PART NUMBER
312520501

VERSION NUMBER
4.0

EDITION NUMBER
1

ASM

Application Storage Manager ™

DISASTER RECOVERY GUIDE (UNIX)

For ASM, ASM-QFS, and ASM/QFS-Standalone

PRODUCT TYPE
SOFTWARE

StorageTek®

Application Storage Manager™ (ASM)

ASM, ASM-QFS, and ASM/QFS-Standalone Disaster Recovery Guide

Version 4.0

for UNIX

First Edition

Part Number 312520501

Information contained in this publication is subject to change without notice. Comments concerning the contents of this publication should be directed to:

Global Learning Solutions
Storage Technology Corporation
One StorageTek Drive
Louisville, CO 80028-3256
USA

Limitations on Warranties and Liability

Storage Technology Corporation cannot accept any responsibility for your use of the information in this document or for your use in any associated software program. You are responsible for backing up your data. You should be careful to ensure that your use of the information complies with all applicable laws, rules, and regulations of the jurisdictions in which it is used.

Restricted Rights

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) and (2) of the Commercial Computer Software - Restricted Rights at FAR 52.227-19 (June 1987), as applicable.

Export Destination Control Statement

These commodities, technology or software were exported from the United States in accordance with the Export Administration Regulations. Diversion contrary to U.S. law is prohibited.

Information Control

First Edition (May 2003)

Contents

Codes

Preface

Disaster recovery preparation should be an essential part of any site's operational policies. This manual describes the steps to prepare for disaster recovery and steps to recover from a disaster, should one occur. The information in this manual pertains to the ASM and ASM-QFS 4.0 releases, which are supported on the Solaris™ 7, Solaris 8, and Solaris 9 operating environments.

This manual describes the system data (metadata) you need to protect and how to use that data to reconstruct or recover lost data. The types of data recovery addressed in this manual range from recovering a single lost file to recovering large amounts of data lost in a fire, flood, or other disaster.

Before You Read This Book

You, the system administrator, are assumed to be knowledgeable about Solaris system and network administration procedures, including installation, configuration, creation of accounts, and system backups.

Before you read this book, you need to understand how to administer ASM/QFS-Standalone, ASM, and ASM-QFS file systems as described in the other manuals under "Related Documentation" on page xiv.

How This Book Is Organized

Disaster preparation procedures described in Chapter 1 are applicable for ASM/QFS-Standalone, ASM, and ASM-QFS file systems and for all types of archive media. The recovery procedures in the other chapters of this manual apply only to ASM, or ASM-QFS file systems.

Also, while the procedures in Chapter 2 are for recovering individual files from all supported types of archive media, the recovery procedures for damaged file systems in Chapter 3 apply only to file systems archived on tape or on magneto optical disk. Procedures for recovering file systems archived on hard disks are outside the scope of this manual.

This manual contains the following chapters:

- Chapter 1 describes what to do to prepare for disaster recovery.
- Chapter 2 explains how to recover individual data files.
- Chapter 3 explains how to recover data from damaged volumes.

- Chapter 4 explains how to recover data from damaged file systems.
- Chapter 5 provides overall guidelines for recovery after a catastrophic failure.

Related Documentation

This manual is part of a set of documents that describes the operations of the ASM, and ASM-QFS software products. Table 1. shows the complete release 4.0 documentation set for these products.

TABLE P-1

Title	Part Number
ASM, ASM-QFS, and ASM/QFS-Standalone Storage and Archive Management Guide	312520101
ASM-Remote Administrator's Guide	312520201
ASM, ASM-QFS, and ASM/QFS-Standalone Installation and Configuration Guide	312502301
ASM, ASM-QFS, and ASM/QFS-Standalone File System Administrator's Guide	312502401
ASM, ASM-QFS, and ASM/QFS-Standalone Disaster Recovery Guide	312502501

Note that the *ASM Remote Administrator's Guide* has not been updated for the 4.0 release. An updated version of this manual will be provided at a later date.

How to Obtain Documentation

All the ASM publications are available from the following

sources:

- Contact StorageTek Publication Sales and Service at 800-436-5554 or send a fax to 303-661-7367.
- Online (for viewing and printing), at the StorageTek Customer Resource Center (CRC) website at: www.support.storagetek.com. Click on Software and go to the ASM Software list.

Access to the CRC site requires a password. To obtain a password, call StorageTek Customer Support at 1-800-678-4430.

Support

The publication “Requesting Software Support” is included in your media package. Please consult this book for the most information on your ASM support options, as well as regional phone numbers and procedures.

Using UNIX Commands

This document does not contain information on basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices.

See one or more of the following for this information:

- *Solaris Handbook for Sun Peripherals*
- AnswerBook2™ online documentation for the Sun Solaris OE
- Other software documentation that you received with your system

Typographic Conventions

Table 1. lists the typographic conventions used in this manual.

Table 1. Typographic Conventions

Typeface or Symbol	Meaning	Examples
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output.	Edit your `.login` file. Use `ls -a` to list all files. `% You have mail.`
`AaBbCc123`	What you type, when contrasted with on-screen computer output.	**`% su`** `Password:`
AaBbCc123	Book titles; new words or terms; words to be emphasized; and command line variables to be replaced with a real name or value.	Read Chapter 6 in the *User’s Guide*. These are called *class* options. You *must* be `root` to do this. To delete a file, type `rm` *filename*.
`[]`	In syntax, brackets indicate that an argument is optional.	`scmadm [-d sec] [-r n[:n][,n]...] [-z]`

Table 1. Typographic Conventions

Typeface or Symbol	Meaning	Examples
`{ arg | arg}`	In syntax, braces and pipes indicate that one of the arguments must be specified.	`sndradm -b {phost | shost}`
`\`	At the end of a command line, the backslash (\) indicates that the command continues on the next line.	`atm90 /dev/md/rdsk/d5 \ /dev/md/rdsk/d1`

Shell Prompts

Table 2. shows the shell prompts that this manual uses.

Table 2. Shell Prompts

Shell	Prompt
C shell	*machine-name%*
C shell superuser	*machine-name#*
Bourne shell and Korn shell	$
Bourne shell and Korn shell superuser	#

Disaster Preparation

1

This chapter provides the backup and dump processes and information you need for preparing for disaster recovery.

This chapter includes the following subsections:

Planning for Disaster Recovery

Data must be backed up and disaster recovery processes must be put in place so that data can be retrieved if any of the following occur:

- Data is accidentally deleted
- Storage media fail
- Systems fail

- Any combination of the above events occurs on a small or large scale

This chapter provides the information you need to know about backing up metadata and other important configuration data. The rest of the chapters in this manual describe how to use the data you back up to recover from various types of disasters.

Setting up processes for doing backups and system dumps is only part of preparing to recover from a disaster. The following are also necessary:

- Documenting everything
  - Document your hardware configuration, backup policies and scripts and all your restoration processes.
  - Keep paper copies of the documents offsite with copies of the backup media.
- Verifying that the files and the system are actually recoverable
  - Test all scripts that you create (see “Testing Backup Scripts and cron Jobs” on page 3).
  - Routinely test the retrieval procedures that are described in the other chapters in this manual. See “Testing the Disaster Recovery Process” on page 3.

Recovering from Failure of the Operating Environment Disk

When a disk containing the operating environment for a system fails, after you replace the defective disk(s), you need to do what is called *bare metal recovery* before you can do anything else. Two bare metal recovery approaches are available:

- Reinstall the operating environment, patches, and backed-up configuration files

  This process is slower than the second alternative described below.
- Restore a system image backup made ahead of time on a separate hard disk.

  Image backups need to be made only when system configuration changes are made. A negative consideration about this approach is that it is difficult to safely transport hard disks to off site storage.

Testing Disaster Recovery

After you have done all the recovery preparations described in this chapter, do the test described in the following sections:

- “Testing Backup Scripts and `cron` Jobs.”

- “Testing the Disaster Recovery Process”

Testing Backup Scripts and cron Jobs

Always test backup scripts and `cron(1)` jobs on a development or test system before rolling it out to all systems.

- Test each script’s syntax.
- Test each script on one system.
- Test each script on a small number of systems.
- Try to simulate every possible error a script might encounter in the middle of the backup:
  - Eject the volume.
  - Switch the machine off.
  - Pull out the network connection.
  - Switch off the backup server or device.

Testing the Disaster Recovery Process

Use the information in the other chapters in this manual to do the following tests, to verify how well your disaster recovery process works:

- Restore a single file that is currently on the system.
- Restore an older version of a file.
- Restore an entire file system and compare it against the original.
- Enact a scenario where the system is down and restore the system.
- Retrieve some volumes from off-site storage.
- Enact a scenario in which last night’s backup failed, and you need to restore data using system and archiver logs.
- Enact a scenario in which the system is destroyed and recover the system’s data.
- Enact a scenario in which the disk containing the operating environment fails.

Do these tests periodically. Especially make it a point to do these tests anytime you make changes to the software.

Guarding Against or Troubleshooting Data Loss

Table 1. shows the usual causes of data loss, with notes and suggestions about how to avoid or respond to each type of loss.

Table 1. Causes of Data Loss, With Notes and Suggestions

Causes	Notes	Suggestions
User Error	ASM and ASM-QFS file systems are protected from access by unauthorized users because of the UNIX superuser mechanism. You can also restrict administrative actions to an optional administrative group.	
System reconfiguration	File systems can be made unavailable by any of the following: Dynamically-configured SAN components Overwritten system configuration files Failure of connectivity components	Rebuild the file system only after verifying that a configuration problem is not the cause of the apparent failure. See “Precautions Before Starting Data Restoration” on page 5 and “To Troubleshoot an Inaccessible File System” on page 5, and “Recovering From Catastrophic Failure” on page 71.
Hardware failure	Using disk storage systems managed by hardware RAID has the following advantages over systems managed using software RAID: • More reliability • Fewer resources are consumed on the host system	• Use hardware RAID disk storage systems wherever possible • Use `samfsck(1M)` to check and fix hardware-based file system consistency problems.

Table 1. Causes of Data Loss, With Notes and Suggestions

Causes	Notes	Suggestions
Hardware failure	• Better performance • Hardware-based inconsistencies in ASM and ASM-QFS file systems can be checked and fixed by unmounting the file system and running `samfsck(1M)` command.	• See “To Troubleshoot an Inaccessible File System” on page 5 for an example. Also see “Recovering From Catastrophic Failure” on page 71.

Precautions Before Starting Data Restoration

Some apparent data losses are actually caused by cabling problems or configuration changes.

Caution: Do not reformat a disk, relabel a tape, or make other irreversible changes until you are convinced that the data on the disk or tape is completely unrecoverable.
Make sure to eliminate the fundamental causes for a failure before making irreversible changes. Back up anything you change before you change it, if possible.

Do the procedure in “To Troubleshoot an Inaccessible File System” before commencing a data recovery process.

To Troubleshoot an Inaccessible File System

1. Check cables and terminators.
2. If you cannot read a tape or magneto-optical cartridge, try cleaning the heads in the drive, or try reading the cartridge in a different drive.
3. Check the current state of your hardware configuration against the documented hardware configuration.

   Go to Step 4 only when you are certain that a configuration error is not to blame.
4. Unmount the file system, and run `samfsck(1M)`.

```
# umount file_system_name
# samfsck file_system_name
```

5. If you find the file system is still inaccessible, use the procedures in the other chapters in this manual to restore the file system.

Prerequisites for Data Recovery

For ASM and ASM-QFS file systems, the following are prerequisites for disaster recovery:

- Up-to-date archive copies

  The effectiveness of any of the ASM and ASM-QFS recovery methods relies primarily on frequent archiving being done.

- Up-to-date metadata dumps

  See “Metadata Used in Disaster Recovery” on page 6.

- Archiver logs

  If recent metadata is not available, archiver logs can help you recreate the filesystem directly from archive media. This method can be used whether or not ASM or ASM-QFS are installed.

  See “Using Archiver Logs” on page 23.

Note: Using archiver logs is a lot more time consuming that using metadata to retrieve data, so this approach should not be relied upon. It is not used unless there is no alternative.

Metadata Used in Disaster Recovery

Metadata consists of information about files, directories, access control lists, symbolic links, removable media, segmented files, and the indexes of segmented files. Metadata must be restored before lost data can be retrieved.

With the up-to-date metadata, the data can be restored as follows:

- File data can be restored even if the file has been removed from the file system.
- Individual files or entire file systems can be moved from one file system to another, or even from one server to another.

.inodes File Characteristics

In ASM/DMS file systems, the `.inodes` file contains all the metadata except for the directory namespace (which consists of the pathnames to the directories where the files are stored). The `.inodes` file is located in the `root (/)` directory of the file system. For a file system to be restored, the `.inodes` file is needed along with the additional metadata.

Figure 1. illustrates some characteristics of the `.inodes` file. The arrows with the dashed lines indicate that the `.inodes` file points to file contents on disk and to the directory namespace. The namespace also points back to the

`.inodes` file. Also indicated is that in ASM and ASM-QFS file systems where archiving is being done, the `.inodes` file also points to archived copies.

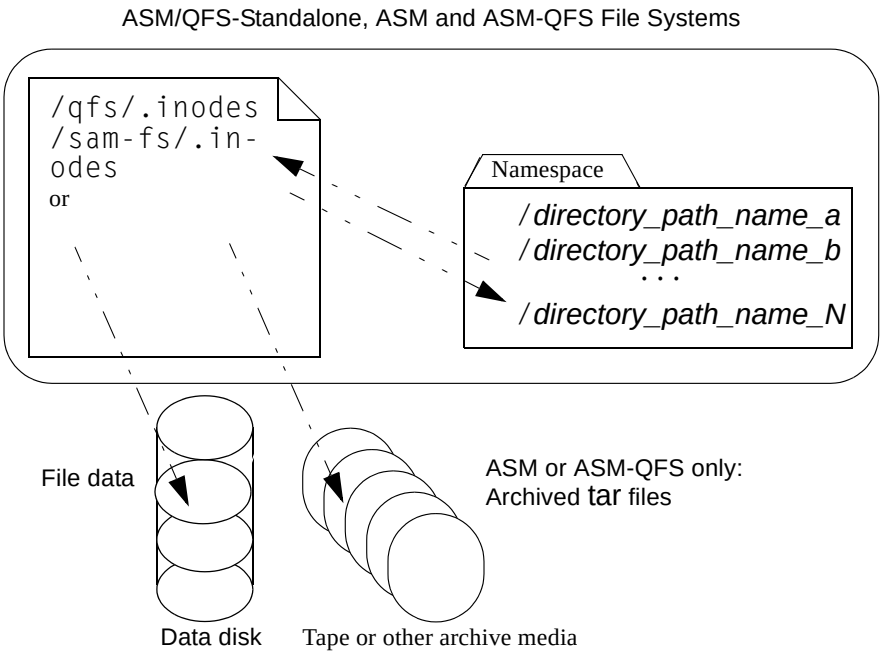


Figure 1. The .inodes File in ASM/DMS File Systems

Note: ASM/QFS-Standalone has no archiving capability. See the *ASM, ASM-QFS, and ASM/QFS-Standalone Installation and Configuration Guide* for how to back up ASM/QFS-Standalone metadata.

The `.inodes` file is not archived. For more about protecting the `.inodes` file in these types of file systems, see “ASM and ASM-QFS Disaster Recovery Features” on page 8 and “Backing Up the Metadata in ASM and ASM-QFS File Systems” on page 11.

More About Directory Pathnames

As indicated in Figure 1., the namespace (in the form of directories) does not point to the archive media. The directory pathnames for each archived file *are* copied into the headers of the `tar(1)` files on the archive media that contain the files, *but* for reasons illustrated elsewhere (in Table 3.), the directory pathnames in the `tar` file headers may get out of sync with the actual locations of the files on the disk.

One reason why the two pathnames can get out of sync is that the pathnames in the `tar` file header do not show the originating file system. Table 2. shows how the directory pathname shown in the left column would appear in the `tar` file header in the right column, without the component that shows the name of the originating file system `/samfs1`.

Table 2. Comparing a Full Pathname With a Pathname in a `tar` Header

Full Pathname	Pathname in `tar` Header on Archive Media
`/samfs1/dir1/filea`	`dir1/` `dir1/filea`

Table 3. summarizes an example scenario, shows the result, and suggests a precaution.

Table 3. Example of Potential Pitfalls

Scenario	Result	Precaution
File is saved to disk, archived, then later moved, either by use of the `mv`(1) command or by restoration from a `samfsdump`(1M) output file using `samfsrestore`(1M) into an alternate path or file system.	• Archive copy is still valid. `.inodes` file still points to the archive media Pathname in the `tar` file header no longer matches the namespace on disk. Name of the file system is not available in the `tar` file header.	Keep the data from each file system on its own unique set of tapes or other archive media, and do not mix data from multiple file systems.

The potential for inconsistency does not interfere with recovery in most cases, because the directory pathnames in the `tar` headers are not used when data is being recovered from an archive. The directory pathnames on the `tar` headers on the archive media are only used in an unlikely disaster recovery scenario where no metadata is available and the file system must be reconstructed from scratch using the `tar` command.

ASM and ASM-QFS Disaster Recovery Features

The features of ASM and ASM-QFS file systems described in Table 4. streamline and speed up data restoration and minimize the risk of losing data in the case of unplanned system outage.

Table 4. Disaster Recovery Features of ASM and ASM-QFS File Systems

Feature	Comparison	Advantage
Identification records, serial writes, and error checking are dynamically used to check and manage file system consistency.	Eliminates the need to check file systems (by running the `fsck(1M)` command) before re-mounting the file systems or to rely on journal recovery mechanisms.	*Speed*. Because each file system is already checked and repaired when the server reboots after an outage, the server gets back into production more quickly.
Files are archived into a robotic library transparently and continuously. Archiving is configurable: after specified sleep intervals, via scheduled `cron`(1M) jobs, or on demand.	Nightly or weekly backups interfere with normal use of the system while the backups are being done and protection is not continuous.	*Data protection*. Because archiving is continuous, there are no gaps in data protection. Data backups no longer interfere with production.
Data can remain on disk or can be automatically released from the disk and then transparently staged back from archive media when needed.	Files no longer need to take up disk space. Files that are removed from the disk, files are instantly available without administrator intervention.	*Speed*. Disk space requirements may be lessened without inconvenience to users.
Files can be archived to as many as four separate media, each of which can be of a different type, and with ASM-Remote, to remote locations.	Multiple copies can be easily made in multiple locations.	*Data Protection*. With the potential for multiple copies at multiple locations, the loss of one copy or even of an entire location does not mean a complete loss of data.
Files are archived in standard `tar(1)` format files.	`tar` files can be restored onto any file system type.	*Flexibility*. ASM and ASM-QFS file systems do not need to be available.

Table 4. Disaster Recovery Features of ASM and ASM-QFS File Systems

Feature	Comparison	Advantage
Metadata can be restored separately from data. Restoration of the files' contents to disk is configurable: files can be staged only when they are accessed or in advance of anticipated need.	Restoring metadata allows users to access the system and their data without waiting until all data is restored to disk.	*Speed*. Access to the server is quicker than if all data needed to be restored before user access was allowed.

Guidelines for Performing Dumps

- Perform dumps with the file system mounted.
- Perform metadata dumps at a time when files are not being created or modified.

  At any given time, some files need to be archived because they are new, while others need to be rearchived because they are modified or because their archive media is being recycled. See the following table for definitions of terms that apply to files archived onto archive media.

Table 5. Terms Related to Dumping Metadata

Term	When Used	Comments
stale	The archived copy does not match the online file.	A new copy must be created. Stale files can be detected using the `sls` command with the `-D` option. See the `sls(1M)` man page. Also see "Error Messages That Identify Damaged Files."
expired	No inode points to the archived copy.	A new archive copy was already created, and the file's inode correctly points to the new archive copy.

  Dumping metadata during a time when files are not being created or modified avoids the dumping of metadata for files that are stale and minimizes the creation of damaged files.

- If an error message identifies a file as damaged, run the `samfsdump(1M)` command again after the specified file is archived.

  When any stale files exist while metadata and file data are being dumped, the `samfsdump` command generates a warning message. The following

warning message is displayed for any files that do not have an up-to-date archive copy:

```
/pathname/filename: Warning! File data will not be
recoverable (file will be marked damaged).
```

Caution: If you see the above message and do not rerun the `samfsdump` command after the specified file is archived, the file will not be retrievable.

If `samfsrestore(1M)` is later used to attempt to restore the damaged file, the following message is displayed:

```
/pathname/filename: Warning! File data was previously not
recoverable (file is marked damaged).
```

Backing Up the Metadata in ASM and ASM-QFS File Systems

In ASM and ASM-QFS file systems, the `archiver(1M)` command can copy both file data and metadata—other than the `.inodes` file—to archive media. For example, if you create an ASM file system with a family-set name of `samfs1`, you can tell the `archiver` command to create an archive set also called `samfs1.` (See the `archiver.cmd(4)` man page for more information.) You can later retrieve damaged or destroyed file systems, files, and directories as long as the archive media onto which the archive copy was written has not been erased and as long as a recent metadata dump file is available.

The `samfsdump(1M)` command allows you to back up metadata separate from the file system data. The `samfsdump` command creates metadata dumps (including the `.inodes` file) either for a complete file system or of a portion of a file system. A `cron(1M)` job can be set up to automate the process.

If you dump metadata often enough using `samfsdump`, the metadata is always available to restore file data from the archives using `samfsrestore(1M).`

Note: Files written to the file system after metadata dumps begin might not be archived, and archive copies on cartridges might not be reflected in the metadata dump. Consequently, the files might not be known to the system if the dump is used to restore the file system. Files written to the file system or archived after the metadata dump are picked up during the next metadata dump.

In summary, using the samfsdump method to dump metadata has the following advantages:

- The samfsdump command saves the relative path for each file.
- The samfsdump command is run on mounted file systems.
- The metadata dump file generated by the samfsdump command contains all information required for restoring an ASM or ASM-QFS file system. The metadata dump file contains the .inodes file, directory information, and symbolic links.
- The samfsdump and samfsrestore method is flexible. This process enables you to restore an entire file system, a directory hierarchy, or a single file. With samfsdump(1M) and samfsrestore(1M), you can split an existing file system into multiple file systems or you can join multiple file systems into a single file system.
- The samfsrestore command defragments the .inodes file, the file system name space, and file data. See the following table for details.

Table 6.

.inodes file and file system name space	During a file system restoration, files and directories are assigned new inode numbers based on directory location; only the required number of inodes are assigned. Inodes are assigned as the samfsrestore process restores the directory structure.
file data	File data is defragmented because files that were written in a combination of small disk allocation units (DAUs) and large DAUs are staged back to the disk using appropriately sized DAUs.

- When the samfsrestore process is complete, all directories and symbolic links are online and files are ready to be accessed.

Creating samfsdump Dump Files

If you have multiple ASM or ASM-QFS file systems, make sure that you routinely dump the metadata for every file system. Look in /etc/vfstab for all file systems of type samfs.

Make sure to save the dump for each file system in a separate file.

The following procedures describe how to find all the samfs type file systems and to dump metadata using samfsdump(1M):

- “To Find ASM and ASM-QFS File Systems” on page 13
- “To Create an ASM or ASM-QFS Metadata Dump File Manually” on page 14

- “To Create an ASM or ASM-QFS Metadata Dump File Automatically” on page 14

Note: The examples in these procedures use the names `/sam1` for an ASM file system mount point and `/dump_sam1` for the dump file system.

Using samfsdump With the –u Option

The `samfsdump(1M)` command `-u` option causes unarchived file data to be interspersed with the metadata. Note the following about the use of the `-u` option:

- A `samfsdump` command run with the `-u` option on a version 3.5 or 4.0 ASM or ASM-QFS file system cannot be restored to an earlier version (3.3.*x*) file system of the same type because versions 3.5 and 4.0 have new data structures. Dumps from a 4.0 version of either file system type can be restored on a 3.5 version and vice versa.
- A `samfsdump` dump taken using the `-u` option can be very large. The `samfsdump` command does not have any tape management or estimations such as those associated with `ufsdump(1M)`. You need to weigh the amount of dump storage space available against the risks of having unarchived data when using the `-u` option (as you do when setting up any data protection procedures). For more information, see also the `samfsdump` and `ufsdump` man pages.

To Find ASM and ASM-QFS File Systems

- Look in the `vfstab(4)` file to find mount points for all samfs-type file systems.

Note: Both ASM and ASM-QFS file systems are identified as type `samfs` in the `/etc/vfstab` file.

The following screen example shows three `samfs`-type filesystems with the family names `samfs1`, `samfs2`, and `samfs3`, whose mount points are `/sam1`, `/sam2`, and `/sam3`.

```
# vi /etc/vfstab
samfs1 -       /sam1 samfs   -       no
high=80,low=70,partial=8
samfs2 -       /sam2 samfs   -       no high=80,low=50
samfs3 -       /sam3 samfs   -       no high=80,low=50
```

To Create an ASM or ASM-QFS Metadata Dump File Manually

1. Log in as root.
2. Go to the mount point for the `samfs` type file system mount point or to the directory that you are dumping.

```
# cd /sam1
```

See “To Find ASM and ASM-QFS File Systems” on page 13 if needed.

3. Enter the `samfsdump(1M)` command to create a metadata dump file.

The following example command line shows an ASM file system metadata dump file being created on February 14, 2004 in a `dumps` subdirectory in the dump file system `/dump_sam1/dumps`. The output of the `ls`(1) command line shows the date is assigned in the *`yymmdd`* format as the dump file's name, `040214`.

```
# samfsdump -f /dump_sam1/dumps/‘date +\%y\%m\%d‘
# ls /dump_sam1/dumps
040214
```

To Create an ASM or ASM-QFS Metadata Dump File Automatically

1. Log in as `root`.
2. Enter the `crontab(1M)` command with the `-e` option to make an entry to dump the metadata for each file system.

   The `crontab` entry in the following screen example runs at 10 minutes past 2 a.m. every day and does the following:

   - In the dump file system's dumps directory `(/dump_sam1/dumps)`, removes files older than three days
   - Dumps the metadata from`/sam1`
   - Assigns the date of the metadata dump as the file's name in *`yymmdd`* format.

```
# crontab -e
10 2 * * * ( find /dump_sam1/dumps -type f -mtime +72 -
print | xargs -l1 rm -f; cd /sam1 ; /opt/SUNWsamfs/
sbin/samfsdump -f /sam1/dumps/‘date +\%y\%m\%d ‘ )
:wq
```

Note: Make the `crontab` entry on a single line. Because the line in the previous screen example is too wide for the page's format, it breaks into multiple lines.

If the `crontab` entry in the previous screen example ran on March 20, 2004, the full pathname of the dump file would be: `/dump_sam1/dumps/040320.`

Disaster Recovery Commands and Tools

The following table summarizes the commands used most frequently in disaster recovery efforts. For more information about these commands, see their `man(1)` pages.

Table 7. Disaster Recovery Commands and Tools

Command	Description	Used By
`qfsdump(1M)`	Dumps ASM/QFS-Standalone file system metadata and data.	ASM/QFS-Standalone
`qfsrestore(1M)`	Restores ASM/QFS-Standalone file system metadata and data.	ASM/QFS-Standalone
`samfsdump(1M)`	Dumps ASM and ASM-QFS file system metadata.	ASM, ASM-QFS
`samfsrestore(1M)`	Restores ASM and ASM-QFS file system metadata.	ASM, ASM-QFS
`star(1M)`	Restores file data from archives.	ASM, ASM-QFS

Other scripts and helpful sample files are located `/opt/SUNWsamfs/examples` or are available from StorageTek.

The following table describes some disaster recovery utilities in the `/opt/SUNWsamfs/examples` directory and explains their purpose. You must modify all of the listed shell scripts, except for `recover.sh(1M)`, to suit your configuration before using them. See the comments in the files.

.

Table 8. Disaster Recovery Utilities

Utility	Description
restore.sh(1M)	Executable shell script that stages all files and directories that were online at the time a samfsdump(1M) was taken. This script requires that a log file generated by sammkfs(1M) or samfsrestore(1M) be used as input. Modify the script as instructed in the comments in the script. See also the restore.sh(1M) man page.
recover.sh(1M)	Executable shell script that recovers files from tape, using input from the archiver log file. For more information about this script, see the recover.sh(1M) man page and the comments in the script itself. Also see “Using Archiver Logs” on page 23.
stageback.sh	Executable shell script that stages files that have been archived on accessible areas of a partially damaged tape. Modify the script as instructed in the script’s comments. For how the script is used, see “Damaged Tape Volume—No Other Copies Available” on page 57.
tarback.sh(1M)	Executable shell script that recovers files from tapes by reading each tar(1) file. Modify the script as instructed in the script’s comments. For more information about this script, see the tarback.sh man page. See also “Unreadable Label—No Other Copies Available” on page 65.

Caution: Improper use of the restore.sh, recover.sh, or tarback.sh scripts can damage user or system data. Please read the man pages for these scripts before attempting to use them. For additional help with using these scripts, contact StorageTek customer support.

The info.sh Script

The /opt/SUNWsamfs/sbin/info.sh script is not a backup utility, but it should be run whenever changes are made to the system’s configuration.

The info.sh(1M) script creates a file containing all the configuration information needed for reconstructing an ASM or ASM-QFS installation from scratch if you ever need to rebuild the system. You can use the crontab(1) command with the -e option to create a cron(1M) job to run the info.sh script at desired intervals.

The `info.sh` script writes the reconfiguration information to `/tmp/SAMreport`.

Make sure that the `SAMreport` file is moved from the `/tmp` directory after creation to a fixed disk that is separate from the configuration files and outside the ASM or ASM-QFS environment. For more information about managing the `SAMreport` file, see the `info.sh(1M)` man page.

What to Back Up and How Often

Table 9. describes the files that should be backed up and how often the files should be backed up onto a location outside the file system environment.

Where “Regularly” is shown in the “Backup Frequency” column, each site’s system administrator should decide the appropriate intervals based on that site’s requirements. Except where specified, use whatever backup procedures you choose.

Table 9. Which Files to Back Up and How Often (Sheet 1 of 4)

Data Type	Backup Frequency	Comments
Site-modified versions of filesystem backup and restoration shell scripts.	After modification	See the default scripts listed in “Disaster Recovery Commands and Tools” on page 15.
Site-created shell scripts and `cron(1)` jobs created for backup and restoration.	After creation and after any modification	
`SAMreport` output from the `info.sh(1M)` script.		See the `info.sh` script and `SAMreport` output file described in “The info.sh Script” on page 16.
ASM/QFS-Standalone metadata and data (see “Metadata Used in Disaster Recovery” on page 6 for definitions).	Regularly	Files altered after `qfsdump(1M)` is run cannot be recovered by `qfsrestore(1M)`, so take dumps frequently. For more information, see “Metadata Used in Disaster Recovery” on page 6.

Table 9. Which Files to Back Up and How Often (Sheet 2 of 4)

Data Type	Backup Frequency	Comments
ASM and ASM-QFS metadata (see "Metadata Used in Disaster Recovery" on page 6 for definitions).	Regularly	Use the `samfsdump(1M)` command to back up metadata. Files altered after `samfsdump` is run cannot be recovered by `samfsrestore(1M)`, so take dumps frequently or at least save the inodes information frequently. For more information, see "Backing Up the Metadata in ASM and ASM-QFS File Systems" on page 11.
ASM and ASM-QFS device catalogs.	Regularly	Back up all library catalog files, including the historian file. A library catalog for each automated library, pseudolibrary on ASM-Remote clients, and for the historian (for cartridges that reside outside the automated libraries) are in `/var/opt/SUNWsamfs/catalog`.
archiver log files from an ASM or ASM-QFS file system where the archiver is being used.	Regularly	Specify a pathname and name for an archiver log file in the `archiver.cmd` file and back up the archiver log file. See the `archiver.cmd(4)` man page for how to specify an archiver log file for each file system. Also see "Using Archiver Logs" on page 23.

Configuration files and other similar files modified at your site. Note that these reside outside the ASM or ASM-QFS file system.	At installation and after any modification	The following files may be created at your site in the `/etc/opt/SUNWsamfs` directory:
		`archiver.cmd(4)`
		`defaults.conf(4)`
		`diskvols.conf(4)`
		`hosts.`*`fsname`*
		`LICENSE.`*`rel_level`*
		`mcf(4)`
		`preview.cmd(4)`
		`recycler.cmd(4)`
		`releaser.cmd(4)`
		`samfs.cmd(4)`
		`samlogd.cmd(4)`
		`stager.cmd(4)`

Table 9. Which Files to Back Up and How Often (Sheet 3 of 4)

Data Type	Backup Frequency	Comments
Network-attached-library configuration files.	At installation and after any modification	If using network-attached libraries, make sure to back up the configuration files. The exact names of the files are listed in the `Equipment Identifier` field of the `/etc/opt/SUNWsamfs/mcf` file on each line that defines a network-attached robot. See the `mcf(4)` man page for more details.
ASM-Remote configuration files.	At installation and after any modification	If using ASM-Remote software, make sure to back up the configuration files. The exact names of the files are listed in the `Equipment Identifier` field of the `/etc/opt/SUNWsamfs/mcf` file on each line that defines an ASM-Remote client or server. See the `mcf(4)` man page for more details.
Installation files.	At installation and after any modification	The following files are created by the software installation process. If you have made local modifications, preserve (or back up) these files: `/etc/opt/SUNWsamfs/inquiry.conf`[1] `/opt/SUNWsamfs/sbin/ar_notify.sh`[1] `/opt/SUNWsamfs/sbin/dev_down.sh`[1] `/opt/SUNWsamfs/sbin/recycler.sh`[1] `/kernel/drv/samst.conf`[1] `/kernel/drv/samrd.conf`

Table 9. Which Files to Back Up and How Often (Sheet 4 of 4)

Data Type	Backup Frequency	Comments

Files modified at installation time.	At installation and after any modification	The following files are modified as part of the software installation process:
		`/etc/syslog.conf`
		`/etc/system`
		`/kernel/drv/sd.conf`[1]
		`/kernel/drv/ssd.conf`[1]
		`/kernel/drv/st.conf`[1]
		`/usr/kernel/drv/dst.conf`[1]
		Back the above files up so you can restore them if any of the files are lost or if the Solaris OE is reinstalled And if you modify the files, make sure to back them up again.

`SUNWqfs` and `SUNWsamfs` software packages.	Once, shortly after downloading	The ASM/QFS-Standalone, ASM, and ASM-QFS software can be reinstalled easily from the release package. Make sure you have a record of the revision level of the currently running software. If the software is on a CD-ROM, store the CD-ROM in a safe place. If you download the software from the StorageTek Customer Resource Center (CRC), back up the downloaded package(s). This saves time if you have to reinstall the software because you avoid having to download a fresh copy if you lose data.
Solaris OE and patches,2 and unbundled patches.	At installation	The Solaris OE can be reinstalled easily from the CD-ROM, but make sure you have a record of all installed patches. This information is captured in the `SAMreport` file generated by the `info.sh(1M) script`, which is described under “The info.sh Script” on page 16. This information is also available from the Sun Explorer tool.

1.Protect this file only if you modify it.

Additional Backup Considerations

The following is a list of questions to also consider when preparing your site's disaster recovery plan.

- What is the right number of `samfsdump(1M)` or `qfsdump(1M)` files to retain at your site?

  Table 10. compares the types of dumps that are done in the various file system types.

Table 10. Types of Dumps Performed on ASM/QFS-Standalone Compared to ASM and ASM-QFS File Systems

Filesystem Type	Dump Command Output	Notes
ASM/QFS-Standalone	A `qfsdump(1M)` command generates a dump of both metadata and data.	See the *ASM, ASM-QFS, and ASM/QFS-Standalone Installation and Configuration Guide* for how to back up ASM/QFS-Standalone metadata.
ASM, ASM-QFS	The `samfsdump(1M)` command *without* the `-u` option generates a metadata dump file.	A metadata dump file is relatively small, so you should be able to store many more metadata dump files than data dump files. Restoration of the output of `samfsdump` without the `-u` option is quicker, because the data is not restored until accessed by a user.
	The `samfsdump(1M)` command with the `-u` option dumps file data for files that do not have a current archive copy.	The dump files are substantially larger, and the command takes longer to complete. However, restoration of the output from `samfsdump` with `-u` restores the file system back to its state when the dump was taken.

Retain enough data and metadata to ensure that you can restore the file systems according to your site's needs. The appropriate number of dumps

to save depends, in part, on how actively the system administrator monitors the dump output. If an administrator is monitoring the system daily to make sure the `samfsdump(1M)` or `qfsdump(1M)` dumps are succeeding (making sure enough tapes are available and investigating dump errors), then keeping a minimum number of dump files to cover vacations, long weekends, and other absence might be enough.

- If you are archiving data, are you actively recycling archive media? If so, make sure to schedule metadata copies to occur after recycling completes.

  If your site is using the `sam-recycler(1M)` command to reclaim space on archive media, it is critical that you make metadata copies *after* `sam-recycler` has completed its work. If a metadata dump is created before the `sam-recycler` exits, the information in the metadump about archive copies becomes out of date as soon as `sam-recyler` runs. Also, some archive copies may be made inaccessible because the `sam-recycler` command may cause archive media to be relabeled.

  Check root's `crontab(1)` entry to find out if and when the `sam-recycler` command is being run, and then, if necessary, schedule the creation of metadump files around the `sam-recycler` execution times. For more about recycling, see the *ASM, ASM-QFS, and ASM/QFS-Standalone Storage and Archive Management Guide*.

- How much data should you store off site, and in what format?

  Off-site data storage is an essential part of a disaster recovery plan. In the event of a disaster, the only safe data repository might be an offsite vault. Beyond the recommended two copies of all files and metadata that you should be keeping in house as a safeguard against media failure, consider making a third copy on removable media and storing it offsite. To encourage administrators to eject and retain extra media for off-site storage, media ejected from a robotic library is not considered in the licensed slot count.

  ASM-Remote offers you the additional alternative of making archive copies in remote locations on a LAN or WAN. Multiple ASM-Remote servers can be configured as clients to one another in a reciprocal disaster recovery strategy.

- Is it sufficient to restore only the metadata to a predisaster state or do you need also to restore all files that were online when the disaster happened?

  - The `qfsrestore(1M)` command restores both the ASM/QFS-Standalone file system metadata and the file data to the state reflected in the `qfsdump(1M)` file.

  - The `samfsrestore(1M)` command can restore an ASM or ASM-QFS file or file system to the state reflected in the `samfsdump(1M)` file. After the `samfsrestore(1M)` command is run, the metadata is restored, but the file data remains offline.

- If you need to restore all files that were online, you need to run the `samfsrestore` command with the `-g` option.
- The log file generated by the `samfsrestore` command's `-g` option contains a list of all files that were on the disk when the `samfsdump(1M)` command was run. This log file can be used in conjunction with the `restore.sh` shell script to restore the files on disk to their predisaster state. The `restore.sh` script takes the log file as input and generates stage requests for files listed in the log. By default, the `restore.sh` script restores all files listed in the log file.
- If your site has thousands of files that need to be staged, consider splitting the log file into manageable chunks and running the `restore.sh` script against each of those chunks separately to ensure that the staging process does not overwhelm the system. You can also use this approach to ensure that the most critical files are restored first. For more information, see the comments in `/opt/SUNWsamfs/examples/restore.sh`.

Using Archiver Logs

Archiver logging should be enabled in the `archiver.cmd(4)` file. Because archiver logs list all the files that have been archived and their locations on cartridges, archiver logs can be used to recover lost files since the last set of metadata dumps and backup copies were created.

Be aware of the following considerations:

- Processes writing to the archiver log continue to do so until they complete.
- The ASM and ASM-QFS systems create a new log file each time a process initiates a new write to the log, if a log file is not found.
- If a log file exists, data is appended to the existing file.
- The archiver log files grow over time, so they must be managed.

Set up and manage the archive logs by performing these procedures:

- "To Set Up Archiver Logging"
- "To Save Archiver Logs" on page 24

To Set Up Archiver Logging

- Enable archive logging in the `archiver.cmd` file (in the `/etc/opt/SUNWsamfs` directory).

  See the `archiver.cmd(4)` man page. The archiver log files are typically written to `/var/adm/`*`logfilename`*. The directory where you direct the logs to be written should reside on a disk outside the ASM or ASM-QFS environment.

To Save Archiver Logs

- Ensure that archiver log files are cycled regularly by creating a `cron(1M)` job that moves the current archiver log files to another location.

  The screen example shows how to create a dated copy of an archiver log named `/var/adm/archlog` every day at 3:15 a.m. The dated copy is stored in `/var/archlogs`.

Note: If you have multiple archiver logs, create a `crontab` entry for each one.

```
# crontab -e
15 3 * * 0 ( mv /var/adm/archlog /var/archlogs/‘date +%y%m%d
‘ ; touch /var/adm/archlog )
:wq
```

How and Where to Keep Copies of Disaster Recovery Files and Metadata

Consider writing scripts to create `tar(1)` files that contain copies of all the relevant disaster recovery files and metadata described in this chapter and to store the copies outside the file system. Depending on your site’s policies, put the files into one or more of the locations described in the following list:

- Store the files on another file system of any type.
- Store the files directly on removable media files.

  For information on removable media files, see the `request(1)` man page.
- If running the `archiver(1M)` on an ASM or ASM-QFS file system, store the files on a separate ASM or ASM-QFS file system that is being archived on a separate set of cartridges.

  This approach ensures that the disaster recovery files and metadata are archived separately from file system to which they apply. You might also consider archiving multiple backup copies for additional redundancy.

Observe the following precautions:

- Keep a written (nonelectronic) listing of where the disaster recovery files are kept.

  You can obtain lists of all directories containing removable media files by using the `sls`(1M) command. These listings can be emailed. For more information about obtaining file information, see the `sls(1M)` man page.
- Keep a written record of your hardware configuration.

- Do not assign the cartridges used to hold the removable media files to the archiver.

Restoring Files and Directories 2

This chapter describes how to restore individual files and directories.

Table 12. lists the tasks for restoring files and directories with cross references to where the procedures are located.

Table 11. Tasks for Restoring Files and Directories (Task Map)

Type of FileSystem	Where Described	Notes
ASM/QFS-Standalone	"Restoring Single Files and Directories With qfsdump(1M) Output" on page 27	The same procedure is used for both regular files and directories.
ASM or ASM-QFS	"Restoring Single Files and Directories With samfsdump(1M) Output" on page 28[1] "Restoring Files and Directories Without samfsdump(1M) Output (Task Map)" on page 30[2] "Tips for Retrieving Unarchived Files from ASM or ASM-QFS File Systems" on page 51[3] "To Restore a File Archived to Disk" on page 52	The first three procedures are for restoring files archived to tape or magneto optical cartridges from ASM or ASM-QFS file systems. These procedures are effective only if recent `samfsdump` files and recent archive copies of the files being restored are available.

1. The same procedure is used for regular files, segmented files, volume overflow files, and directories.
2. This section has a task map pointing to different procedures for when the file is a regular file, a segmented file, or a volume overflow file.
3. This section provides some additional information you should know when you need to try to recover files or directories for which archived copies are not available

Restoring Single Files and Directories With qfsdump(1M) Output

The following procedure uses the `qfsfsrestore(1M)` command to restore a file from a dump file created by the `qfsdump(1M)` command. If you are not already familiar with using the `qfsdump` command, see the section on creating `qfsdump` files in the *ASM, ASM-QFS, and ASM/QFS-Standalone Installation and Configuration Guide*.

Note: `qfsdump` and `qfsrestore` only work on an ASM/QFS-Standalone file system. When you have an ASM-QFS file system, use `samfsdump` as described in “Restoring Single Files and Directories With samfsdump(1M) Output” on page 28.”

To Restore Using a qfsdump File

1. List the name of the file or directory that you want restored.

```
# qfsrestore -t -f dump_file
```

2. Restore the file relative to the current directory.

   The *file_name* must exactly match the name of the file or directory as it was listed in the previous step.

```
# qfsrestore  -f  dump_file  file_name
```

Restoring Single Files and Directories With samfsdump(1M) Output

The example in the following procedure uses the `samfsrestore`(1M) command to restore a lost file from a dump file created by the `samfsdump` command.

Note: `samfsdump` and `samfsrestore` work on StorageTek’s ASM and ASM-QFS file systems. If needed, see “To Find ASM and ASM-QFS File Systems” on page 13.

To Restore Using a samfsdump(1M) File

This example restores a file `(pathname: /sam1/mary/mary1)` from a `samfsdump` metadata dump file called `/dump_sam1/041126`. The example creates a temporary restoration directory called `restore` in the `/sam1` file system.

1. Use the `mkdir(1)` command to create a directory in which to restore the files within a StorageTek ASM or ASM-QFS file system.

```
# mkdir restore
```

2. Use the `archive(1)` command with the `-r` option and `-n` option to prevent the archiver from archiving from this temporary directory location.

```
# archive -r -n restore
```

3. Use the `cd(1)` command to change to the temporary restoration directory.

```
# cd restore
```

4. Use the `samfsrestore(1M)` command with the `-t` and `-f` options to list the contents of the dump file.

   After the `-f` option specify the dump file's pathname.

Figure 2.

```
# samfsrestore -t -f /dump_sam1/041126
samfsrestore -t -f /dump_sam1/041126
./lost+found
./neptune
./mary
./fileA
./fileB
./fileC
./fileD
./fileE
./mary/mary1
./mary/mary2
./neptune/vmcore.0
./neptune/UNIX.0
./neptune/bounds
```

5. Search the listing from the previous step to verify that the lost file is in the dump file. If you find the file you are looking for, copy down the exact pathname shown in the output to use in the following step.

   In the previous screen example, the lost file called `mary1` is shown as residing in the `./mary` directory.

6. Use the `samfsrestore` command with the `-T` and `-f` options to restore the file's inode information to the current directory.

The *`filename`* must match exactly the pathname as it was listed in the previous output from Step 4. The following screen example shows using `samfsrestore` to retrieve the file `./mary/mary1` from the dump file `/dump_sam1/041126`.

```
# samfsrestore -T -f /dump_sam1/041126 ./mary/mary1
```

7. Use the `sls(1)` command with the `-D` option to list detailed information about the file, and verify that the inode information for the correct file has been retrieved.

The following screen example shows the `./mary/mary1` file's inode information.

Figure 3.

```
# sls -D ./mary/mary1
mary/mary1:
  mode: -rw-rw----  links:   1  owner: mary     group: sam
  length:        53  inode:          43
  offline;  archdone;
  copy 1: ---- Nov 17 12:35          8ae.1 xt 000000
  copy 2: ---- Nov 17 15:51       cd3.7f57 xt 000000
  access:      Nov 17 12:33  modification: Nov 17 12:33
  changed:     Nov 17 12:33  attributes:   Nov 17 15:49
  creation:    Nov 17 12:33  residence:    Nov 17 15:52
```

8. Use the `mv(1)` command to move the file to the desired location.

```
# cd mary
# mv mary1 /sam1/mary/
```

Restoring Files and Directories Without samfsdump(1M) Output (Task Map)

Table 12. lists the tasks for restoring various types of files when no `samfsdump(1M)` output is available.

Table 12. Tasks for Restoring Files When No samfsdump Output is Available (Task Map)

Type of File	Condition	Where Described
Regular File	An archiver log file exists with an entry for the file or you have output from the `sls` command with the `-D` option that lists the file.	"To Restore a Regular File Using Information From an Archiver Log or sls Command Output" on page 35 "To Restore a Regular File Using Information From an Archiver Log or sls Command Output" on page 35
Regular File	No archiver log file exists	"Restoring a Regular File Without Information From an Archiver Log" on page 38 "To Restore a Regular File Without Information From an Archiver Log" on page 38.

Table 12. Tasks for Restoring Files When No samfsdump Output is Available (Task Map)

Type of File	Condition	Where Described
Segmented File	An archiver log file exists with entries for the file.	"Restoring a Segmented File Using Information From an Archiver Log" on page 44 "To Restore a Segmented File Using Information From Archiver Log Entries" on page 45
Volume Overflow File	An archiver log file exists with entries for the file.	"Restoring a Volume Overflow File Using Information From an Archiver Log" on page 49 "To Restore a Volume Overflow File Using Information From an Archiver Log" on page 50

When you have an archiver log with an entry or entries for a missing file, see the following sections for how to interpret the information in the archiver log file and how to determine which of the above procedures to use:

- "Information Needed to Restore a File" on page 31
- "Determining Whether a File is a Regular File, a Segmented File, or a Volume Overflow File" on page 34

Information Needed to Restore a File

Table 13. shows the information needed when restoring a regular file.

Table 13. Information Needed for Restoring a Regular File

Definition	Field in Archiver Log Output	Field in Archive Copy Line in `sls -D` Output
media type	4	5
VSN (volume serial name)	5	6
position[1]	7	4

1. The position is the value on the left of the field with the format: *position.offset*.

If you can get the needed information about a regular file either from its archiver log entry or from output about the file from the `sls(1)` command with the `-D` option, you can restore the file with the `request(1M)` and `star(1M)` commands. As shown in the examples that follow, the `request` command is first used to create a file whose contents represent the contents of one or more pieces of removable media (which is sometimes referred to as a

"request file"). The `star` command is then used to extract the file, as shown in the following examples.

Example 1: Archiver Log

Figure 4. Typical Archiver Log Entry for a File on a Magneto-Optical Disk

```
A 96/01/05 10:55:56 mo v1 set_1.1 d2e.1 samfs2 770.11 test/
file3 0 0 0
```

In Figure 6., the media type `(mo)`, the file's position `(d2e)` and its `VSN (v1)` from the archiver log file entry are entered as arguments to the `request(1M)` command, which creates a temporary archive file (xxx) in another file system: `/sam3`. The example shows the change of directories in the Then the example shows the request file `/sam3/xxx` entered as an argument to the `star`(1M) command with the `-x` option, which extracts all the files from the archive file (including the lost file `file3`) into the `/sam2` directory.

Figure 5.

```
# request -p 0xd2e -m mo -v v1 /sam3/xxx
# cd /sam2

# star -x -b 32 -f /sam3/xxx
...
-rw-rw----  0/1     2673 May  1 15:41 1996 test/file3
...
tar: directory checksum error              <--- this is OK
```

Example 2: Comparing Archiver Log to sls -D Output

This example shows how you can obtain the needed information from either an archiver log entry or from output from the `sls(1)` command with the `-D` option for the file.

Figure 6. Typical Archiver Log Entry for a File on Tape

```
A 96/06/04 10:55:56 lt DLT001 set_1.1 286.1324f samfs1 770.11
tape_test/file4 0 0 0
```

In example above, the media type `(lt)` is shown in field 4, the VSN `(DLT001)` is shown in field 5, and the position (286) is shown in the left portion of field 7.

The following screen example shows the output from the `sls(1M)` command with the `-D` option for the file.

Figure 7.

```
# sls -D /sam1/tape_test/file4
/sam1/test/file4:
 mode: -rw-rw----  links:   1  owner: root      group: other
 length:    130543
 offline;
 copy 1:   Jun  4 10:55       286.1324f lt DLT001
 access:   May 24 16:55  modification: May 24 16:38
 changed:  May 24 16:38  attributes:   Jun  4 10:55
 creation: May 24 16:38  residence:    Jun  4 10:55
```

If an archive copy exists for the file, an archive copy line appears in the `sls -D` output below the `file states` line (as described in the `sls`(1) man page). In the example, the line that indicates the archive copy exists begins with `copy 1`. The file's position is shown in the left of field 4 (286), the file's type is shown in field 5 (lt) and the VSN is shown in field 6 (DLT001).

In the following screen example, the media type (`lt`), the file's position (`286`) and its VSN (`DLT001`) are entered as arguments to the `request`(1M) command, which creates a temporary archive file (`xxx`) in another file system: `/sam2`. The following screen shows how the `star`(1M) command can be used to reference the file on tape.

Note: You can ignore the directory checksum error.

Figure 8.

```
# request -p 0x286 -m lt -v DLT001 /sam2/file4
# cd /sam1
# star -xv -b 32 -f /sam2/file4
...
-rw-rw----  0/1   130543 May 24 16:38 1996 test/file4
...
tar: directory checksum error              <--- this is OK
```

- If you labeled the tape with a block size other than the default (16 kilobytes), you would use the block size in bytes divided by 512 (in place of the value 32) for the star command's `-b` option. You can see the tape block size by mounting the tape and observing either the `samu(1M)` utility's `t` display, the `samu` utility's `v` display (type CTRL-i for detail lines), or the output of the `dump_cat(1M)` command.

Determining Whether a File is a Regular File, a Segmented File, or a Volume Overflow File

This section shows how to determine from a missing file's archiver log file entries whether the file is a regular file, a segmented file, or a volume overflow file. You need this information to decide which of the restoration procedures to follow from "Restoring Files and Directories Without samfsdump(1M) Output (Task Map)" on page 30.

Regular File

Each *regular* file has a single entry in an archiver log. Figure 9. shows a typical entry for a regular file in an archiver log. In field 12 of the archiver log entry a regular file is identified with an `f`.

Figure 9. Archiver Log Entry for a Regular File

```
A 96/01/05 10:55:56 mo v1 set_1.1 d2e.1 samfs2 770.11 test/
file3 f 0 0
```

Segmented File

A *segmented* file is a file that has the segment attribute set and a *`segment_size`* specified using the `segment(1)` command. When a file has the segment attribute set, it is archived and staged in segment_size chunks. The length of the segment (`segment_size`) is shown in field 10 of the archiver log file in kilobytes.

For each segmented file, an archiver log has multiple entries. Figure 9. shows three entries for segmented file `seg/aaa`. Field 12 has a S indicating that the file type is *file segment*.

Figure 10. Archiver Log Entry for a Segmented File Volume Overflow File

```
A 2000/06/15 17:07:28 ib E00000 all.1 1276a.1 samfs4 14.5
10485760 seg/aaa/1 S 0 51

A 2000/06/15 17:07:29 ib E00000 all.1 1276a.5002 samfs4 15.5
10485760 seg/aaa/2 S 0 51

A 2000/06/15 17:07:29 ib E00000 all.1 1276a.a003 samfs4 16.5
184 seg/aaa/3 S 0 51
```

A volume overflow file is one that is written on multiple volumes. For a volume overflow file, an archiver log has multiple entries, one for each section of the

file. The following screen example shows two entries for the two sections of file `big2d`.

Figure 11. Archiver Log Entry for a Volume Overflow File

```
A 2001/10/31 09:47:29 lt CFX600 arset1.1 3668e.1 samfs9
71950.15 2011823616 testdir1/big2d f 0 43

A 2001/10/31 09:47:29 lt CFX603 arset1.1 3844a.0 samfs9
71950.15 1209402048 testdir1/big2d f 1 41
```

The `big2d` file is identified as a volume overflow file because it has two entries, the `f` in field 12 indicates that the entry is for a regular file, and the `0` and the `1` in field 13 are section numbers. Field 5 shows that the file starts on `VSN CFX600` and overflows to `VSN CFX603`.

Summary of Differences

summarizes the defining characteristics of regular, segmented, and volume overflow files.

Table 14. Defining Characteristics of Regular, Segmented, and Volume Overflow Files

A file is a regular file if . . .	It has only a single entry and the file type in field 12 is f.
A file is a segmented file if . . .	It has multiple entries, the VSN in field 5 is identical in both entries, the file type in field 12 is S, and the section numbers in field 13 for both entries are the same.
A file is a volume overflow file if . . .	It has multiple entries, the VSN in field 5 is different for each entry, the file type in field 12 is f, and the section numbers in field 13 are different for each entry.

To Restore a Regular File Using Information From an Archiver Log or sls Command Output

Note: For the procedure to work, the ASM or ASM-QFS file system must be mounted.

1. Log in as or switch users to root.
2. Find the media type, the file's position, and the VSN.

a. If you have an archiver log, use `cat`(1M) or another command to search the archiver log file for an entry for the missing file.

The following screen example shows the sample entry for a file that is archived on a tape followed by a sample entry for a file archived on an optical disk.

```
# cat
...
A 96/06/04 10:55:56 lt DLT001 arset0.1 286.1324f samfs1
770.11 tape_test/file4 0 0 0
A 96/01/05 10:55:56 mo v1 set_1.1 d2e.1 samfs2 770.11
mod_test/file3 0 0 0
```

If needed, see Table 13. for definitions of the fields in the archiver log file.

b. If you have output from the `sls` command with the `-D` option about the missing file, search that output.

The following screen example shows output from the `sls(1M)` command with the `-D` option for the `tape_test/file4`.

Figure 12.

```
# sls -D /sam1/tape_test/file4
/sam1/test/file4:
 mode: -rw-rw----  links:   1  owner: root      group: other
 length:    130543
 offline;
 copy 1:   Jun  4 10:55       286.1324f lt DLT001
 access:   May 24 16:55  modification: May 24 16:38
 changed:  May 24 16:38  attributes:   Jun  4 10:55
 creation: May 24 16:38  residence:    Jun  4 10:55
```

c. Record the media type, the file's position, and the VSN to use as input to the `request(1M)` command in the next step.

Table 15.

media type	
position	
VSN	

3. Use the `request(1M)` command with the `-p` option using the position from the archiver log to position to the beginning of the `tar(1)` header for the file.

   Use hexadecimal notation, prefacing the position number after the `-p` option with `0x`.

The following screen example shows two request commands, the first to create a request file with the contents of the archive containing the example file that is on tape and the second to create a request file with the contents of the example file that is on optical disk.

Figure 13.

```
# request -p 0x286 -m lt -v DLT001 /sam1/xxxx <-For a
file on tape
# request -p 0xd2e -m mo -v v1 /sam2/xxxx <-For a file on
magneto-optical disk
```

4. Use the `star(1M)` command to extract the file.

Note: The `star(1M)` command restores all the files from the archive file that you are pointing to with the request file.

Figure 14.

```
# cd /sam1
# star -xv -b 32 -f /sam1/xxxx <-For the file on tape

...
file4
...
tar: directory checksum error              <--- this is OK

# cd /sam2
# star -xv -b 32 -f /sam2/xxxx <-For the file on magneto-
optical disk
...
file3
...
tar: directory checksum error              <--- this is OK
#
```

5. Use the `sls(1M)` command to verify that you have extracted the lost file.

The following screen example shows is the `sls -Di` output for the file on the optical disk.

Figure 15.

```
# sls -Di /sam2/mod_test/file3
/sam2/mod_test/file3:
mode: -rw-rw----  links:   1  owner: root      group:
other
length:       468  admin id: 7 inode:      161.2
copy 1:---- May  1 15:41       286.1324f mo v1
```

Figure 15.

```
access:   May  1 16:50  modification: May  1 15:41
changed:  May  1 15:40  attributes:   May  1 15:44
creation: May  1 15:40  residence:    May  1 16:50
```

Restoring a Regular File Without Information From an Archiver Log

If you do not have an archive log available with an entry for the file, you can use the procedure "To Restore a Regular File Without Information From an Archiver Log" on page 38.

Note: If the only resources available consist of a cartridge containing archive copies and a Solaris system without ASM or ASM-QFS software installed, you can still restore the file by starting this procedure with Step 3.

You can perform the procedure "To Restore a Regular File Without Information From an Archiver Log" on page 38 using either an automated library or a manually mounted, standalone drive, under the following conditions:

- If you are using an automated library, the automated library daemon must be active on the system.
- If you are using a manually mounted, standalone drive, make sure that `/kernel/drv/st.conf` is correctly configured for the tape drive that you are using. For more information about performing this task, see how to add tape support to the `st.conf` file in the *ASM, ASM-QFS, and ASM/QFS-Standalone Installation and Configuration Guide*.

To determine which cartridge contains the missing file, you need to examine only those volumes that are assigned to the archive set for the file in question. You can use the `-t` option to `tar` or `star` repeatedly on each volume as described in the procedure "To Restore a Regular File Without Information From an Archiver Log" on page 38 to find out which volume contains the archive copy. When you have found the archive copy of the file, you then use the `-x` option to `tar` or `star` to extract the file.

To Restore a Regular File Without Information From an Archiver Log

1. (Optional) Prevent the ASM or ASM-QFS software from using the tape drive.

Note: If you are using a manually mounted, standalone drive, skip this step.

You can use either the `samu(1M)` command with the `:unavail` *`eq`* `option`, the `samcmd(1M)` command with the `unavail` *`eq`* option, the

devicetool(1M) or the libmgr(1M) command. For the samu and samcmd commands, specify the equipment ordinal of the drive as *eq*. The Equipment Ordinal for each device is specified in the mcf(4) file.

The following screen example shows the use of the samcmd command with the unavail subcommand when the drive number is 51.

```
# samcmd unavail 51
```

2. (Optional) Use the samload(1M) command to load the desired volume into the drive.

Note: If you are using a manually mounted, standalone drive, skip this step.

For the command line options to use, see the man(1) page. The following screen example shows the use of the samload command to load the cartridge that is in slot 3 of library 50 into the drive with equipment ordinal 51

```
# samload 50:03 51
```

3. Use the mt(1M) command to rewind the tape.

The following example shows how to do this using the mt(1M) command. If your tape drive is not /dev/rmt/2, substitute the correct name in the following examples.

```
# mt -f /dev/rmt/2cbn rewind
```

Note: Because the device name used in these examples ends with the n (no rewind) option, each of the commands in the following steps examines the next file on the tape.

4. Use od(1M) or another command to examine the ANSI label on the cartridge, and find the line that starts with 0000240.

The first file on the cartridge is the ANSI label. The information you are looking for appears on the line that starts with 0000240.

l

Figure 16. ANSI Label

```
# od -c /dev/rmt/2cbn
0000000   V   O   L   1   X   X   X
0000020                               S   A   M   -   F
S       1
0000040   .   0
0000060
0000100
4
0000120   H   D   R   1
0000140                                           0   0
0   1   0
0000160   0   0   1   0   0   0   1   0   0       2   4   9
0   9
0000200                                               S
A   M   -
0000220   F   S       1   .   0
0000240   H   D   R   2       1   6   3   8   4
1
0000260                                               2
0   g 031
0000300
*
0000360
```

5. Note the five characters that appear after `H D R 2` on the line that starts `0000240`.

   The five characters that appear after `H D R 2` on the line that starts with `0000240` are the five bottom digits of the block size, in decimal. In the previous screen example, the characters are `1 6 3 8 4`.

6. Use the five bottom digits of the block size to determine the block size used on the media.

   Locate the bottom five digits of the block size in the left column of the following table. For the `dd(1M)` command, the block size is found in the second column. For both the `star(1M)` and `tar(1)` commands, the block size is specified in units of 512-byte blocks, which are shown in column 3.

Table 16. Block Sizes Corresponding to the Bottom Five Digits of Block Size in the ANSI Label

Bottom Five Digits of Block Size	Block Size for `dd`**(1)**	512-byte Blocks for `tar(1)` and `star(1M)`
16384	16 kilobytes	32 blocks
32768	32 kilobytes	64 blocks
65536	64 kilobytes	128 blocks
31072	128 kilobytes	256 blocks
62144	256 kilobytes	512 blocks
24288	512 kilobytes	1024 blocks
48576	1024 kilobytes	2048 blocks
97152	2048 kilobytes	4096 blocks

Note: In the following screen examples, all files are archived twice, so each file is inspected twice.

7. If the `star(1M)` command is available, enter it with the number of 512-byte blocks obtained in the previous two steps to find the file in the archive.

   You can download the `star` command from an ASM or ASM-QFS system onto any Solaris system. If you do not have access to the `star` command, you can use the `dd(1M)` command with the `tar(1)` command, as shown in Step 8.

Note: `star` files have an extended maximum file size of 1 `Tbytes-1. tar` and `star` files have compatible formats only at file sizes less than or equal to (≤) 8Gbytes-1. At larger than (≥) 8Gbytes, the formats of `star` and `tar` files are not compatible. Therefore, you must use the `star` command to read archives larger than 8Gbytes-1.

The following screen example shows the s`tar` command being used to examine the first `tar` file. The block size for both the `star`(1M) and `tar`(1) commands is specified in units of 512-byte blocks. (The number 32 used after `-b` in the example is the number of 512-byte blocks that corresponds to the number 16384 in the ANSI label in Step 4, from the table in Step 6.)

Figure 17.

```
# star -tv -b 32 -f /dev/rmt/2cbn
-rw-rw----  0/1   102564 Sep  6 13:02 1996 test
6+1 records in
11+1 records out
```

The following screen example shows the same command examining the next `tar(1)` file.

Figure 18.

```
# star -tv -b 32 -f /dev/rmt/2cbn
or
# dd if=/dev/rmt/2cbn ibs=16k obs=10k conv=sync | tar
tvf -
-rw-rw----  0/1   102564 Sep  6 13:02 1996 test
6+1 records in
11+1 records out
```

The following shows two copies of another file being examined.

Figure 19.

```
# star -tv -b 32 -f /dev/rmt/2cbn
-rw-rw----  0/1   102564 Sep  6 13:02 1996 test2
6+1 records in
11+1 records out
# star -tv -b 32 -f /dev/rmt/2cbn
-rw-rw----  0/1   102564 Sep  6 13:02 1996 test2
6+1 records in
11+1 records out
```

The following example shows the end of the tape has been reached:

Figure 20.

```
# star -tv -b 32 -f /dev/rmt/2cbn
0+0 records in
0+0 records out
tar: blocksize = 0
# mt -f /dev/rmt/2cbn status
Other tape drive:
 sense key(0x13)= EOT   residual= 0   retries= 0
 file no= 5   block no= 0
```

8. If the `star(1M)` command is not available, use the `dd(1M)` and `tar(1)` commands to examine the archives.

   The following screen example shows the `dd` command being used to examine the first `tar` file. The value 16k used for the input block size (`ibs=`) is the number in the third column of the table in Step 6 that corresponds to the number 16384 in the ANSI label in Step 4.

.

Figure 21.

```
# dd if=/dev/rmt/2cbn ibs=16k obs=10k conv=sync | tar
tvf -
-rw-rw----  0/1   102564 Sep  6 13:02 1996 test
6+1 records in
11+1 records out
```

The following screen example shows the same command examining the next `tar(1)` file.

Figure 22.

```
# dd if=/dev/rmt/2cbn ibs=16k obs=10k conv=sync | tar
tvf -
-rw-rw----  0/1   102564 Sep  6 13:02 1996 test
6+1 records in
11+1 records out
```

The following shows the examination of two copies of another file.

Figure 23.

```
# dd if=/dev/rmt/2cbn ibs=16k obs=10k conv=sync | tar
tvf -
-rw-rw----  0/1   102564 Sep  6 13:02 1996 test2
6+1 records in
11+1 records out
# dd if=/dev/rmt/2cbn ibs=16k obs=10k conv=sync | tar
tvf -
-rw-rw----  0/1   102564 Sep  6 13:02 1996 test2
6+1 records in
11+1 records out
```

The following example shows the end of the tape has been reached:

Figure 24.

```
# dd if=/dev/rmt/2cbn ibs=16k obs=10k conv=sync | tar
tvf -
0+0 records in
0+0 records out
tar: blocksize = 0
# mt -f /dev/rmt/2cbn status
Other tape drive:
 sense key(0x13)= EOT   residual= 0   retries= 0
 file no= 5   block no= 0
```

Note: You might receive errors during this process. The following error indicates that the block size you selected does not match that of the tape:

```
read:  not enough space
```

Correct the block size and try again.

9. When you find the missing file in an archive, use the `-x` option with either the `star` command alone or the `dd` command with the `tar` command to extract the files from that archive.

Note: You can ignore the `dd: read error` in the first line of output.

Figure 25. Using the `dd` and `tar` Commands or the or `star` Command by Itself to Extract a File

Figure 26.

```
# dd if=/dev/samst/c0t1u0 bs=1k iseek=3374 of=/tmp/junk
count=10
dd: read error: I/O error    <---- This is OK!
8+0 records in
8+0 records out
# tar xvf /tmp/junk
or
# star -xv -f /tmp/junk
tar: blocksize = 1
-rw-rw---- 0/1 2673 May 1 15:41 1996 dir3/dir2/file0
-rw-rw---- 0/1 946 May  1 15:41 1996 dir3/dir1/file1
-rw-rw---- 0/1 468 May  1 15:41 1996 dir1/dir3/file0
```

Restoring a Segmented File Using Information From an Archiver Log

When a segmented file is archived or staged, it is archived and staged in chunks. For each segmented file, an archiver log has multiple entries.

If an archiver log file exists, you can search the archiver log for the multiple entries for the missing segmented file. (See “To Set Up Archiver Logging” on page 23, if needed.)

If you can find entries for a missing segmented file in an archiver log, you can use the file’s position, segment size, VSN, and media type, to restore the file using the `request(1M)` and `star(1M)` commands. The procedure is described in “To Restore a Segmented File Using Information From Archiver Log Entries” on page 45.

If needed, see Table 13. for definitions of the fields in the archiver log file.

The segmented file named `aaa` is used in the examples in this section and in the procedure. The following screen example show three entries for segmented file `aaa` in the archiver log file.

Figure 27.

```
A 2000/06/15 17:07:28 ib E00000 all.1 1276a.1 samfs4 14.5
10485760 seg/aaa/1 S 0 51
A 2000/06/15 17:07:29 ib E00000 all.1 1276a.5002 samfs4 15.5
10485760 seg/aaa/2 S 0 51
A 2000/06/15 17:07:29 ib E00000 all.1 1276a.a003 samfs4 16.5
184 seg/aaa/3 S 0 51
```

Table 17. gives you a place to record the information to use when restoring a segmented file.

Table 17. Archiver Log Entry Information Needed for Restoring a Segmented File

Field	Definition	Comments
4	media type	
5	VSN	
7	position	
12	type of file	The `S` in field 12 indicates that the entry is for a segment of a segmented file.
11	name of file	In the file name field of the three example entries, the three segments of the file `aaa` are identified as `seg/aaa1`, `set/aaa/2`, and `seg/aaa/3`.
10	length	For entries for file segments, the segment size (length) is shown. You specify the segment size of the first segment on the `segment`(1M) command line to recover a segmented file.

To Restore a Segmented File Using Information From Archiver Log Entries

Note: Free space must be available in the file system equal to two times the size of the file to be recovered.

1. Find the archiver log entries for the segmented file by the *filesystem name* (from field 8) and *file name* (from field 11).

   The following screen example show three entries for segmented file `aaa` in the `archiver.log` file.

Figure 28.

```
A 2000/06/15 17:07:28 ib E00000 all.1 1276a.1 samfs4 14.5
10485760 seg/aaa/1 S 0 51
A 2000/06/15 17:07:29 ib E00000 all.1 1276a.5002 samfs4 15.5
10485760 seg/aaa/2 S 0 51
A 2000/06/15 17:07:29 ib E00000 all.1 1276a.a003 samfs4 16.5
184 seg/aaa/3 S 0 51
```

If needed, see Table 13. for definitions of the fields in the archiver log file.

In all the lines in the previous screen example, the filesystem name is `samfs4`. Each segment has its own entry and filename: `seg/aaa/1`, `seg/aaa/2`, and `seg/aaa/3`.

2. Note the file's *position* (from the position indicator portion to the left of the dot in field 7), the *media type* on which the file is stored (from field 4), and the *VSN* (from field 5), to use as input to the `request(1M)` command in Step 3. Also note the segment size (from the length field 10) to be used as input to the `segment(1M)` command in Step 8.

In the first line in the previous screen example:

- The media type is `ib` (for the IBM 3590 tape drive).

  For the supported media types, see the `mcf`(4) man page.

- The file's position is `1276a`.
- The VSN is `E00000`.
- The segment size is `10485760`.

Table 18.

	Field	Missing File's Value
media type	4	
position	portion of field 7 to the left of the dot (.)	
VSN	5	
segment size	10	

3. Enter the `request(1M)` command to create a removable media file that points to the segments.

Supply the following:

- *position* number after the `-p` option in hexadecimal notation, prefacing the position number with `0x`.
- *media type* after the -m option
- *VSN* after the -v option
- *filename* for a removable media file

The following screen example uses the values from the example lines in Step 1.

```
# request -p 0x1276a -m ib -v E00000 /sam3/rmfile
```

4. Enter the `star(1M)` command with the name of the file created in the previous step to read the segments from tape onto the disk.

Figure 29.

```
# star xvbf 512 /sam3/rmfile
seg/aaa/1
seg/aaa/2
seg/aaa/3
```

5. Change directories into the directory where the segmented files reside.

   The following screen example shows segmented files `1`, `2`, and `3` in the `seg/aaa` directory.

Figure 30.

```
# cd seg
# pwd
/sam3/seg
# ls -l
total 8
drwxrwx---   2 root     other       4096 Jun 15 17:10 aaa/
# ls -l aaa
total 40968
-rw-rw----   1 root     other    10485760 Jun 15 17:06 1
-rw-rw----   1 root     other    10485760 Jun 15 17:06 2
-rw-rw----   1 root     other         184 Jun 15 17:07 3
# pwd
/sam3/seg
# cd aaa
# pwd
/sam3/seg/aaa
```

6. Use the `ls(1)` and `sort(1)` commands to list and sort the numbered files in numerical order, and use the `cat`(1M) command to join the files.

   The temporary file created in this step is not segmented.

Figure 31.

```
# ls | sort -n | xargs cat > ../bbb
```

7. Change to the directory above where the numbered files reside, and then use the `rm(1)` command to remove the numbered files.

Figure 32.

```
# cd ..
# pwd
/sam3/seg
# ls -l
total 41000
drwxrwx---   2 root     other       4096 Jun 15 17:10 aaa/
-rw-rw----   1 root     other    20971704 Jun 15 17:11 bbb
# ls -l aaa
total 40968
-rw-rw----   1 root     other    10485760 Jun 15 17:06 1
-rw-rw----   1 root     other    10485760 Jun 15 17:06 2
-rw-rw----   1 root     other         184 Jun 15 17:07 3
# rm -rf aaa
```

8. Enter the `touch(1M)` command to create an empty file.

Figure 33.

```
# touch aaa
```

9. Use the `segment(1M)` command to set the segment attribute on the file created in Step 8.

   Enter the `segment` command with the `-l` option following by the segment length in megabytes followed by `m` followed by the filename of the empty file created in the previous step.

   Convert the segment length (from field 10 of the archiver log file entry) to megabytes by dividing 1048576. For example, the segment length in the archiver log entry example in Step 2 is 10485760. Dividing the segment length by 1048576 gives 10 megabytes, which is entered as `-l 10m` in the following screen example.

Figure 34.

```
# segment -l 10m aaa
```

10. Copy the temporary file created in Step 6 into the empty file created in Step 8, and remove the temporary file.

Figure 35.

```
# cp bbb aaa
# rm bbb
```

11. Enter the `sls(1)` command with the `-2K` option to list the segments of the segmented file in two lines of output.

Figure 36.

```
# sls -2K aaa
-rw-rw----   1 root     other     20971704 Jun 15 17:12 aaa
---------- ----- sI {3,0,0,0}
-rw-rw----   1 root     other     10485760 Jun 15 17:12 aaa/1
---------- ----- sS
-rw-rw----   1 root     other     10485760 Jun 15 17:12 aaa/2
---------- ----- sS
-rw-rw----   1 root     other          184 Jun 15 17:12 aaa/3
---------- ----- sS
```

Restoring a Volume Overflow File Using Information From an Archiver Log

A volume overflow file is a file that is written on multiple volumes. If an archiver log file exists, you can search the archiver log for entries for the missing file. (See “To Set Up Archiver Logging” on page 23, if needed.) If you can find entries for a missing volume overflow file in an archiver log, you can use the file’s position, segment size, VSN, and media type, to restore and reassemble the file using the `request(1M)`, `star(1M)`, `dd(1M)`, and `cat(1)` commands. The procedure is described in “To Restore a Volume Overflow File Using Information From an Archiver Log.”

If needed, see Table 13. for definitions of the fields in the archiver log file.

The volume overflow file named `big2d` is used in this section and in the procedure. The following screen example shows two entries for the two sections of file `big2d` in the `archiver.log` file

```
A 2001/10/31 09:47:29 lt CFX600 arset1.1 3668e.1 samfs9
71950.15 2011823616 testdir1/big2d f 0 43

A 2001/10/31 09:47:29 lt CFX603 arset1.1 3844a.0 samfs9
71950.15 1209402048 testdir1/big2d f 1 41
```

The file is identified as a volume overflow file with two sections because the `f` in the third-to-last field indicates that the entry is for a regular file, and the `0` and the `1` in the second-to-last fields are section numbers. The fifth field shows that the file starts on VSN `CFX600` and overflows to information about `CFX603`.

The following procedure assumes that free space is available in the file system equal to two times the recovered file.

To Restore a Volume Overflow File Using Information From an Archiver Log

Note: Free space must be available in the file system equal to two times the size of the file to be recovered.

1. Use `vi(1M)` or another command to examine the archiver log file that contains an entry for the file you are trying to recover.

For example, the following is the archiver log file for `big2d`:

```
A 2001/10/31 09:47:29 lt CFX600 arset1.1 3668e.1 samfs9
71950.15 2011823616 testdir1/big2d f 0 43

A 2001/10/31 09:47:29 lt CFX603 arset1.1 3844a.0 samfs9
71950.15 1209402048 testdir1/big2d f 1 41
```

2. Use the `request(1M)` command to create a removable media file that points to each section.

For example:

```
# request -p 0x3668e -m lt -v CFX600 /sam3/rmfile.0
# request -p 0x3844a -m lt -v CFX603 /sam3/rmfile.1
```

3. Use the `cd(1M)` and `star(1M)` commands to recover the first section.

   A block size of 128 kilobytes is assumed for both tapes.

```
# cd /sam3/temp
# star xvbf 256 /sam3/rmfile.0
testdir1/big2d
star: Unexpected EOF on archive file
star: Error exit delayed from previous errors
```

4. Use the `mv(1M)` command to move this first section to another name, for convenience.

For example, the following command moves the file sections to `big2d.0`, `big2d.1`, and so on.

```
# mv testdir1/big2d testdir1/big2d.0
```

5. Use the `dd(1M)` command to recover the remaining sections.

   For example:

```
# dd if=rmfile1 of=testdir1/big2d.1 files=1 ibs=128k
9228+0 records in
2362368+0 records out
```

Repeat this step for each section after the first one.

6. Use the `ls(1M)` command to examine the output and ensure that all pieces of the file are on the disk.

```
# ls -l testdir1
total 6291712
-rw-rw----   1 root     sam      2011823616 Oct 31 08:47
big2d.0
-rw-rw----   1 root     other    1209532416 Nov  1 11:20
big2d.1
```

7. Use the `cat(1M)` command to reassemble the files.

```
# cat big2d.0 big2d.1 > big2d
# sls -D big2d
big2d:
mode: -rw-rw----  links:   1  owner: root      group: other
length: 3221356032  admin id:      0  inode:    71949
access:      Nov  1 12:59  modification: Nov  1 12:24
changed:     Nov  1 12:24  attributes:   Nov  1 11:25
creation:    Nov  1 11:25  residence:    Nov  1 11:25
```

Tips for Retrieving Unarchived Files from ASM or ASM-QFS File Systems

Unarchived files that resided within an ASM or ASM-QFS file system may not be recoverable. The following list describes some facts that might help you to retrieve unarchived files:

- If the samfsdump(1M) method was used to dump and back up metadata, the samfsrestore(1M) command identifies files without archive copies and flags them as damaged.
- ASM and ASM-QFS log files cannot help you determine which files were not archived and were therefore lost between the last archiver run and the system outage. However, you can determine the files that might not have been archived by analyzing the archiver.cmd file for archiving directives and intervals. If all files are eligible for archiving, you can find the age of the oldest unarchived (lost) files in the archiver.cmd file's contents.
- You can use the -l and -v options with the archiver(1M) command to generate information you can use to determine whether volumes were available to archive each archive set's data before the outage. Lack of sufficient volumes can prevent archiving of data in one or more archive sets. For information about the archiver(1M) command, see the sam-archiverd(1M) man page.
- If you are recovering files straight from a backup tape in tar(1) format, the files are restored to their locations according to the information on the tape. The path name is relative to the mount point of the file system. If any files have been moved within the system since the archive copies were created, they are restored to their original locations, not to their new locations.
- You can use the sfind(1M) command line to identify all files in a file system that are not archived. The following screen example finds all unarchived files associated with the /sam1 mount point.

```
# sfind /sam1 \! -archived
```

To Restore a File Archived to Disk

1. Use the sls(1) command with the -D option to find the volume serial name (VSN) for the disk where the file is archived.

Figure 37.

```
# sls -D /sam1/dir1/dir3/filea
/sam1/dir1/dir3/filea:
mode: -rw-r-----  links:   1  owner: root      group: other
length:   1664041  inode:     1331
archdone;
copy 1: ---- Jan 22 02:14         0.0    dk disk02
copy 2: ---- Jan 22 02:36     995f1.1    mo opt02b
access:      Jan 21 09:34  modification: Jan 21 09:34
changed:     Jan 21 09:34  attributes:   Jan 21 09:34
creation:    Jan 21 09:34  residence:    Jan 21 09:34
```

This example shows output from the `sls(1)` command for `filea`, which has one copy (`copy 1`) archived to disk. In the example output, the last field in the line for `copy 1` shows `disk02` as the VSN.

2. Use `vi(1)` or another command to find the pathname defined for the VSN in the `diskvols.conf(4)` file.

   The following example shows two disk volumes defined for receiving archive copies in the `/etc/opt/SUNWsamfs/diskvols.conf file.`

```
# vi /etc/opt/SUNWsamfs/diskvols.conf
disk01   /sam_arch1
disk02   mars:/sam_arch3/proj_3
```

   The output shows that `VSN disk02` points to the destination path `/sam_arch3/proj_3` on remote server `mars`.

3. Use the `rsh(1)` and `ls(1)` commands to verify the existence of the file.

```
# rsh mars:ls -al /sam_arch3/proj_3/dir1/dir3/filea
```

4. Use the `ftp(1)` command or `rcp(1)` command to restore the file.

```
# rcp mars:/sam_arch3/proj_3/dir1/dir3/filea .
```

Salvaging Damaged Volumes

3

This chapter describes how to restore data from tapes or magneto-optical disks that are not usable in an ASM or ASM-QFS environment. This procedures in this chapter describe what to do when a volume is partially corrupted, accidentally relabeled, has a destroyed label, or is entirely destroyed. The procedures in this chapter describe how to recover data both when archive copies are available and when there are no other copies available.

Before attempting the procedures in this chapter, determine whether or not the volume can be read by using software other than ASM or ASM-QFS tools. Try reading the volume in multiple drives, or try using the `tar(1)` command.

This chapter covers the following topics:

Recovering Data From a Tape Volume

The procedures for recovering data from a tape volume differ depending on the nature of the damage and whether or not additional archive copies of the volume's files are present on another tape. This section describes how to recover data in the following scenarios:

- Tape volume is damaged, and alternative archive copies are available.
- Tape volume is partially corrupt, and no alternative archive copies are available.
- Tape volume is accidentally relabeled, and no alternative archive copies are. available.
- Neither the ASM software nor the ASM-QFS software can read the tape volume label and no alternative archive copies are available.

Damaged Tape Volume—Other Copies Available

The ASM and ASM-QFS storage and archive manager allows you to make up to four archive copies of each online file. By default, only one copy is made, but StorageTek recommends that you make at least two copies, preferably to physically different archive media.

When an alternative archive copy is available, the recovery procedure includes a step for rearchiving all archive copies currently stored on the damaged volume before dispensing with the damaged volume. The new archive copies are made from the available alternative archive copy.

To Recycle a Damaged Tape—Other Copies Available

Use this procedure if alternative archive copies exist on volumes that are stored on-site and are available for staging.

1. Export the damaged volume from the tape library, and flag it as unavailable in the historian catalog.

Enter the `export(1M)` and `chmed(1M)` commands as shown in the following screen example, specifying the media type (*`mt`*) and `VSN (`*`vsn`*`)` of the damaged volume.

```
# export mt.vsn
# chmed +U mt.vsn
```

2. Flag the unavailable volume for recycling.

   Use the `chmed(1M)` command and specify the media type (*`mt`*) and the `VSN (`*`vsn`*`)` of the damaged volume.

```
# chmed +c mt.vsn
```

3. Set the `-ignore` option for the library in the `recycler.cmd` file.

   The following screen example shows the `-ignore` option set on the `lt20` library. See the `recycler-cmd(4)` man page for more information about the `-ignore` option.

```
# vi /etc/opt/SUNWsamfs/recycler.cmd
logfile = /var/adm/recycler.log
lt20 -hwm 75 -mingain 60 -ignore
:wq
```

4. Run the `sam-recycler(1M)` command with the `-x` option from the command line.

```
# sam-recycler -x
```

When the recycler runs, it does not select any volumes for recycling other than the volume you have marked as unavailable. The recycler identifies all active archive copies on this volume and flags those archive copies for rearchiving.

The next time the archiver runs, the archive copies marked for rearchiving will be written to new volumes.

After the archive copies have been written to new volumes, the damaged volume you are recycling is considered to be drained of active archive copies.

5. Dispense with the volume.

After the damaged volume is drained of active archive copies, you can dispense with the volume. How you dispense with it depends on the nature of the damage. Use the following guidelines:

- If the tape was accidentally relabeled, use the `tplabel(1M)` command to relabel the volume.
- If the tape label is unreadable, use the `tplabel(1M)` command to relabel the volume.
- If relabeling the volume fails, export the volume from the historian and dispose of the tape.

If the tape is either partially corrupt or completely destroyed, it is possible (but not recommended) to reuse the tape VSN after the volume has been exported from the historian catalog.

Damaged Tape Volume—No Other Copies Available

If a tape volume is partially corrupt, it is possible to recover data from the parts of the tape volume that are not corrupt. This process is not an exact science, and it requires some trial and error to recover as much data as possible.

Errors logged in the device log can help you determine the area of a tape that is damaged. The `archive_audit(1M)` command can be used to generate the position and offset information for all archived files for a specific file system. You can use this position and offset information to help determine which archive copies are written to an area of a tape that is damaged.

To Recover Files From a Damaged Tape—No Other Copies Available

1. Use the `archive_audit(1M)` command to generate a list of all files with archive copies on the partially corrupt tape volume.

   Use the command syntax shown in the following screen example, specifying the file system's mount point, the `VSN (`*`vsn`*`)` of the volume, and an output file name.

```
# archive_audit /mount_point | grep vsn > filename
```

2. Edit the output file from the `archive_audit(1M)` command in the previous step, deleting the lines for the files in the damaged area, and saving the list of deleted files for inspection in Step 3.
3. Use the list of files with archive copies that cannot be accessed (the ones that are written in the area of the tape determined to be damaged) to determine if any of the files are still on the disk.

Files that are not on disk cannot be recovered. These unrecoverable files can be removed from the file system.

4. Edit and run the `stageback.sh` script on the `archive_audit` output file you edited in Step 2.

   The `stageback.sh` script can stage each file from `archive_audit` output, set it to `no-release`, and mark the file for rearchiving.

   See below for information about the `stageback.sh` script.

   a. Open the `/opt/SUNWsamfs/examples/stageback.sh` file for editing.

```
# cd /opt/SUNWsamfs/examples
# vi stageback.sh
```

   b. Find the section that begins with `# echo rearch $file`.

```
# echo rearch $file
#
# Edit the following line for the correct media type and VSN
#
# eval /opt/SUNWsamfs/bin/rearch -m media -v VSN $file
```

   c. In the section shown in the previous screen example, replace the word “media” with the media type (*mt*) and the word “VSN” with the VSN of the damaged volume, which are the same as the VSNs in Step 1.
   d. Remove the pound sign from the beginning of the lines in the section shown in Step b.

```
 echo rearch $file

# Edit the following line for the correct media type and VSN

eval /opt/SUNWsamfs/bin/rearch -m media -v VSN $file
```

   e. Save and quit the file.
   f. Run the `stageback.sh` script.

Relabeled Tape Volume—No Other Copies Available

The ASM and ASM-QFS software cannot read beyond the EOD. If a tape is accidentally relabeled, the only possibility for recovering any data is to contact the tape manufacturer to determine if they offer a method for reading beyond EOD.

If the tape manufacturer can provide a mechanism for reading beyond EOD, you can recover the data by combining that process with the procedure for recovering files from a tape volume with a label not readable by the ASM or ASM-QFS software. This procedure is described under “Unreadable Tape Label—No Other Copies Available” on page 59.

Unreadable Tape Label—No Other Copies Available

Whenever the ASM or ASM-QFS software receives a request to mount a tape volume into a drive, one of the first actions taken is to verify the tape label written on the tape. If the tape label cannot be read, the ASM and ASM-QFS software cannot use the tape for staging or archiving activities.

The `tarback.sh(1M)` script is used to recover data from a tape that has a label that cannot be read. The shell script automates the process of recovering data written to a tape, using the `star(1M)` command to read each archive file written on a specific tape volume. The file data is read back onto disk (into an ASM, ASM-QFS, or UFS file system) as data. File data recovered in this manner can then be moved to the appropriate location in the ASM or ASM-QFS file system. It must then be archived as new data.

To Recover Files From a Tape Whose Label is Unreadable

1. If you are using this process to recover file data from several tapes, disable any currently occurring recycling.

   When recycling is going on, data on the tape volumes may be inaccessible.

2. Use the `cp(1M)` command to copy the `tarback.sh` file to a working location.

   For example, the following command copies the script from the default location `/opt/SUNWsamfs/examples/tarback.sh` to `/var/tarback.sh`.

```
# cp /opt/SUNWsamfs/examples/tarback.sh /var/tarback.sh
```

3. Enter the `samcmd(1M)` command with the `unavail` command to make the tape drive unavailable.

To prevent the tape drive from being used for staging and archiving activities, use the syntax shown in the following screen example. Specify the Equipment Ordinal of the drive, as specified in the mcf(4) file, for *eq*.

```
# samcmd unavail eq
```

4. Edit the working copy of the tarback.sh(1M) script to specify the variables shown in the following table.

Table 20. Variables to Specify in the tarback.sh(1M) Script

Variable	Definition
EQ="*eq*"	The Equipment Ordinal of the tape drive as defined in the mcf file.
TAPEDRIVE="*path*"	The raw path to the device that is described by EQ=.
BLOCKSIZE="*size*"	The block size in 512-byte units. Specify 256 for a block size of 128 kilobytes.
MEDIATYPE="*mt*"	The two-character media type for this tape as defined in the mcf(4) man page.
VSN_LIST="*vsn1 vsn2 ...*"	The list of VSNs to be read. There is no limit on the number of VSNs that can be specified. Use a space character to separate the VSNs. This list can be continued onto another line by using a backslash (\) character. For example: VSN_LIST="vsn1 vsn2 \ vsn3"

5. Execute the tarback.sh(1M) script.

Recovering Data From a Magneto-optical Volume

The procedures for recovering data from a magneto-optical volume differ, depending on the nature of the damage and whether or not additional archive copies of the volume's files are present on another tape. This section describes how to recover data in the following scenarios:

- Magneto-optical volume is damaged, and alternative archive copies are available.

  See "Damaged Magneto-optical Volume—Copies Available" on page 61.

- Magneto-optical volume is damaged, and no alternative archive copies are available.

See “Damaged Magneto-optical Volume—No Other Copies Available” on page 63.

- Magneto-optical volume is accidentally relabeled, and no alternative archive copies are available.

  See “Relabeled Magneto-optical Volume—No Other Copies Available” on page 65.

- Neither the ASM software nor the ASM-QFS software can read the magneto-optical volume label, and no alternative archive copies are available.

  See “Unreadable Label—No Other Copies Available” on page 65.

Damaged Magneto-optical Volume—Copies Available

Regardless of the nature of the damage to the magneto-optical volume, if an alternative archive copy is available, you should use the good magneto-optical volume as your primary set of archive copies.

The recovery procedure includes a step for rearchiving all archive copies currently stored on the damaged volume before dispensing with the damaged volume. The new archive copies are made from the available alternative archive copy.

To Rearchive Files and Recycle a Damaged Magneto-optical Volume—Copies Available

Use this procedure if readable alternative archive copies exist on volumes that are available on-site for staging.

1. Enter the `samexport(1M)` command to export the damaged volume from the magneto-optical library.

Use the syntax shown in the following screen example, specifying the media type (*mt*) and VSN (*vsn*) of the damaged volume.

```
# samexport mt.vsn
```

1. Enter the `chmed(1M)` command with the `-U` option to flag the damaged volume as unavailable in the historian catalog.

Use the syntax shown in the following screen example, specifying the media type (*mt*) and VSN (*vsn*) of the damaged volume.

```
# chmed +U mt.vsn
```

2. Enter the `chmed(1M)` command with the `-c` option to flag the unavailable volume for recycling.

   Use the syntax shown in the following screen example, specifying the media type (*mt*) and the VSN (*vsn*) of the damaged volume.

```
# chmed +c mt.vsn
```

3. Edit the `recycler.cmd(4)` file to set the `-ignore` option for the library.

   The following screen example shows the `-ignore` option set on the `lt20` library.

```
# vi /etc/opt/SUNWsamfs/recycler.cmd
logfile = /var/adm/recycler.log
lt20 -hwm 75 -mingain 60 -ignore
:wq
```

4. Enter the `sam-recycler(1M)` command with the `-x` option.

```
# sam-recycler -x
```

When the recycler runs, it does not select any volumes for recycling other than the volume you have marked as unavailable. The recycler identifies all active archive copies on this volume and flags those archive copies for rearchiving. The next time the archiver runs, the archive copies marked for rearchiving are written to new volumes.

After the archive copies have been written to new volumes, the damaged volume you are recycling is considered to be drained of active archive copies.

5. Dispense with the volume.

After the damaged volume is drained for active archive copies, you can dispense with the volume. How you dispense with it depends on the nature of the damage. See the following guidelines:

- If the magneto-optical volume was accidentally relabeled, use the `odlabel(1M)` command to relabel the volume.
- If the magneto-optical label is unreadable, export the volume from the historian and dispose of the magneto-optical volume.
- If the magneto-optical volume is partially corrupt, export the volume from the historian and dispose of the magneto-optical volume.
- If the magneto-optical volume is completely destroyed, export the volume from the historian and dispose of the magneto-optical volume.

If the magneto-optical platter is either partially corrupt or completely destroyed, it is possible (but not recommended) to reuse the magneto-optical label after the volume has been exported from the historian catalog.

If the magneto-optical volume is completely destroyed and no alternative archive copies exist, there is no chance for recovering any data from this magneto-optical platter.

Damaged Magneto-optical Volume—No Other Copies Available

If a magneto-optical volume is only partially corrupt, it is possible to recover data written to the parts of the magneto-optical volume that are not damaged. This process requires some trial and error to recover as much data as possible.

It is possible to determine the area of an magnetic optical platter that is damaged from errors logged in the device logs. By using file names for files that cannot be retrieved, you can determine the location of the damage using the position and offset data.

The `archive_audit(1M)` command audits all archive copies for a specific file system. The output of the `archive_audit` command includes the position and offset information for each archive copy. You can use this position and offset information to help determine which archive copies are written to an area of a damaged magneto-optical disk.

To Recover From a Damaged Magneto-optical Volume—No Other Copies Available

Copies of files that were archived outside the damaged area on a magneto-optical volume may be accessible. You can use the following procedure to recover files in accessible areas of a partially corrupted magneto-optical volume.

1. Use the `archive_audit(1M)` command to generate a list of all files with archive copies on the partially corrupt tape volume:

   Use the syntax shown in the following screen example, specifying the file system's mount point, the VSN of the damaged volume, and an output file name.

```
# archive_audit /mount_point | grep vsn > filename
```

2. Edit the `archive_audit` output file and create three separate files with the following contents:

- Files that appear before the damaged area on the magneto-optical disk

- Files that appear within the damaged area
- Files that appear after the damaged area.

3. Look for the files with archive copies within the damaged area of the magneto-optical disk to determine if any of the files are still in disk cache.

Files that are not in disk cache cannot be recovered.

4. Remove unrecoverable files from Step 2 from the file system.
5. Edit and run the `stageback.sh` script using the files created in Step 2 that list files outside the damaged area.

   The `stageback.sh` script stages each file from `archive_audit` output, sets it to `no-release`, and marks the file for rearchiving.

   See Table 1 for information about the `stageback.sh` script.

   a. Open the `/opt/SUNWsamfs/examples/stageback.sh` file for editing.

```
# cd /opt/SUNWsamfs/examples
# vi stageback.sh
```

   b. Find the section that begins with `# echo rearch $file.`

```
# echo rearch $file
#
# Edit the following line for the correct media type and VSN
#
# eval /opt/SUNWsamfs/bin/rearch -m media -v VSN $file
```

   c. In the section shown in the previous screen example, replace the word “media” with the media type and the word “VSN” with the same VSN specified in Step 1.

   d. Remove the pound sign from the beginning of the lines in the section shown in Step b.

```
 echo rearch $file
# Edit the following line for the correct media type and VSN
eval /opt/SUNWsamfs/bin/rearch -m media -v VSN $file
```

   e. Save and quit the file.

   f. Run the `stageback.sh` script.

Relabeled Magneto-optical Volume—No Other Copies Available

Unlike tape media, magneto-optical media does not have an EOD marker. When a magneto-optical volume is accidentally relabeled, the ASM and ASM-QFS software cannot access data written previously because of the label date. The ASM and ASM-QFS systems assume that if the label date on the magneto-optical volume is newer than the archive copy date of files, that data is no longer accessible.

Contact StorageTek customer support if a magneto-optical volume is accidentally relabeled. It is sometimes possible to recover some of this data with a special (but unsupported) `samst` driver that ignores the magneto-optical label date. This driver is not a standard part of the ASM or ASM-QFS products, and it is not released as part of the product. It can only be made available by StorageTek's customer support.

Unreadable Label—No Other Copies Available

For magneto-optical media, there is no standard approach for locating and skipping to the `various tar(1M)` files. Contact StorageTek customer support if you need to access files on a magneto-optical volume with an unreadable label.

Recovering File Systems

4

This chapter describes how to recover data when an ASM/QFS-Standalone, ASM, or ASM-QFS file system is corrupted or lost. These procedures differ, depending on the type of file system and whether or not you have a `samfsdump`(1M) or `qfsdump`(1M) of the file system available. You might require the assistance from your ASP or a StorageTek customer support staff member for this process to be successful.

This chapter covers the following topics

Recovering an ASM or ASM-QFS File System With a Metadata Dump File

If you have `samfsdump(1M)` metadata output for a file system, you can use the `samfsrestore(1M)` command to recover a file system that has been corrupted, accidentally remade, or destroyed. For details about the syntax and options used in the procedure, see the `samfsdump` and `samfsrestore man(1)` pages.

To Restore With a Metadata Dump File

This example restores a file system from a `samfsdump` dump file called `/dump_sam1/dump/041126`.

1. Use the `cd(1M)` command to change to the mount point for the file system or to the directory location where you want to restore the file system.

Caution: Consider restoring the file system first into a temporary directory and verifying that the restoration succeeds before restoring directly into the existing file system. This removes the risk of destroying the current file system before you can be sure the restoration is going to work. If the restoration fails, the file system may be recoverable by some other process.

In the following example, the mount point is `/sam1`.

```
# cd /sam1
```

2. Use the `samfsrestore` command with the `-T` and `-f` options to restore the entire file system relative to the current directory.

   Use the syntax shown in the following screen example, specifying the pathname of the dump file after the `-f` option, and the pathname of a log file after the `-g` option.

```
# samfsrestore -T -f /dump_sam1/dumps/041126 -g log
```

Note: The *log* file in the previous screen example can be used as input to `restore.sh(1M)` script to stage back files that were online at the time of the dump.

Recovering an ASM or ASM-QFS File System Without a Dump File

You may be able to recover data from an ASM or ASM-QFS file system even if you do not have access to output from a `samfsdump(1M)` command, or to an archiver log file.

The following procedure shows you how to recreate user files by reloading tape or optical disk and using the `star(1M)` command's `-n` option.

Note: Recovering file systems from archive cartridges and using the `star` command is a tedious and time-consuming process. This should not be considered the normal condition for disaster recovery.

To Recover Without a Dump File

1. (Optional) Disable any automated processes that are related to ASM or ASM-QFS operations.

If you have any of the following automated processes running, disable them during the recovery process to ensure that no data is lost:

- Recycling. Disable any recycling activities, including those triggered by an entry in root's `crontab(4)`. Failure to disable recycling activity could result in tapes being recycled and relabeled that contain active data.
- Archiving
- Processes that capture `samfsdump(1M)` files. Suspending these processes saves an existing `samfsdump` output file, and provides an opportunity for easier recovery.

- Writes into the file system

2. (Optional) Disable NFS-sharing for the file system.

It can be easier to recover data if the file system is not NFS-sharing the file systems during the recovery period.

3. Use the `sammkfs(1M)` command to remake the ASM or ASM-QFS file system to be restored.
4. Identify the cartridges that contain the archive copy information.
5. Read all the archive media.

   If you are using tapes, use `tar(1M)`, `gnutar(1M)`, or `star(1M)`.

6. If recovering from tape media, use the `tarback.sh` script.

The `tarback.sh(1M)` script is described in “Disaster Recovery Commands and Tools” on page 15. For more information about this script, see the `tarback.sh` man page. See also “Unreadable Tape Label—No Other Copies Available” on page 59 for an example of how to use the script.

The script is located in `/opt/SUNWsamfs/examples/tarback.sh`. This script identifies a single tape drive for use during recovery, and provides a list of VSNs to recover. The script uses `star(1M)` to loop through a volume, reading all available archive files.

The `star(1M)` command is an enhanced version of `gnutar(1M)`. The `tarback.sh` script uses `star(1M)` and the `-n` option, which is a `star(1M)` extension to `gnutar(1M)`. The `-n` option restores only files that are newer than existing copy. If the archive copy you are about to restore is older than the existing copy, the restore is skipped. This is important because it means that you do not have to worry about reading archive media in a specific order.

7. If recovering from magnetic-optical media, contact StorageTek support.

Recovering an ASM/QFS-Standalone File System

To recover an ASM/QFS-Standalone file system, you must have a `qfsdump(1M)` file available. The following procedure shows how to use a `qfsdump(1M)` file to recover an ASM/QFS-Standalone file system.

To Recover an ASM/QFS-Standalone File System Using a qfsdump File

This procedure assumes that the ASM/QFS-Standalone file system is not currently mounted at the `/qfs1` mount point used in the example.

1. If the disk slices you want to use for the file system are not already defined in the `mcf`(4) file, define them.

   Use `vi`(1) or another editor to make the desired changes to the `/etc/opt/SUNWsamfs/mcf file`.

2. Enter the `samd`(1M) command with the `config` subcommand.

```
# /opt/SUNWsamfs/sbin/samd config
```

3. Enter the `sammkfs(1M)` command with the `-a` option to make a new file system.

   Use the syntax shown in the following screen example, specifying a DAU after the `-a` option. The example uses a DAU of 128.

```
# /opt/SUNWsamfs/sbin/sammkfs -a 128 /qfs1
```

4. Enter the `mount(1M)` command to mount the file system.

```
# mount /qfs1
```

5. Enter the `cd(1M)` command to change to the mount point of the ASM/QFS-Standalone file system.

```
# cd /qfs1
```

6. Enter the `qfsrestore(1M)` command with the `-T` and `-f` options to restore the file system.

The `-T` option provides statistical information upon completion of the `qfsrestore(1M)` command's activities. Specify the pathname to the `qfsdump(1M)` output file after the `-f` option.

```
# qfsrestore -T -f /dump_qfs1/dumps/041111
```

Note: The ASM/QFS-Standalone file system (files and inode information) is fully restored by the `qfsrestore(1M)` command.

Recovering From Catastrophic Failure

5

Certain events can be classified as catastrophic failures. These include the damage caused by natural disasters, such as flooding in a computer room. This chapter provides a procedure to follow after such an event. You might require the assistance from your ASP or from StorageTek customer support to successfully complete the procedures described in this chapter.

To Recover From a Catastrophic Failure

Any system component, software element, ASM file system, or ASM-QFS file system that has not failed should not be recovered. However, you might need to reconfigure the ASM or ASM-QFS file system on a restored system to regain access to file systems or to determine whether any file system has failed. For details in performing these tasks, see the other chapters of this manual.

1. Determine the failed system component

   See “To Restore Failed System Components” on page 72.

2. Disable the archiver and the recycler until all files are restored.

   See “To Disable the Archiver and Recycler Until All Files are Restored” on page 72.

3. Compare previous and current configuration files, and reconcile inconsistencies.

   See “To Keep and Compare Previous and Current Configuration and Log Files” on page 75.

4. Repair disks.

   See “To Repair Disks” on page 75

5. Restore or build new library catalog files.

   See “To Restore or Build New Library Catalog Files” on page 75.

6. Make new file systems and restore from `samfsdump` output.

   See “To Make New File Systems and Restore from samfsdump Output” on page 75.

To Restore Failed System Components

■ Ascertain which components have failed.

The following steps describe how to restore the following types of components:

- Hardware
- Operating environment
- ASM or ASM-QFS packages.

7. If a hardware component has failed, restore it to operation, preserving any available data.

   If the failing component is a disk drive that has not totally failed, preserve any information possible. Before replacing or reformatting the disk, identify any salvageable files (including those in the following list), and copy these files to a tape or to another disk for future use in the recovery process.

   - ASM or ASM-QFS file system dumps
   - ASM or ASM-QFS configuration files, archiver log files, or library catalogs

8. If the ASM operating environment has failed, restore it to operation.

   See “Recovering from Failure of the Operating Environment Disk” on page 2. Verify that the ASM operating environment is functioning correctly before proceeding.

9. If the ASM or ASM-QFS packages have been damaged, remove and reinstall them from a backup copy or from its distribution file.

   You can verify whether a package has been damaged by using the `pkgchk`(1M) utility.

10. If disk hardware used by ASM or ASM-QFS was repaired or replaced in Step 7, configure the disks (RAID binding or mirroring) if necessary.

    Reformat disks only if they have been replaced or if it is otherwise absolutely necessary, because reformatting destroys all the file system information.

To Disable the Archiver and Recycler Until All Files are Restored

Caution: If the recycler is enabled so that it runs before all files are restored, cartridges with good archive copies may be improperly relabeled.

1. Add a single global `wait` directive to the `archiver.cmd` file or add a file-system-specific wait directive for each file system for which you want to disable archiving.

Note: The `wait` directive can be applied globally or individually to one or more file systems.

   a. Open the `/etc/opt/SUNWsamfs/archiver.cmd` file for editing and find the section where you want to insert the `wait` directive.

   The following screen example shows using the `vi`(1) command to edit the file. In the example, local archiving directives exist for two file systems `samfs1` and `samfs2`.

Figure 40.

```
# vi /etc/opt/SUNWsamfs/archiver.cmd
...
fs = samfs1
allfiles   .
1   10s
fs = samfs2
allfiles   .
1   10s
```

   b. Add the `wait` directive.

   The following screen example shows a global `wait` directive inserted before the first `fs = command (fs = samfs1)`.

Figure 41.

```
wait
fs = samfs1
allfiles   .
1   10s
fs = samfs2
allfiles   .
1   10s
:wq
```

   The following screen example shows two file system-specific `wait` directives inserted after the first and second `fs = commands (fs = samfs1` and `fs = samfs2)`.

Figure 42.

```
fs = samfs1
wait
allfiles   .
1   10s
fs = samfs2
wait
allfiles   .
1   10s
:wq
```

Add a global `ignore` directive to the `recycler.cmd` file or add a file-system-specific `ignore` directive for each library for which you want to disable recycling.

a. Open the `/etc/opt/SUNWsamfs/recycler.cmd` file for editing.

The following screen example shows using the `vi(1)` command to edit the file.

Figure 43.

```
# vi /etc/opt/SUNWsamfs/recycler.cmd
...
         logfile = /var/adm/recycler.log
         lt20 -hwm 75 -mingain 60
         lt20 75 60
         hp30 -hwm 90 -mingain 60 -mail root
         gr47 -hwm 95 -mingain 60 -mail root
```

b. Add the `ignore` directives.

The following screen example shows `ignore` directives added for three libraries.

Figure 44.

```
#  recycler.cmd.after  - example recycler.cmd file
#
         logfile = /var/adm/recycler.log
         lt20 -hwm 75 -mingain 60 -ignore
         hp30 -hwm 90 -mingain 60 -ignore -mail root
         gr47 -hwm 95 -mingain 60 -ignore -mail root
```

To Keep and Compare Previous and Current Configuration and Log Files

1. Recover any available ASM or ASM-QFS configuration files or archiver log files from the system's disks before rebuilding the system.
2. Compare the restored versions of all configuration files represented in the `SAMreport` with those restored from the system backups.
3. If inconsistencies exist, determine the effect of the inconsistencies and reinstall the ASM or ASM-QFS file system, if necessary, using the configuration information in the `SAMreport`.

   For more information on `SAMreport` file, see the `info.sh`(1M) man page.

To Repair Disks

- For ASM and ASM-QFS file systems that reside on disks that have not been replaced, run the `samfsck`(1M) utility to repair small inconsistencies, reclaim lost blocks, and so on.

  For command line options to the `samfsck` utility, see the `man(1)` page.

To Restore or Build New Library Catalog Files

1. Replace the most recent library catalog file copies from the removable media files, from the ASM or ASM-QFS server disks, or from the most recent file system archive copies (which are likely to be slightly out of date).
2. If the library catalogs are unavailable, build new catalogs by using the `build.cat(1M)` command, and using the library catalog section of the most recent `SAMreport` as input. Use the newest library catalog copy available for each automated library.

Note: ASM and ASM-QFS systems automatically rebuild library catalogs for SCSI-attached automated libraries. This does not occur for ACSLS-attached automated libraries. Tape usage statistics are lost.

To Make New File Systems and Restore from samfsdump Output

For those ASM and ASM-QFS file system that were resident (partially or totally) on disks that were replaced or reformatted, perform the following procedure.

1. Obtain the most recent copy of the `samfsdump(1M)` output file.
2. Make a new file system and restore the ASM or ASM-QFS file system using the `samfsdump` output file.

a. Use the `sammkfs(1M)` command to make a new file system.

```
# mkdir /sam1
# sammkfs samfs1
# mount samfs1
```

b. Use the `samfsrestore(1M)` command with the `-f` option and the `-g` option.

Specify the location of the `samfsdump` output file after the `-f` option. Specify the name of a log file after the `-g` option. The `-g` option creates a log of the files that had been online.

```
# cd /sam1
# samfsrestore -f /dump_sam1/dumps/040120 -g /var/adm/
messages/restore_log
```

Note: Once all file systems have been restored, the system can be made available to users in degraded mode.

3. On the file systems restored in Step 2, perform the following steps:

a. Run the `restore.sh(1M)` script against the log file created in Step b of Step 2, and stage all files that were known to be online prior to the outage.

b. Run the `sfind(1M)` command against the ASM or ASM-QFS file system to determine which files are labeled as damaged.

These files might or might not be restorable from tape, depending on the content of the archive log files. Determine the most recently available archive log files from one of the following sources:

- The removable media file.
- The ASM or ASM-QFS server disk.
- The most recent file system archive if not available from either of the previous two sources. This source is likely to be slightly outdated.

c. Run the `grep(1)` command against the most recent archive log file to search for the damaged files, to determine whether any of the damaged files were archived to tape since the last time the `samfsdump(1M)` command was run.

d. Examine the archive log files to identify any archived files that do not exist in the file system.

e. Use the `star(1M)` command to restore files from the archive media and to restore files that have been labeled as damaged.

These are files identified in Step c and Step d.

4. Reimplement disaster recovery scripts, methods, and `cron(1M)` jobs using information from the backup copies.

Glossary

A

addressable storage

The storage space encompassing online, nearline, offsite, and offline storage that is user-referenced through an ASM/QFS-Standalone, ASM, or ASM-QFS file system.

archive media

The media to which an archive file is written. Archive media can be removable tape or magneto-optical cartridges in a library. In addition, archive media can be a mount point on another system.

archive storage

Copies of file data that have been created on archive media.

archiver

The archive program that automatically controls the copying of files to removable cartridges.

ASM

The StorageTek File System known are the Application Storage Manager™ (ASM). The ASM software controls the access to all files stored and all devices configured in the master configuration file (`mcf`).

ASM-QFS

The ASM-QFS software combines the StorageTek Storage and Archive Manager with the ASM/QFS-Standalone file system. ASM-QFS offers a high-speed, standard UNIX file system interface to users and administrators in conjunction with the storage and archive management utilities. It uses many of the commands available in the ASM command set as well as standard UNIX file system commands.

ASM-Remote client

an ASM-Remote client is an ASM or ASM-QFS system that establishes an ASM-Remote client daemon that contains a number of pseudodevices. It might or might not have its own library devices. The client depends on an ASM-Remote server for archive media for one or more archive copies.

ASM-Remote server

The ASM-Remote server is both a full-capacity ASM or ASM-QFS storage management server and an ASM-Remote server daemon that defines libraries to be shared among ASM-Remote clients.

audit (full)

The process of loading cartridges to verify their VSNs. For magneto-optical cartridges, the capacity and space information is determined and entered into the automated library's catalog.

automated library

A robotically controlled device designed to automatically load and unload removable media cartridges without operator intervention. An automated library contains one or more drives and a transport mechanism that moves cartridges to and from the storage slots and the drives.

B

backup storage

A snapshot of a collection of files for the purpose of preventing inadvertent loss. A backup includes both the file's attributes and associated data.

block allocation map

A bitmap representing each available block of storage on a disk and indicating whether the block is in use or free.

block size

See DAU.

C

cartridge

A physical entity that contains media for recording data. A tape or optical disk. Sometimes referred to as *a piece of media*, *a volume*, or *the medium*.

catalog

A record of the VSNs in an automated library. There is one catalog for each automated library, and at a site, there is one historian for all automated libraries.

client-server

The model of interaction in a distributed system in which a program at one site sends a request to a program at another site and awaits a response. The requesting program is called the client. The program satisfying the response is called the server.

connection

The path between two protocol modules that provides reliable stream delivery service. A TCP connection extends from a TCP module on one machine to a TCP module on the other.

D

data device

For an ASM/QFS-Standalone, ASM, or ASM-QFS file system, a device or group of devices upon which file data is stored.

DAU (disk allocation unit)

The basic unit of online storage. Also called block size.

The ASM and ASM-QFS file systems support both a small and a large DAU. The small DAU is 4 kilobytes (2^{14} or 4096 bytes). The large DAU is 16, 32, or 64 kilobytes. The available DAU size pairs are 4/16, 4/32, and 4/64.

In addition, the ASM/QFS-Standalone and ASM-QFS file systems support a fully adjustable DAU, sized from 16 kilobytes through 65,528 kilobytes. The DAU you specify must be a multiple of 8 kilobytes.

device logging

A configurable feature that provides device-specific error information used to analyze device problems.

device scanner

Software within the ASM or ASM-QFS file system that periodically monitors the presence of all manually mounted removable devices and that detects the presence of mounted cartridges that can be requested by a user or other process.

direct access

A file attribute (stage never) designating that a nearline file can be accessed directly from the archive media and need not be retrieved to disk cache.

direct-attached library

An automated library connected directly to a server using a SCSI interface. A SCSI attached library is controlled directly by the

ASM or ASM-QFS software by using the SCSI standard for automated libraries.

direct I/O

An attribute used for large block-aligned sequential I/O. The `setfa`(1) command's `-D` option is the direct I/O option. It sets the direct I/O attribute for a file or directory. If applied to a directory, the direct I/O attribute is inherited.

directory

A file data structure that points to other files and directories within the file system.

disk allocation unit

See DAU.

disk buffer

When using ASM-Remote software, the disk buffer is a buffer on the server system that is used when archiving data from the client to the server.

disk cache

The disk-resident portion of the ASM and ASM-QFS file system software. It is used to create and manage data files between online disk cache and archive media. Individual disk partitions or an entire disk can be used as disk cache.

disk space thresholds

An administrator-defined amount of disk space that is available to a user. This defines the range of desirable disk cache utilization. The high threshold indicates the maximum level of disk cache utilization. The low threshold indicates the minimum level of disk cache utilization. The releaser controls disk cache utilization based on these predefined disk space thresholds.

disk striping

The process of recording a file across several disks, thereby improving access performance and increasing overall storage capacity. Also see entries for striping.

drive

A mechanism for transferring data to and from a removable media volume.

E

Ethernet

A local-area, packet-switched network technology. Originally designed for coaxial cable, it is now found running over shielded, twisted-pair cable. Ethernet is a 10- or 100-megabytes-per-second LAN.

extent array

The array within a file's inode that defines where each data block assigned to the file is located on the disk.

F

family device set

See family set.

family set

A storage device that is represented by a group of independent physical devices, such as a collection of disks or the drives within an automated library. Also see disk cache family set.

FDDI

Fiber distributed data interface. A 100-megabytes-per-second fiber-optic LAN.

fibre channel

The ANSI standard that specifies high-speed serial communication between devices. Fibre channel is used as one of the bus architectures in SCSI-3.

fibre-distributed data interface

See FDDI.

file system

A hierarchical collection of files and directories.

file system specific directives

Archiver and releaser directives that follow global directives, are specific to a particular file system, and begin with `fs =`. File system specific directives apply until the next `fs =` directive line or until the end of file is encountered. If multiple directives affect a file system, the file system-specific directives override the global directives.

FTP

File Transfer Protocol. An internet protocol for transferring files between two hosts over a TCP/IP network.

G

global directives

Archiver and releaser directives that apply to all file systems and that appear before the first `fs =` line.

grace period

For disk quotas, this is the amount of time that can elapse during which a user is allowed to create files and/or allocate storage after a user reaches their soft limit.

H

hard limit

For disk quotas, a maximum limit on file system resources (blocks and inodes) that users cannot exceed.

I

indirect block

A disk block that contains a list of storage blocks. The ASM/QFS-Standalone, ASM, and ASM-QFS file systems have up to three levels of indirect blocks. A first-level indirect block contains a list of blocks used for data storage. A second-level indirect block contains a list of first-level indirect blocks. A third-level indirect block contains a list of second-level indirect blocks.

inode

Index node. A data structure used by the file system to describe a file. An inode describes all the attributes associated with a file other than the name. The attributes include ownership, access, permission, size, and the file location on the disk system.

inode file

A special file (`.inodes`) on the file system that contains the inode structures for all files resident in the file system. All ASM/QFS-Standalone, ASM, and ASM-QFS inodes are 512 bytes long. The inode file is a metadata file, which is separated from file data in the ASM/QFS-Standalone and ASM-QFS file systems.

K

kernel

The central controlling program that provides basic system facilities. The UNIX kernel creates and manages processes, provides functions to access the file system, provides general security, and supplies communication facilities.

L

LAN

Local area network.

lease

In an ASM/QFS-Standalone shared file system, a lease grants a client host permission to perform an operation on a file for as long as the lease is valid. The

metadata server issues leases to each client host. The leases are renewed as necessary to permit continued file operations.

library

See automated library.

library catalog

See catalog.

LUN

Logical unit number.

M

mcf

Master configuration file. The file that is read at initialization time that defines the relationships between the devices (the topology) within an ASM/QFS-Standalone, ASM, and ASM-QFS environment.

media

Tape or optical disk cartridges.

media recycling

The process of recycling or reusing archive media with low use (that is, archive media with few active files).

metadata

Data about data. Metadata is the index information needed to locate the exact data position of a file on a disk. It consists of information about files, directories, access control lists, symbolic links, removable media, segmented files, and the indexes of segmented files. Metadata must be protected because if data is lost, the metadata that locates the data must be restored before the lost data can be retrieved.

metadata device

A separate device (for example, a solid-state disk or mirrored device) upon which ASM/QFS-Standalone and ASM-QFS file system metadata is stored. Separating file data from metadata can increase performance. In the `mcf` file, a metadata device is declared as an `mm` device within an `ma` file system.

mirror writing

The process of maintaining two copies of a file on disjointed sets of disks to prevent loss from a single disk failure.

mount point

The directory on which a file system is mounted.

N

name space

The metadata portion of a collection of files that identifies the file, its attributes, and its storage locations.

nearline storage

Removable media storage that requires robotic mounting before it can be accessed. Nearline storage is usually less expensive than online storage, but it incurs a somewhat longer access time.

network-attached automated library

A library, such as those from StorageTek, ADIC/Grau, IBM, or Sony, that is controlled using a software package supplied by the vendor. The ASM and ASM-QFS file systems interface with the vendor software using an ASM or ASM-QFS media changer daemon designed specifically for the automated library.

NFS

Network file system. A StorageTek distributed file system that provides transparent access to remote file systems on heterogeneous networks.

NIS

The ASM OS 4.0 (minimum) Network Information Service. A distributed network database containing key information about the systems and the users on the network. The NIS database is stored on the master server and all the slave servers.

O

offline storage

Storage that requires operator intervention for loading.

offsite storage

Storage that is remote from the server and is used for disaster recovery.

online storage

Storage that is immediately available (for example, disk cache storage).

P

partition

A portion of a device or a side of a magneto-optical cartridge.

preallocation

The process of reserving a contiguous amount of space on the disk cache for writing a file. This ensures that the space is contiguous. Preallocation can be performed only on zero-sized files. That is, the `setfa -l` command can be specified only for a file that is size zero. For more information, see the `setfa`(1) man page.

prioritizing preview requests

Assigning priority to archive and stage requests that cannot be immediately satisfied.

pseudo device

A software subsystem or driver with no associated hardware.

Q

quota

The amount of system resources that a user is allowed to consume. Quotas are not supported for removable media or disk archive resources.

R

RAID

Redundant array of inexpensive/ independent disks. A disk technology that uses several independent disks to reliably store files. It can protect against data loss from a single disk failure, can provide a fault-tolerant disk environment, and can provide higher throughput than individual disks.

recycler

an ASM and ASM-QFS utility that reclaims space on cartridges that is occupied by expired archive copies.

release priority

A method of calculating the release priority of a file within a file system by multiplying various weights by the corresponding file properties and then summing the results.

releaser

an ASM and ASM-QFS component that identifies archived files and releases their disk cache copies, thus making more disk cache space available. The releaser automatically regulates the amount of online disk storage to high and low thresholds.

remote procedure calls

See RPC.

removable media file

A special type of user file that can be accessed directly from where it resides on a removable media cartridge, such as

magnetic tape or optical disk cartridge. also used for writing archive and stage file data.

robot

The portion of an automated library that moves cartridges between storage slots and drives. Also called a transport.

round robin

A data access method in which entire files are written to logical disks in a sequential fashion. When a single file is written to disk, the entire file is written to the first logical disk. The second file is written to the next logical disk, and so on. The size of each file determines the size of the I/O.

By default, ASM/QFS-Standalone, ASM, and ASM-QFS file systems implement striped data access unless striped groups are present. Files are round-robined if round robin access is specified. If the file system contains mismatched striped groups, striping is not supported and round robin is forced.

Also see glossary entries for disk striping and striping.

RPC

Remote procedure calls. The underlying data exchange mechanism used by NFS to implement custom network data servers.

S

samfsdump

A program that creates a control structure dump and copies all the control structure information for a given group of files. It is analogous to the UNIX `tar`(1) utility, but it does not generally copy file data.

samfsrestor e

A program that restores inode and directory information from a control structure dump.

SCSI

Small Computer System Interface. An electrical communication specification commonly used for peripheral devices such as disk and tape drives and automated libraries.

shared writer/shared reader

The ASM/QFS-Standalone shared reader/ shared writer capability enables you to specify a file system that can be shared by multiple servers. Multiple hosts can read the file system, but only one host can write to the file system. Shared readers are specified with the `-o shared_reader` option on the `mount`(1M) command. The one-writer host is specified with the `-o shared_writer` option on the `mount`(1M) command. For more information on the `mount`(1M) command, see the `mount_samfs`(1M) man page.

small computer system interface

See SCSI.

soft limit

For disk quotas, a threshold limit on file system resources (blocks and inodes) that you can temporarily exceed. Exceeding the soft limit starts a timer. When you exceed the soft limit for the specified time (default is one week), no further system resources can be allocated until you reduce file system use to a level below the soft limit.

staging

The process of copying a nearline or offline file from archive storage back to online storage.

storage family set

A set of disks that are collectively represented by a single disk family device.

storage slots

Locations inside an automated library in which cartridges are stored when not being used in a drive. If the library is direct-

attached, the contents of the storage slots are kept in the automated library's catalog.

stripe size

The number of disk allocation units (DAUs) to allocate before moving to the next device of a stripe. If `stripe=0`, the file system uses round-robin access, not striped access.

striped group

A collection of devices within an ASM/QFS-Standalone or ASM-QFS file system and defined in the `mcf` file as one (usually two) or more `g`*`XXX`* devices. Striped groups are treated as one logical device and are always striped with a size equal to the disk allocation unit (DAU). You can specify up to 128 striped groups within a file system, but you can specify no more than 252 total devices.

striping

A data access method in which files are simultaneously written to logical disks in an interlaced fashion. All ASM/QFS-Standalone, ASM, and ASM-QFS file systems enable you to declare either striped or round robin access for each individual file system. The ASM/QFS-Standalone and ASM-QFS file systems enable you to declare striped groups within each file system. Also see the glossary entry for round robin.

superblock

A data structure in the file system that defines the basic parameters of the file system. It is written to all partitions in the storage family set and identifies the partition's membership in the set.

T

tar

Tape archive. A standard file/data recording format used by the ASM and ASM-QFS software for archive images.

TCP/IP

Transmission Control Protocol/Internet Protocol. The internet protocols responsible for host-to-host addressing and routing, packet delivery (IP), and reliable delivery of data between application points (TCP).

thresholds

A mechanism for defining the desirable available storage window for online storage. Thresholds set the storage goals for the releaser. Also see disk space thresholds.

timer

Quota software that keeps track of the time elapsed between a user reaching a soft limit and a hard limit being imposed on the user.

V

volume

A named area on a cartridge for sharing data. A cartridge has one or more volumes. Double-sided cartridges have two volumes, one on each side.

volume overflow

A capability that enables the system to span a single file over multiple volumes. Volume overflow is useful for sites using very large files that exceed the capacity of their individual cartridges.

VSN

Volume serial name. If you are archiving to removable media cartridges, the VSN is a logical identifier for magnetic tape and optical disk that is written in the volume label. If you are archiving to disk cache, this is the unique name for the disk archive set.

W

WORM

Write once read many. A storage classification for media that can be written only once but read many times.

Index

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

U

V

W

Reader’s Comment Form

Contact Us

Submit your questions, comments, and suggestions to StorageTek’s Global Learning Solutions. We appreciate your correspondence and are committed to responding to you.

Publication Information

Publication Name:

Publication Part Number:

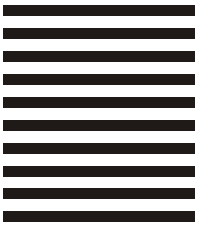
Questions and Comments:

Note: Staples can cause problems with automated mail sorting equipment. Please use pressure sensitive or other gummed tape to seal this form. If you would like a reply, please supply your name and address on the reverse side of this form.

Thank you for your cooperation. No postage stamp is required if mailed in the U.S.A.

TO COMPLY WITH POSTAL REGULMATIONS, FOLD EXACTLY ON DOTTED LINES AND TAPE (DO NOT

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES



BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 2 LOUISVILLE, CO U.S.A.

POSTAGE WILL BE PAID BY ADDRESSEE

GLOBAL LEARNING SOLUTIONS MS 3256
STORAGE TECHNOLOGY CORPORATION
ONE STORAGETEK DRIVE
LOUISVILLE CO 80028-9989
USA

FOLD HERE AND TAPE DO NOT STAPLE FOLD HERE AND TAPE

If you would like a reply, please print:

Your Name:_____

Company Name:_____ Department:_____

Street Address:_____

City:_____

State:_____ Zip Code:_____

Storage Technology Corporation
One StorageTek Drive
Louisville, CO 80028-3256
USA

NEED MORE INFORMATION?

www.storagetek.com

ABOUT STORAGETEK®

StorageTek® (NYSE:STK), a $2 billion world-wide company with headquarters in Louisville, Colo., delivers a broad range of storage solutions for digitized data. StorageTek solutions are easy to manage and allow universal access to data across servers, media types and storage networks. StorageTek is the innovator and global leader in virtual storage solutions for tape automation, disk storage systems and storage networking. Because of StorageTek, customers can manage and leverage their digital assets as their businesses grow, and can maximize IT productivity to ensure enterprise-class business continuity.

WORLD HEADQUARTERS
Storage Technology Corporation
One StorageTek Drive
Louisville, Colorado 80028 USA
Phone: **1.800.525.0369**

StorageTek®