[-P <priority>]	Execution priority of the entered command. Default: 15 Valid values: 1 to 32

Options / Arguments	Description
[-R <retries>]	Number of retries that the software attempts if a timeout is received. Default: 3
[-T <timeout>]	Time interval (in seconds) that the software waits for status before returning a timeout. Default: 120
[-V <number>]	RPC program number for the software. Default: 300016
-p <drivepool>	Specifies the drivepool for which the state of associated drives will be varied.

Examples

- Requests that the Media Manager software vary all drives in the **drvpoolusr** drive pool to the offline state and to return status on every drive in **drvpoolusr**:

    vsdrivevary -p drvpoolusr -s offline -V

vsmedclassqry

The **vsmedclassqry** command queries for the attributes of a specified Media Class group or all Media Class groups. The members of the Media Class group and any additionally requested information on each piece of media is returned.

Command Syntax

**vsmedclassqry <mediaclass> | -a [-m | -v] [-I] [-h] [-H <hostname>]
[-P <priority>] [-R <retries>] [-T <timeout>] [-V <number >]**

Options / Arguments

Table 47 lists options and arguments for the **vsmedclassqry** command.

Options / Arguments	Description
<mediaclass> | -a	Specifies a media class (**mediaclass**) or all media classes (**-a**) on which to request information.
[-m]	Requests a list of media identifiers for all media associated with each reported media class.
[-v]	Requests detailed information for all media associated with each reported media class.
[-I]	Indicates that command line options are to be read from **stdin**.
[-h]	Displays command usage.
[-H <hostname>]	Hostname of the Media Manager server.
[-P <priority>]	Execution priority of the entered command. Default: 15 Valid value: 1 to 32
[-R <retries>]	Number of retries that the software attempts if a timeout is received. Default: 3
[-T <timeout >]	Time interval (in seconds) that the software waits for status before returning a timeout. Default: 120
[-V <number>]	RPC program number for the software. Default: 300016

Examples

- Requests that the Media Manager software return information on every Media Class group known to the Media Manager system. No media-specific information is requested.

    vsmedclassqry -a

- Requests that the Media Manager software return detailed information for every piece of media in the **medclassh1user** Media Class group.

    vsmedclassqry medclassh1user -m

vsmedqry

The **vsmedqry** command queries for the attributes of one or more specified media.

Command Syntax

vsmedqry <mediaID> | -a [-I] [-h] [-H <hostname>] [-P <priority>]
[-R <retries>] [-T <timeout>] [-V <number>]

Options / Arguments

Table 48 lists options and arguments for the **vsmedqry** command.

Table 48 Options and Arguments: vsmedqry

Options / Arguments	Description
**<mediaID>	-a**	Specifies a media type (**mediaID**) or all media (**-a**) to be queried. Separate multiple media IDs with a space.
[-I]	Indicates that command line options are to be read from **stdin**.
[-h]	Displays command usage.
[-H <hostname>]	Hostname of the Media Manager server.

Options / Arguments	Description
[-P <priority>]	Execution priority of the entered command. Default value: 15 Valid values: 1 to 32
[-R <retries>]	Number of retries that the software attempts if a timeout is received. Default value: 3
[-T <timeout>]	Time interval (in seconds) that the software waits for status before returning a timeout. Default value: 120
[-V <number>]	RPC program number for the software. Default value: 300016

Examples

- Requests that the Media Manager software return information on every piece of media known to the Media Manager system:

    vsmedqry -a

vspoolcfg

The **vspoolcfg** command is used to configure a drive pool.

Command Syntax

vspoolcfg -p <drivepool> -c <driveID> | -d | -i <driveID> | -r <driveID>
[-I] [-h] [-H <hostname>] [-P <priority>] [-R <retries>] [-T <timeout>] [-V <number>]

Options / Arguments

Table 49 lists options and arguments for the **vspoolcfg** command.

Table 49 Options and
Arguments: vspoolcfg

Options / Arguments	Description
-p <drivepool>	Configures the specified drive pool. (Maximum of 16 characters.)
-c <driveID>	Creates a drive pool with a list of drive identifiers.
-d	Deletes the drive pool.
-i <driveID>	Inserts drives into an existing drive pool.
-r <driveID>	Removes drives from an existing drive pool.
[-I]	Indicates that command line options are to be read from **stdin**.
[-h]	Displays command usage.
[-H <hostname>]	Hostname of the Media Manager server.
[-P <priority>]	Execution priority of the entered command. Default value: 15 Valid values: 1 to 32
[-R <retries>]	Number of retries that the software attempts if a timeout is received. Default value: 3
[-T <timeout>]	Time interval (in seconds) that the software waits for status before returning a timeout. Default value: 120
[-V <number>]	RPC program number for the software. Default value: 300016

vspoolqry

The **vspoolqry** command is used to display information for a specified drive pool or for all drive pools.

Command Syntax

vspoolqry -a | <drivepool> [-v] [-I] [-h] [-H <hostname>]
[-P <priority>] [-R <retries>] [-T <timeout>] [-V <number>]

Options / Arguments

Table 50 lists options and arguments for the **vspoolqry** command.

Table 50 Options and Arguments: vspoolqry

Options / Arguments	Description
-a	Displays information for all drive pools.
<drivepool>	Displays information for the specified drivepool. (Maximum of 16 characters.)
[-v]	Specifies verbose output.
[-I]	Indicates that command line options are to be read from **stdin**.
[-h]	Displays command usage.
[-H <hostname>]	Hostname of the Media Manager server.
[-P <priority>]	Execution priority of the entered command. Default: 15 Valid values: 1 to 32
[-R <retries>]	Number of retries that the software attempts if a timeout is received. Default: 3

Options / Arguments	Description
[-T <timeout>]	Time interval (in seconds) that the software waits for status before returning a timeout. Default: 120
[-V <number>]	RPC program number for the software. Default: 300016

Examples

• Requests the Media Manager to return detailed drive information for every drive in the **drvpoolsml** drive pool.

> **vspoolqry drvpoolsml -v**

vsreclassify

The **vsreclassify** command changes the Media Class group associated with the specified media.

Command Syntax

vsreclassify <mediaID> -c <currentmediaclass> -n <newmediaclass>
[-v] [-I] [-h] [-H <hostname>] [-P <priority>] [-R <retries>]
[-T <timeout>] [-V <number>]

Options / Arguments

Table 51 lists options and arguments for the **vsreclassify** command.

Table 51 Options and
Arguments: vsreclassify

Options / Arguments	Description
mediaID	Specifies one or more media to reclassify.

Options / Arguments	Description
-c <currentmediaclass>	Specifies the media class with which the specified media is to be associated.
-n <newmediaclass>	Specifies the new media class with which the specified media is to be associated.
[-v]	Specifies verbose output.
[-I]	Indicates that command line options are to be read from **stdin**.
[-h]	Displays command usage.
[-H <hostname>]	Hostname of the Media Manager server.
[-P <priority>]	Execution priority of the entered command. Default: 15 Valid values: 1 to 32
[-R <retries>]	Number of retries that the software attempts if a timeout is received. Default: 3
[-T <timeout>]	Time interval (in seconds) that the software waits for status before returning a timeout. Default: 120
[-V <number>]	RPC program number for the software. Default: 300016

Examples

- Requests that the Media Manager software reclassify media **MED002**, **MED023** and **MED044** from media class **medclassm1** to **medclassstgsm11** and return status on the media only if processing for that piece of media was unsuccessful.

> **vsreclassify MED002 MED023 MED044 -c medclassm1 -n medclassstgsm1**

Contacting Quantum

More information about this product is available on the Quantum Service and Support website at www.quantum.com/ServiceandSupport. The Quantum Service and Support website contains a collection of information, including answers to frequently asked questions (FAQs). You can also access software, firmware, and drivers through this site.

To request a software upgrade, visit www.quantum.com/ServiceandSupport/Upgrade/Index.aspx. For further assistance, or if training is desired, contact Quantum Global Services:

United States	800-284-5101 (toll free)
	949-725-2100
EMEA	00800-4-782-6886 (toll free)
	+49 6131 3241 1164
APAC	+800 7826 8887 (toll free)
	+603 7953 3010

For worldwide support:

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