

# **Sun Storage Archive Manager 5.3 Configuration and Administration Guide**

Copyright © 2011, 2012, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

---

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée d'The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation.

# Contents

---

<b>Preface</b> .....	13
<b>1 About SAM-QFS</b> .....	15
Components of SAM-QFS .....	15
Archiving .....	15
Releasing .....	16
Staging .....	16
Recycling .....	16
Supported Storage Devices .....	17
SAM-Remote Software .....	18
<b>2 Configuring Storage Devices for Archiving</b> .....	19
Task Map: Configuring Storage Devices for Archiving .....	19
Adding Tape Devices for Archiving .....	20
Files that Contain Configuration Information .....	20
▼ How to Add Tape Devices for Archiving (Command Line) .....	21
▼ How to Add Tape Drive Interface Target IDs and LUNs for Archiving (Command Line) .....	23
▼ How to Add Tape Devices for Archiving (SAM-QFS Manager) .....	24
Adding Libraries or Magneto-Optical Drives for Archiving .....	24
▼ How to Configure Device Support in SCSI or FC Environments Using SAM-QFS Manager .....	25
▼ How to Configure Device Support for a Direct-Attached Library .....	25
Recovering After a Direct-Attached Library Fails to Initialize .....	26
Verifying and Implementing Configurations .....	26
▼ How to Verify That All Devices Are Configured and Implement the Changes .....	27
Handling Errors in the st.conf File .....	28

<b>3</b>	<b>Performing Additional SAM-QFS Configuration</b> .....	29
	Sharing the File System With NFS Client Systems .....	29
	▼ How to NFS Share the File System .....	30
	▼ How to Mount the File System on Clients .....	30
	Editing the defaults.conf File .....	31
	Features You Can Control From defaults.conf .....	32
	▼ How to Customize Default Values .....	33
	Configuring the Remote Notification Facility .....	34
	▼ How to Enable Remote Notification .....	34
	▼ How to Disable Remote Notification .....	35
	Adding the Administrator Group .....	35
	▼ How to Add the Administrator Group .....	36
	▼ How to Enable System Logging .....	36
	Configuring Other Sun Storage Products .....	37
<b>4</b>	<b>Creating Parameters Files for Network-Attached Automated Libraries</b> .....	39
	Creating Parameters Files for Network-Attached Automated Libraries .....	39
	▼ How to Create a Parameters File for Network-Attached Automated Libraries .....	40
	Configuring a StorageTek ACSLS-Attached Automated Library Parameters File .....	40
	Configuring an ADIC/Grau Automated Library Parameters File .....	42
	Configuring an IBM 3494 Automated Library Parameters File .....	44
	Configuring a Sony Network-Attached Automated Library Parameters File .....	46
	About Shared Drives .....	48
<b>5</b>	<b>Checking the Drive Order in Libraries</b> .....	49
	Checking the Drive Order in Libraries .....	49
	▼ How to Check the Drive Order of Libraries With a Front Panel .....	49
	▼ How to Check the Drive Order of a Tape Library Without a Front Panel .....	50
	▼ How to Check the Drive Order of Magneto-Optical Libraries Without a Front Panel .....	51
	▼ How to Check the Drive Order of Network-Attached Libraries .....	52
<b>6</b>	<b>Populating the Catalog</b> .....	55
	Populating the Catalog .....	55
	▼ How to Populate an Automated Library With Many Volumes .....	55

▼ How to Populate an Automated Library With a Small Number of Volumes .....	56
▼ How to Populate an IBM 3494 Automated Library .....	57
▼ How to Populate a StorageTek ACSLS-Attached Library Quickly .....	57
StorageTek ACSLS-Attached Automated Libraries: Common Problems and Error Messages .....	58
StorageTek ACSLS Parameters File Errors .....	58
StorageTek ACSLS Library Errors .....	58
VSN Import Errors .....	59
<b>7 Managing Automated Libraries and Manually Loaded Drives .....</b>	<b>61</b>
Vendor-Specific Library Operational Procedures .....	61
▼ How to Start Removable Media Operations .....	61
▼ How to Stop Removable Media Operations .....	62
▼ How to Turn On an Automated Library .....	62
▼ How to Turn Off an Automated Library .....	63
▼ How to Load a Cartridge Manually .....	63
▼ How to Unload a Cartridge Manually .....	64
Labeling and Unlabeling Cartridges .....	64
▼ How to Label or Relabel a Tape .....	65
▼ How to Label or Relabel an Optical Disc .....	65
▼ How to Audit a Volume .....	66
▼ How to Audit a Direct Attached Automated Library .....	67
Using a Cleaning Cartridge .....	67
▼ How to Use a Cleaning Cartridge With a Barcode .....	67
▼ How to Use a Cleaning Cartridge Without a Barcode .....	68
▼ How to Reset the Number of Cleaning Cycles .....	68
▼ How to Limit the Number of Cleaning Cycles .....	69
▼ How to Clean a Tape Drive Automatically .....	69
▼ How to Clean a Tape Drive Manually .....	70
▼ How to Clear Media Errors .....	70
▼ How to Remove a Stuck Cartridge From a Drive .....	71
Catalog Operations, Importing Cartridges, and Exporting Cartridges .....	72
Tracking Exported Media - The Historian .....	73
About Importing and Exporting From an Automated Library .....	74
▼ How to Import a Cartridge From a Library With a Mailbox .....	75

▼ How to Export a Cartridge From a Library With a Mailbox .....	75
▼ How to Import a Cartridge From a Library Without a Mailbox .....	76
▼ How to Export a Cartridge From a Library Without a Mailbox .....	76
▼ How to Enable Load Notification .....	77
Using Drives With Encryption Capability .....	77
Manually Loaded Drive Operations .....	77
▼ How to Load a Cartridge Into a Manually Loaded Device .....	78
▼ How to Unload a Cartridge .....	78
▼ How To View a Library Catalog .....	78
<b>8 Managing Vendor-Specific Libraries .....</b>	<b>79</b>
StorageTek ACSLS-Attached Automated Libraries .....	79
▼ How to Import Tapes .....	80
▼ How to Export Tapes Using a Mailbox .....	80
ADIC/Grau Automated Libraries .....	81
▼ How to Import a Cartridge .....	81
▼ How to Export a Cartridge .....	82
IBM 3584 UltraScalable Tape Libraries .....	82
Importing Cartridges .....	82
Cleaning Drives .....	83
Partitioning .....	83
▼ How To Remove a Cartridge From a Logical Library .....	83
IBM 3494 Libraries .....	84
▼ How to Import a Cartridge .....	84
▼ How to Export a Cartridge .....	84
Sony Direct-Attached 8400 PetaSite Automated Libraries .....	85
▼ How to Import Tapes .....	85
▼ How to Export a Tape Without Using Mailbox Slots as Storage Slots .....	86
▼ How to Export a Tape Using Mailbox Slots as Storage Slots .....	86
▼ How to Move a Cartridge to a Different Slot .....	87
Sony Network-Attached Automated Libraries .....	87
▼ How to Import a Cartridge .....	88
▼ How to Export a Cartridge .....	88

<b>9</b>	<b>About Archiving</b> .....	89
	Planning Archiving Operations .....	89
	Preview Queue .....	90
	Archiver Daemons .....	90
	Archiving Operations .....	91
	Step 1: Identifying Files to Archive .....	92
	Step 2: Composing Archive Requests .....	94
	Step 3: Scheduling Archive Requests .....	95
	Step 4: Archiving the Files in an Archive Request .....	97
	Sample Default Output From the archiver -l Command .....	98
	Log Files and Event Logging for Archive Operations .....	98
	Data Verification .....	100
<b>10</b>	<b>Configuring the Archiver</b> .....	101
	About the archiver .cmd File .....	101
	Archive Directives .....	102
	Archive Set Directives .....	102
	▼ How to Create an archiver .cmd File Using the Command Line .....	103
	▼ How to Create an archiver .cmd File Using SAM-QFS Manager .....	103
	Examples of archiver .cmd Files .....	104
	About Disk Archiving .....	114
	About the diskvols .conf File .....	115
	Disk Archiving Directives .....	117
	▼ How to Enable Disk Archiving on the Client Host .....	119
	▼ How to Configure Disk Archiving on the Server Host .....	120
	▼ How to Enable Disk Archiving .....	121
	Disk Archiving Examples .....	122
<b>11</b>	<b>Archive Directives (archiver .cmd)</b> .....	125
	Global Archiving Directives .....	125
	archivemeta Directive: Controlling Whether Metadata Is Archived .....	126
	archmax Directive: Controlling the Size of Archive Files .....	126
	bufsize Directive: Setting the Archiver Buffer Size .....	127
	drives Directive: Controlling the Number of Drives Used for Archiving .....	128
	examine Directive: Controlling Archive Scans .....	128

interval Directive: Specifying an Archive Interval .....	129
logfile Directive: Specifying an Archiver Log File .....	129
notify Directive: Renaming the Event Notification Script .....	130
ovflmin Directive: Controlling Volume Overflow .....	130
scanlist_squash Directive: Controlling Scanlist Consolidation .....	132
setarchdone Directive: Controlling the Setting of the archdone Flag .....	132
wait Directive: Delaying Archiver Startup .....	132
File System Directives .....	133
fs Directive: Specifying the File System .....	133
Global Directives as File System Directives .....	133
Archive Copy Directives .....	134
-release Directive: Releasing Disk Space After Archiving .....	134
-norelease Directive: Delaying Disk Space Release .....	134
Using -release and -norelease Together .....	135
Setting the Archive Age .....	135
Unarchiving Automatically .....	135
Specifying More Than One Copy for Metadata .....	136
<b>12 Archive Set Directives (archiver.cmd) .....</b>	<b>137</b>
About Archive Set Directives .....	137
Archive Set Assignment Directive .....	138
File Age <i>search-criterion</i> Arguments: -access and -nftv .....	139
File Age <i>search-criterion</i> Arguments: -after .....	140
File Size <i>search-criterion</i> Arguments: -minsize and -maxsize .....	140
Owner and Group <i>search-criterion</i> Arguments: -user and -group .....	141
File Name <i>search-criterion</i> Arguments Using Pattern Matching: -name <i>regex</i> .....	141
Release and Stage <i>file-attributes</i> Argument: -release and -stage .....	143
Membership Conflicts in Archive Sets .....	143
Archive Set Copy Parameters .....	144
Controlling the Size of Archive Files: -archmax Parameter .....	145
Setting the Archiver Buffer Size: -bufsize Parameter .....	146
Specifying the Number of Drives for an Archive Request: -drivemax, -drivemin, and -drives .....	146
Maximizing Space on a Volume: -fillvsns Parameter .....	148
Specifying Archive Buffer Locks: -lock Parameter .....	148

Making Archive Copies of Offline Files: -offline_copy Parameter .....	149
Specifying Recycling .....	149
Sorting Archive Files: -sort and -rsort Parameters .....	149
Controlling Unarchiving .....	150
Controlling How Archive Files Are Written: -tapenonstop Parameter .....	151
Reserving Volumes: -reserve Parameter .....	151
Setting Archive Priorities: -priority Parameter .....	153
Scheduling Archiving: -startage, -startcount, and -startsize Parameters .....	154
VSN Association Directives .....	156
VSN Pools Directives .....	157
<b>13 Data Integrity Validation in SAM-QFS .....</b>	<b>159</b>
About DIV .....	159
Minimum Hardware and Software Requirements for Using DIV in SAM-QFS .....	160
Configuring DIV in SAM-QFS .....	160
▼ How to Configure DIV in SAM-QFS .....	160
Examples of Using DIV .....	161
Tape Verify (tpverify) Command .....	162
Tape Verify (tpverify) Command Return Values .....	162
<b>14 About Releasing .....</b>	<b>165</b>
Releaser Process .....	165
Releaser Concepts .....	166
Age .....	166
Candidate .....	166
Priority .....	167
Weight .....	167
Partial Release .....	167
About Partial Releasing and Partial Staging .....	168
Summary of System Administrator Options .....	169
Summary of User Options .....	170
<b>15 Configuring the Stager .....</b>	<b>173</b>
About Staging .....	173

---

About the <code>stager.cmd</code> File .....	174
▼ How to Create a <code>stager.cmd</code> File .....	174
drives Directive: Specifying the Number of Drives for Staging .....	175
bufsize Directive: Setting the Stage Buffer Size .....	175
logfile Directive: Specifying a Log File .....	176
maxactive Directive: Specifying the Number of Stage Requests .....	178
Archive Set Assignment Directive: Specifying Stage Attributes for All Files in an Archive Set	178
copysel Directive: Specifying the Copy Sequence for Staging .....	179
Prioritizing Preview Requests .....	179
About the <code>preview.cmd</code> File .....	180
▼ How to Set the Global VSN and Age Priority Directives .....	181
▼ How to Set Global or File System-Specific Water Mark Directives .....	181
Setting Up a Preview Request Priority Scheme .....	183
<b>16 Configuring the Recycler .....</b>	<b>187</b>
About Recycling .....	187
Recycle Process .....	188
Planning for Recycling .....	188
Recycling Methods .....	189
Controlling Recycling .....	190
Starting the Recycler .....	190
Configuring Recycling on Removable Media Cartridges .....	191
Creating a <code>recycler.cmd</code> File .....	191
logfile Directive: Specifying a Log File .....	192
no_recycle Directive: Preventing Recycling .....	193
library Directive: Specifying Recycling for an Automated Library .....	193
Creating a <code>recycler.sh</code> File .....	195
Configuring Recycling for Disk Archive Volumes .....	196
Editing the <code>archiver.cmd</code> File .....	196
Recycler Logging for Disk Archives .....	197
Recycling for Archive Copy Retention .....	197
<b>17 Advanced SAM-QFS Topics .....</b>	<b>199</b>
Using Device Logging .....	199
When to Use the Device Log .....	200

▼ How to Enable the Device Log by Using the <code>samset</code> Command .....	200
▼ How to Enable the Device Log by Editing the <code>defaults.conf</code> File .....	200
Using Removable Media Files .....	201
Creating a Removable Media or Volume Overflow File .....	201
Using Segmented Files .....	202
Archiving Segmented File .....	203
Using System Error Facility Reporting .....	203
▼ How to Enable SEF Reporting .....	204
▼ How to Generate SEF Report Output .....	205
Managing the SEF Log File .....	207
SEF <code>sysevent</code> Functionality .....	207
▼ How to Create the SEF <code>sysevent</code> Handler .....	207
<b>18 Using the Sun SAM-Remote Software .....</b>	<b>209</b>
Sun SAM-Remote Software Overview .....	209
System Requirements .....	210
Software Limitations .....	211
Client and Server Interactions .....	211
Configuring the Sun SAM-Remote Software .....	214
▼ Step 1: Log In to the Potential Server and Client Hosts .....	216
▼ Step 2: Verify Client and Server Software .....	216
▼ Step 3: Edit the <code>mcf</code> Files on Each Client .....	217
▼ Step 4: Create a Sun SAM-Remote Client Configuration File .....	219
▼ Step 5: Edit the Server's <code>mcf</code> File .....	219
▼ Step 6: Create the Server's Configuration File .....	220
▼ Step 7: Enable Archiving .....	222
Recycling With the Sun SAM-Remote Software .....	224
Recycling in a Sun SAM-Remote Environment (Method 1) .....	225
▼ How to Configure Recycling (Method 1) .....	227
Examples for How to Configure Recycling (Method 1) .....	231
▼ How to Recycle <code>no-data</code> VSNS .....	240
▼ How to Recycle <code>partially full</code> VSNS .....	241
Recycling in a Sun SAM-Remote Environment (Method 2) .....	242
▼ How To Configure Recycling (Method 2) .....	243



# Preface

---

The *Sun Storage Archive Manager Configuration and Administration Guide* provides information about managing the Sun Storage Archive Manager software.

## Who Should Use This Book

This guide is intended for system administrators who are interested in managing the Sun Storage Archive Manager software.

## Before You Read This Book

Read *Sun QFS and Sun Storage Archive Manager 5.3 Installation Guide* before you begin to manage the SAM-QFS software.

## Related Books

- *Sun QFS and Sun Storage Archive Manager 5.3 Installation Guide*
- *Sun QFS File System 5.3 Configuration and Administration Guide*
- *Sun QFS and Sun Storage Archive Manager 5.3 Release Notes*
- *Sun QFS and Sun Storage Archive Manager Reference Manual*

## Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

## Typographic Conventions

The following table describes the typographic conventions that are used in this book.

TABLE P-1 Typographic Conventions

Typeface	Description	Example
AaBbCc123	The names of commands, files, and directories, and onscreen computer output	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. <code>machine_name% you have mail.</code>
<b>AaBbCc123</b>	What you type, contrasted with onscreen computer output	<code>machine_name% su</code> <code>Password:</code>
<i>aabbcc123</i>	Placeholder: replace with a real name or value	The command to remove a file is <code>rm filename</code> .
<i>AaBbCc123</i>	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the <i>User's Guide</i> . <i>A cache</i> is a copy that is stored locally. Do <i>not</i> save the file. <b>Note:</b> Some emphasized items appear bold online.

## Shell Prompts in Command Examples

The following table shows the default UNIX system prompt and superuser prompt for shells that are included in the Oracle Solaris OS. Note that the default system prompt that is displayed in command examples varies, depending on the Oracle Solaris release.

TABLE P-2 Shell Prompts

Shell	Prompt
Bash shell, Korn shell, and Bourne shell	\$
Bash shell, Korn shell, and Bourne shell for superuser	#
C shell	machine_name%
C shell for superuser	machine_name#

## About SAM-QFS

---

The Sun Storage Archive Manager (SAM-QFS) software provides a configurable file system with storage, archive management, and retrieval capabilities. The SAM-QFS software archives files by copying them from an online disk cache to archive media. The archive media can consist of the following:

- Disk slices in another file system
- Removable tape or magneto-optical cartridges in automated or manually loaded storage devices

The SAM-QFS software automatically maintains online disk space at site-specified usage thresholds. It releases disk space associated with archived file data and restores the files to an online disk when they are needed.

## Components of SAM-QFS

The SAM-QFS archiving lifecycle consists of the phases described in the following sections:

- “Archiving” on page 15
- “Releasing” on page 16
- “Staging” on page 16
- “Recycling” on page 16

### Archiving

The *archiver* automatically copies online disk cache files to archive media. The archive media can consist of either online disk files or removable media cartridges. The archiver requires that you configure the `archiver.cmd` file to define what to archive. You can configure the archiver to create up to four archive copies on a variety of archive media. If a file is segmented, each

segment is treated as a file and is archived separately. The archiving process is initiated when files match a site-definable set of selection criteria. For more information, see [Chapter 9, “About Archiving.”](#)

## Releasing

The *releaser* automatically maintains the file system's online disk cache at site-specified percentage usage thresholds by freeing disk blocks that are occupied by eligible archived files.

Releasing is the process of freeing primary (disk) storage that is used by an archived file's data. The high-water mark and low-water mark, expressed as a percentage of total disk space, are used to manage free space in the online disk cache. When online disk consumption exceeds the high-water mark, the system automatically begins to release the disk space that is occupied by eligible archived files. Disk space occupied by archived file data is released until the low-water mark is reached.

Files are selected for release depending on their size and age. If a file has been archived in segments, portions of the file can be released individually. The first portion of a file can be retained on disk for speedy access and for masking staging delays. For more information, see [Chapter 14, “About Releasing.”](#)

## Staging

The *stager* restores file data to the disk cache. When a user or process requests file data that has been released from disk cache, the stager automatically copies the file data back to the online disk cache.

When a file whose data blocks have been released is accessed, the stager automatically stages the file or file segment data back to the online disk cache. The read operation immediately follows the staging operation, which enables the file to be immediately available to an application before the entire file is completely staged.

The SAM-QFS software processes stage request errors automatically. If a stage error is returned, the system attempts to find the next available archive copy of the file. Stage errors that can be automatically processed include media errors, unavailability of media, unavailability of an automated library, and others. For more information, see [Chapter 15, “Configuring the Stager.”](#)

## Recycling

The *recycler* clears archive volumes of expired archive copies and makes volumes available for reuse.

As users modify files, archive copies that are associated with the old versions of these files are considered to be expired. Because these copies are no longer needed, they can be purged from the system. The recycler identifies the archive volumes with the largest proportions of expired archive copies and preserves the unexpired copies by moving them to separate volumes. The recycling process is transparent to end users.

If a removable media volume contains only expired copies, you can take one of the following actions:

- Relabel the volume for immediate reuse.
- Export the volume to offsite storage as a historical record of file changes. You can use standard UNIX utilities to restore previous versions of files from expired archive copies.

For more information, see [Chapter 16, “Configuring the Recycler.”](#)

## Supported Storage Devices

The SAM-QFS environment supports a wide variety of tape storage and magneto-optical devices. The automated libraries that SAM-QFS supports can be divided into the following groups, depending on how they are attached to the environment:

- *Direct attachment* — A direct-attached library is connected directly to the host system using a Small Computer System Interface (SCSI). This connection can be either a direct connection or a Fibre Channel connection. For example, a direct attachment is used for StorageTek libraries from Oracle. The SAM-QFS system controls these libraries directly by using the SCSI standard for automated libraries.
- *Network attachment* — The SAM-QFS software can be configured as a client of the library's host system. The network-attached libraries include some of the StorageTek, ADIC/Grau, IBM, and Sony libraries. These libraries use a software package that is supplied by the vendor. In these cases, the SAM-QFS software works with the vendor software by using a daemon that is designed for the automated library.

The relationships between the devices that are managed within the SAM-QFS environment are defined in the master configuration file, `/etc/opt/SUNWsamfs/mcf`. The `mcf` file specifies the removable media devices, libraries, and file systems that are included in the SAM-QFS environment. Each piece of equipment is assigned a unique equipment identifier in the `mcf` file. Entries in the `mcf` file also define manually mounted archiving devices and automated library catalog files.

When possible, the system uses the standard Oracle Solaris disk and tape device drivers. For devices that are not directly supported in the Oracle Solaris OS, such as certain library and optical disc devices, the SAM-QFS software packages include special device drivers.

For a list of supported storage devices, contact your Oracle sales representative. For information about configuring storage devices, see [Chapter 2, “Configuring Storage Devices for Archiving.”](#)

## SAM-Remote Software

Sun SAM-Remote software is a client/server implementation that enables libraries and other removable media devices to be shared between SAM-QFS host systems. Sun SAM-Remote software enables you to configure multiple storage clients that archive and stage files from a centralized tape library or magneto-optical library. For example, if you have host systems on a network that spans a large geographical area, files that are created in one city can be archived to cartridges in a library located miles away.

For more information, see [Chapter 18, “Using the Sun SAM-Remote Software.”](#)

# Configuring Storage Devices for Archiving

---

Perform the tasks in this chapter only if you plan to enable archiving to tape or magneto-optical media.

## Task Map: Configuring Storage Devices for Archiving

The following table shows the procedures for configuring storage devices for archiving.

Task	Description	More Information
Create list of devices	Take an inventory of your devices to configure.	See <a href="#">Table 2-1</a> .
Add tape devices	Perform this task for each tape drive that you want to add to the SAM-QFS environment.	See “ <a href="#">Adding Tape Devices for Archiving</a> ” on page 20 “ <a href="#">Adding Tape Devices for Archiving</a> ” on page 20.
Add tape drive interface target IDs and LUNs	Perform this task if your tape drives are attached through a SCSI or FC interface.	See “ <a href="#">How to Add Tape Drive Interface Target IDs and LUNs for Archiving (Command Line)</a> ” on page 23.
Add libraries or magneto-optical drives	Perform this task if you have any magneto-optical drives, SCSI-attached automated libraries, or FC-attached automated libraries that you want to include in your SAM-QFS environment.	See “ <a href="#">Adding Libraries or Magneto-Optical Drives for Archiving</a> ” on page 24.
Verify configured devices	Verify that you have all of your devices properly configured.	See “ <a href="#">Verifying and Implementing Configurations</a> ” on page 26.

Task	Description	More Information
Enable the storage device configuration	Reboot the system so that your changes take effect.	
Create parameters files	Create parameter files for the network attached automated libraries.	See Chapter 4, “Creating Parameters Files for Network-Attached Automated Libraries.”

## Adding Tape Devices for Archiving

The procedures in “[Task Map: Configuring Storage Devices for Archiving](#)” on page 19 include an example that is based on the inventory list shown in the following table.

TABLE 2-1 Inventory List — Devices to Be Configured

Device Name, Manufacturer, and Model	Target ID	LUN	Node WWN
<i>SCSI-attached tape drives</i>			
QUANTUM DLT7000	1	0	Not applicable
QUANTUM DLT7000	2	0	Not applicable
<i>FC-attached tape drives</i>			
StorageTek 9840	Not applicable	0	500104f00043abfc
StorageTek 9840	Not applicable	0	500104f00045eeaf
IBM ULT3580-TD1	Not applicable	0	500104f000416304
IBM ULT3580-TD1	Not applicable	0	500104f000416303
<i>SCSI-attached automated libraries</i>			
StorageTek 9730	0	0	Not applicable
<i>FC-attached automated libraries</i>			
StorageTek L700	Not applicable	0	500104f00041182b

**Note** – The device names are shown as they appear in the discovery output.

## Files that Contain Configuration Information

The following files contains configuration information:

- `/kernel/drv/st.conf` – Configures tape drives attached to the server through a SCSI or FC attachment.
- `/kernel/drv/samst.conf` – Configures the following devices that the Sun Storage Archive Manager (SAM) software recognizes by default:
  - Direct-attached automated libraries
  - Magneto-optical drives attached to the server through a SCSI or FC attachment

The SAM package includes the `/opt/SUNWsamfs/examples/st.conf_changes` file, which contains configuration information for tape drives that are not supported in the Oracle Solaris kernel by default.

## ▼ How to Add Tape Devices for Archiving (Command Line)

You can also add tape devices from SAM-QFS Manager. For information, see [“How to Add Tape Devices for Archiving \(SAM-QFS Manager\)” on page 24](#).

### 1 Create a backup copy of the `/kernel/drv/st.conf` file.

For example:

```
# cp /kernel/drv/st.conf /kernel/drv/st.conf.orig
```

### 2 In the `/kernel/drv/st.conf` file, remove the pound character (#) from the `tape-config-list` entry.

### 3 Modify the `/opt/SUNWsamfs/examples/st.conf_changes` file.

For each tape drive on your inventory list, do the following:

#### a. Find the device definition entry.

For example, searching for the Quantum DLT 7000 tape drive that is in the example inventory finds the following entry:

```
"QUANTUM DLT7000", "DLT 7000 tape drive", "dlt7-tape",
```

#### b. Copy the entry from the `st.conf_changes` file to the `st.conf` file and place it after the `tape-config-list` line.

The following example shows the resulting `st.conf` file.

```
...
tape-config-list=
"QUANTUM DLT7000", "DLT 7000 tape drive", "dlt7-tape",
...
```

Note that the final string in the entry, which is enclosed in quotation marks, is the tape configuration value. In this example, the final string is `"dlt7-tape"`.

- c. In the `/opt/SUNwsamfs/examples/st.conf_changes` file, find the line that begins with this tape configuration value.

In this example, the value is:

```
dlt7-tape = 1,0x36,0,0xd679,4,0x82,0x83,0x84,0x85,3;
```

- d. Copy the tape configuration value to the `st.conf` file and place it after the device definition line.

The following example shows the lines now contained in the `st.conf` file.

```
...
tape-config-list=
"QUANTUM DLT7000", "DLT 7000 tape drive", "dlt7-tape";
dlt7-tape = 1,0x36,0,0xd679,4,0x82,0x83,0x84,0x85,3;
...
```

- e. Repeat these steps for each type of tape drive you add.

---

**Note** – In the `st.conf_changes` file, a tape configuration value is repeated for each device definition that uses the same tape configuration. In the `st.conf` file, include only one entry for each tape configuration value. For example, the Sony SDT-5000 and the Sony SDT-5200 tape drives both use "DAT" as the final string. A single entry for the DAT tape configuration value is sufficient.

---

- 4 Replace the comma character (,) that appears at the end of the last device definition line with a semicolon character (;).

The following example shows an `st.conf` file that has definitions for the Quantum DLT 7000, the StorageTek 9840, and the IBM ULT3580 tape drives. The semicolon is after "CLASS\_3580"

```
...
tape-config-list=
"QUANTUM DLT7000", "DLT 7000 tape drive", "dlt7-tape",
"STK 9840", "STK 9840 Fast Access", "CLASS_9840",
"IBM ULT3580-TD1", "IBM 3580 Ultrium", "CLASS_3580";
dlt7-tape = 1,0x36,0,0xd679,4,0x82,0x83,0x84,0x85,3;
CLASS_9840 = 1,0x36,0,0xd679,1,0x00,0;
CLASS_3580 = 1,0x24,0,0x418679,2,0x00,0x01,0;
...
```

- 5 Save your changes and close the `/opt/SUNwsamfs/examples/st.conf_changes` file.

Do not close the `st.conf` file because you continue to edit the file in the [“How to Add Tape Drive Interface Target IDs and LUNs for Archiving \(Command Line\)”](#) on page 23 procedure.

## ▼ How to Add Tape Drive Interface Target IDs and LUNs for Archiving (Command Line)

This procedure shows how to verify and, if necessary, add target ID and LUN entries to the `st.conf` file for each tape drive on your hardware inventory list that is attached through a SCSI or FC interface.

---

**Note** – Do not use this procedure to add interface information for magneto-optical drives. See “[Adding Libraries or Magneto-Optical Drives for Archiving](#)” on page 24.

---

- 1 Open the `/kernel/drv/st.conf` file.
- 2 Configure the SCSI Kernel interface to attach tape drives.
  - a. Search for entries that have the following format to locate the list of SCSI target IDs and LUNs:
 

```
name="st" class="scsi" target=target lun=lun;
```

*target* is the target ID for each SCSI drive found and *lun* is the corresponding LUN for each SCSI drive found.
  - b. Find the entries that correspond to each SCSI target ID and LUN on your hardware inventory list. See [Example 2-1](#).
 

Note that some entries might extend over two lines.

    - If an entry is preceded by a hash character (`#`), remove the character. A hash character marks a line as a comment.
    - If an entry is missing, create an entry for the SCSI target and LUN line you need. Follow the format shown in Step 2a and use the information in your hardware inventory list.
- 3 If you are not using the StorageTek SAN Foundation Software I/O stack, create a line for each FC-attached device after the SCSI target ID and LUN list. See [Example 2-2](#).
 

Use the following format:

```
name="st" parent="fp" lun=lun fc-port-wwn="world-wide-name"
```

*lun* specifies the LUN for the drive. *world-wide-name* specifies the World Wide Name (WWN) for the drive.
- 4 Save your changes, and close the `st.conf` file.

### Example 2-1 Entries Corresponding to Tape Drives

The following example shows that two entries correspond to the two Quantum DLT 7000 drives that are attached to LUN 0 and have target IDs 1 and 2, shown in [Table 2-1](#).

```
name="st" class="scsi" target=1 lun=0;
name="st" class="scsi" target=2 lun=0;
```

### Example 2-2 Adding Lines that Support Tape Drives

The following example shows the lines that support the StorageTek 9840 and IBM ULT3580 tape drives that are included in [Table 2-1](#).

```
name="st" parent="fp" lun=0 fc-port-wwn="500104f00043abfc"
name="st" parent="fp" lun=0 fc-port-wwn="500104f00045eeaf"
name="st" parent="fp" lun=0 fc-port-wwn="500104f000416304"
name="st" parent="fp" lun=0 fc-port-wwn="500104f000416303"
```

## ▼ How to Add Tape Devices for Archiving (SAM-QFS Manager)

- 1 In the left pane, click **Archive Media**.
- 2 In the **Tape Library Summary** window, click **Add**.
- 3 Follow the steps in the wizard to add the devices.

## Adding Libraries or Magneto-Optical Drives for Archiving

The `inquiry.conf` file lists all of the supported devices.

The `/kernel/drv/samst.conf` file contains a list of SCSI and FC entries and works with the `/opt/SUNWsamfs/examples/inquiry.conf` file to define the devices that can be included in a SAM-QFS environment.

The following procedures show how to verify the entries in the `samst.conf` file and to update the file, if necessary.

Update the `samst.conf` file as follows:

- If you have only network-attached automated libraries, you do not need to verify device support or update it.
- If you use a SCSI or FC interface to attach a tape library to the server, see [“How to Configure Device Support in SCSI or FC Environments Using SAM-QFS Manager”](#) on page 25 to discover the tape libraries with the current drive information.
- If you have a direct-attached library with a target number larger than 6 or a LUN identifier larger than 1, see [“How to Configure Device Support for a Direct-Attached Library”](#) on page 25.

## ▼ How to Configure Device Support in SCSI or FC Environments Using SAM-QFS Manager

Repeat this procedure for each device in your environment.

- 1 **On the SAM-QFS Manager Managed Hosts page, select the name of the server to which you want to add a library.**

The File Systems Summary page appears.

- 2 **Expand the Archive Media section and select Tape Libraries.**

The Tape Library Summary page appears.

- 3 **Click Add to start the Add a Library wizard.**

- 4 **Follow the steps to add the device.**

When you complete the wizard steps, the `samst.conf` file is automatically updated.

## ▼ How to Configure Device Support for a Direct-Attached Library

- 1 **Create a backup copy of the `/kernel/drv/samst.conf` file.**

```
# cp /kernel/drv/samst.conf /kernel/drv/samst.conf.orig
```

- 2 **Open the `/kernel/drv/samst.conf` file.**

- 3 **Include SCSI-attached magneto-optical drives or SCSI-attached libraries.**

- a. **Search for entries that have the following format to locate the list of SCSI targets and LUNs:**

```
name="samst" class="scsi" target=target lun=lun;
```

*target* is the target ID for each SCSI drive found and *lun* is the corresponding LUN for each SCSI drive found.

- b. **Find the entry that corresponds to each SCSI target ID and LUN on your inventory list.**

For example, the StorageTek 9730 automated library is attached to target 0 and LUN 0. The following line corresponds to that interface:

```
name="samst" class="scsi" target=0 lun=0;
```

Note that some entries might extend over two lines if it includes return characters.

- If an entry starts with a hash character (`#`), remove the character. A hash (`#`) character marks a line as a comment.

- If an entry is missing, create an entry for the SCSI target and LUN line. Follow the format shown in Step 3a and use the information in your hardware inventory list.
- 4 Create a line for each FC-attached magneto-optical drive or FC-attached automated library in your inventory list.**
- Use the following format:
- ```
name="samst" parent="fp" lun=lun fc-port-wwn="world-wide-name"
```
- lun* specifies the LUN for the drive and *world-wide-name* specifies the WWN for the drive.
- The following example shows the line added to support the StorageTek L700 tape drive in [Table 2-1](#).
- ```
name="samst" parent="fp" lun=0 fc-port-wwn="500104f00041182b"
```
- 5 Save your changes, and exit the `samst.conf` file.**

## Recovering After a Direct-Attached Library Fails to Initialize

A direct-attached library may fail to initialize after an ENOENT error while trying to open a `samst` device. The example below shows how to exclude a STK SL500 library so that the `samstsamst` device binds as needed at mount time. Add lines similar to the following to the `/kernel/drv/scsi_vhci.conf` file, replacing STK SL500 with the string for the type of library that you are using:

```
#  
device-type-scsi-options-list =  
"STK SL500", "disable-option";  
  
disable-option = 0x7000000;
```

For more information, see the `scsi_vhci(7D)` man page.

## Verifying and Implementing Configurations

Verify that all devices are configured and implement the modified or new device entries.

## ▼ How to Verify That All Devices Are Configured and Implement the Changes

### 1 Use the `cfgadm` command to list the devices that are included in the SAM-QFS environment.

For example:

```
# cfgadm -al
Ap_Id                                Type          Receptacle  Occupant    Condition
c0                                    scsi-bus     connected   configured  unknown
c0::dsk/c0t6d0                       CD-ROM       connected   configured  unknown
c1                                    fc-private   connected   configured  unknown
c1::500000e0103c3a91                 disk        connected   configured  unknown
c2                                    scsi-bus     connected   unconfigured unknown
c3                                    scsi-bus     connected   unconfigured unknown
c4                                    scsi-bus     connected   configured  unknown
c4::dsk/c4t1d0                       disk        connected   configured  unknown
c4::dsk/c4t2d0                       disk        connected   configured  unknown
c5                                    fc-fabric    connected   configured  unknown
c5::100000e00222ba0b                 disk        connected   unconfigured unknown
c5::210000e08b0462e6                 unknown     connected   unconfigured unknown
c5::210100e08b2466e6                 unknown     connected   unconfigured unknown
c5::210100e08b27234f                 unknown     connected   unconfigured unknown
c5::500104f00043abfc                 tape        connected   configured  unknown
c5::500104f00043bc94                 tape        connected   configured  unknown
c5::500104f00045eeaf                 tape        connected   configured  unknown
c5::500104f000466943                 tape        connected   configured  unknown
c5::500104f00046b3d4                 tape        connected   configured  unknown
c5::500104f0004738eb                 tape        connected   configured  unknown
c6                                    fc           connected   unconfigured unknown
c7                                    scsi-bus     connected   unconfigured unknown
c8                                    scsi-bus     connected   unconfigured unknown
usb0/1                               usb-kbd      connected   configured  ok
usb0/2                               usb-mouse    connected   configured  ok
usb0/3                               unknown     empty      unconfigured ok
usb0/4                               unknown     empty      unconfigured ok
```

### 2 Examine the output to ensure that it shows all the devices you want configured in your SAM-QFS environment.

If a device is not configured, use the `cfgadm` command to configure it. See the `cfgadm(1M)` man page.

You might receive an error similar to the following:

```
# cfgadm -c configure -o force_update c4::500104f000489fe3
cfgadm: Library error: failed to create device node: 500104f00043abfc: Device busy
```

Despite this error, the `cfgadm` command does process the request.

### 3 Reboot the system to have the changes you have made to the `st.conf` and `samst.conf` files take effect.

```
# reboot -- -r
```

## Handling Errors in the st.conf File

Errors can occur if the st.conf file is not configured properly during SAM-QFS software installation.

The following messages in the sam-log file indicate that the appropriate changes have not been made to /kernel/drv/st.conf.

```
May 18 12:38:18 baggins genu-30[374]: Tape device 31 is default
type. Update '/kernel/drv/st.conf'.
```

The following device log messages correspond to the sam-log message:

```
1999/05/18 12:34:27*0000 Initialized. tp
1999/05/18 12:34:28*1002 Device is QUANTUM , DLT7000
1999/05/18 12:34:28*1003 Serial CX901S4929, rev 2150
1999/05/18 12:34:28*1005 Known as Linear Tape(lt)
1999/05/18 12:34:32 0000 Attached to process 374
1999/05/18 12:38:18 1006 Slot 1
1999/05/18 12:38:18 3117 Error: Device is type default. Update /kernel/drv/st.conf
```

## Performing Additional SAM-QFS Configuration

---

This chapter outlines additional tasks that you might need to complete in order to finish the configuration of the Sun Storage Archive Manager (SAM-QFS) or Sun QFS environment. Some of these tasks are optional, depending on your specific environment.

---

**Note** – Before you begin, install the Sun QFS or the SAM-QFS software as described in *Sun QFS and Sun Storage Archive Manager 5.3 Installation Guide*.

---

### Sharing the File System With NFS Client Systems

This section describes how to share the file system with network file system (NFS) clients.

Some NFS mount parameters can affect the performance of an NFS-mounted Sun Storage Archive Manager file system. You can set these parameters in the `/etc/vfstab` file as follows:

- `timeo = n`. This value sets the NFS timeout to  $n$  tenths of a second. The default is eleven tenths of a second. For performance purposes, use the default value. You can increase or decrease the value appropriately for your system.
- `rsize = n`. This value sets the read buffer size to  $n$  bytes. In NFS 2, change the default value (8192) to 32768. In NFS 3, retain the default value of 32768.
- `wsize = n`. This value sets the write buffer size to  $n$  bytes. In NFS 2, change the default value (8192) to 32768. In NFS 3, retain the default value of 32768. For more information about these parameters, see the `mount_nfs(1M)` man page.

## ▼ How to NFS Share the File System

This procedure uses the Oracle Solaris `share` command to make the file system available for mounting by remote systems. The `share` commands are typically placed in the `/etc/dfs/dfsstab` file and are executed automatically by the Oracle Solaris OS when you enter `init` state 3.

### 1 Add a share command to the `/etc/dfs/dfsstab` file.

For example:

```
# share -F nfs -o rw=client1:client2 -d "SAM-FS" /samfs1
```

### 2 Use the `ps` command to determine whether `nfs.server` is running.

For example:

```
# ps -ef | grep nfsd
  root   694      1  0   Apr 29 ?           0:36 /usr/lib/nfs/nfsd -a 16
en17    29996 29940  0 08:27:09 pts/5    0:00 grep nfsd
# ps -ef | grep mountd
  root   406      1  0   Apr 29 ?           95:48 /usr/lib/autofs/automountd
  root   691      1  0   Apr 29 ?           2:00 /usr/lib/nfs/mountd
en17    29998 29940  0 08:27:28 pts/5    0:00 grep mountd
```

In this sample output, the lines that contain `/usr/lib/nfs` indicate that the NFS server is mounted.

### 3 If `nfs.server` is not running, start it:

```
# svcadm enable nfs/server
```

### 4 (Optional) . If you want to NFS-share the file system immediately, type the `share` command at a root shell prompt.

If no NFS-shared file systems exist when the Oracle Solaris OS boots, the NFS server is not started.

The following example shows the commands to use to enable NFS sharing. You must change to run level 3 after adding the first share entry to this file.

```
# init 3
# who -r
.          run-level 3  Dec 12 14:39      3   2   2
# share
-          /samfs1  -   "SAM-FS"
```

## ▼ How to Mount the File System on Clients

On the client systems, mount the server's file system at a convenient mount point.

---

**Note** – There can be a significant delay in the file system's response to NFS client requests if a requested file resides on a cartridge that must be loaded into a DLT tape drive, if all tape drives are full, or if drives are slow. As a consequence, the system might generate an error instead of retrying the operation. To avoid this situation, it is recommended that you mount the file system with either the `hard` option enabled or with the `soft`, `retrans`, and `timeo` options enabled. If you use the `soft` option, also specify `retrans=120` (or greater) and `timeo=3000`.

---

**1 On an NFS client system, add a line to the `/etc/vfstab` file to mount the server's file system at a convenient mount point.**

The following example mounts `server:/samfs1` on the `/samfs1` mount point:

```
server:/samfs1 - /samfs1 nfs - yes hard,intr,timeo=60
```

**2 Save and close the `/etc/vfstab` file.**

**3 Issue the `mount` command.**

For example, the following `mount` command mounts the `samfs1` file system:

```
# mount /samfs1
```

Alternatively, the automounter can do this, if you prefer. Follow your site procedures for adding `server:/samfs1` to your automounter maps. For more information, see the `automountd(1M)` man page.

## Editing the defaults.conf File

The `/opt/SUNWsamfs/examples/defaults.conf` file contains directives that control automated library actions in a Sun Storage Archive Manager environment. You can change these settings at any time after the initial installation, for example, to accommodate changes in your site's library information. If you change the information in the `defaults.conf` file after the system is running, you must then issue commands to propagate the `defaults.conf` file changes to the file system.

The following example shows lines from an example `defaults.conf` file. This file shows several parameters that can affect the configuration of an automated library.

**EXAMPLE 3-1** Example of a `defaults.conf` File

```
exported_media = unavailable
attended = yes
tape = lt
log = LOG_LOCAL7
timeout = 300
# trace
# all on
# endtrace
```

**EXAMPLE 3-1** Example of a defaults.conf File (Continued)

```

labels = barcodes_low
lt_delay = 10
lt_unload = 7
lt_blksize = 256

```

Another sample file is located in the /opt/SUNWsamfs/examples/defaults.conf file.

For more information about the file's content, see the “defaults.conf(4)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* man page.

## Features You Can Control From defaults.conf

This section describes two common features that you can control from the defaults.conf file. For more information, see “defaults.conf(4)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

### Barcodes

If you have a tape library that uses a barcode reader, you can configure the system to set the tape label equal to the first or last characters of the barcode label. You can set the labels directive in the defaults.conf file, as shown in the following table.

**TABLE 3-1** Labels Directive in the defaults.conf File

Directive	Action
labels = barcodes	Default. Uses the first six characters of the barcode as the label. This setting enables the archiver to label new media on blank media automatically if the tape is chosen.
labels = barcodes_low	Uses the last six characters of the barcode as the label.
labels = read	Reads the label from the tape. This setting prevents the archiver from labeling new media automatically.

If labels = barcodes or labels = barcodes\_low is in effect, the Sun SAM system writes a label before the write is started for any tape that is mounted for a write operation that is write enabled, is unlabeled, and has a readable barcode.

### Drive Timing Values

You can set the unload and unload wait time for devices using the dev\_unload and dev\_delay directives, respectively.

The format of the dev\_unload parameter is as follows:

```
dev_unload = seconds
```

For *dev*, specify the device type as described in the `mcf(4)` man page.

For *seconds*, specify the number of seconds that you want the system to wait after an unload command is issued. This gives the automated library time to eject the cartridge, open the door, and perform other operations before the cartridge is removed. The default is 0.

The format of the `dev_delay` directive is as follows:

```
dev_delay = seconds
```

For *dev*, specify the device type as specified in the `mcf(4)` man page.

For *seconds*, specify the minimum number of seconds that you want to have elapse between the time when a cartridge is loaded and the time when the same cartridge is able to be unloaded. The default is 30.

For example:

```
# hp_delay = 10
# lt_unload = 7
```

## ▼ How to Customize Default Values

- 1 Use the `cp` command to copy `/opt/SUNWsamfs/examples/defaults.conf` to its functional location.**

For example:

```
# cp /opt/SUNWsamfs/examples/defaults.conf /etc/opt/SUNWsamfs/defaults.conf
```

- 2 Edit the file.**

Edit the lines that control those aspects of the system that you want to change. Remove the hash character (#) from column 1 of the lines you change.

- 3 Use the `pkill` command to send a `SIGHUP` signal to the `sam-fsd` daemon.**

For example:

```
# pkill -HUP sam-fsd
```

This command restarts the `sam-fsd` daemon and enables the daemon to recognize the changes in the `defaults.conf` file.

# Configuring the Remote Notification Facility

The software can be configured to notify you when potential problems occur in its environment. The system sends notification messages to a management station of your choice. The Simple Network Management Protocol (SNMP) software within the software manages the exchange of information between network devices such as servers, automated libraries, and drives.

The Sun SAM Management Information Base (MIB) defines the types of problems, or events, that the Sun SAM software can detect. The software can detect errors in configuration, tape alert events, and other atypical system activity. For more information, see the `/var/snmp/mib/SUN-SAM-MIB.mib` file.

The following procedures describe how to enable and disable remote notification.

## ▼ How to Enable Remote Notification

**Before You Begin** Ensure that the management station is configured and known to be operating correctly.

### 1 Examine the `/etc/hosts` file contents to ensure that the management station to which notifications should be sent is defined.

The following sample file defines a management station with a host name of `mgmtconsole`.

```
999.9.9.9      localhost
999.999.9.999  loggerhost    loghost
999.999.9.998  mgmtconsole
999.999.9.9    samserver
```

- If the management station is defined, close the file.
- If the management station is not defined, add a definition and save and close the file.

### 2 Open the `/etc/opt/SUNwsamfs/scripts/sendtrap` file and evaluate the `TRAP_DESTINATION='hostname'` directive.

This line specifies that remote notification messages be sent to port 161 of the server upon which the Sun Storage Archive Manager software is installed.

- If you want to change the host name or port, replace the `TRAP_DESTINATION` directive line with `TRAP_DESTINATION="mgmt-console-name:port"`. Note the use of quotation marks (") rather than apostrophes (') in the new directive.
- If you want to send remote notification messages to multiple hosts, specify the directive in the following format:

```
TRAP_DESTINATION="mgmt-console-name:port [mgmt-console-name:port]"
```

For example:

```
TRAP_DESTINATION="localhost:161 doodle:163 mgmt_station:1162"
```

- 3 **Evaluate the `COMMUNITY="pubLiC"` directive. This line acts as a password that prevents unauthorized viewing or use of SNMP trap messages.**

Do one of the following, depending on the community string value of your management station:

- If your management station's community string is also set to `pubLiC`, you do not have to edit this value.
- If your management station's community string is set to a value other than `pubLiC`, edit the directive to replace `pubLiC` with the value that is used in your management station.

- 4 **Save your changes to `/etc/opt/SUNWsamfs/scripts/sendtrap` and exit the file.**

## ▼ How to Disable Remote Notification

The remote notification facility is enabled by default. If you want to disable remote notification, perform this procedure.

- 1 **If the `/etc/opt/SUNWsamfs/defaults.conf` file does not exist, use the `cp` command to copy `/opt/SUNWsamfs/examples/defaults.conf` to `/etc/opt/SUNWsamfs/defaults.conf`.**

- 2 **In the `/etc/opt/SUNWsamfs/defaults.conf` file, find the line that specifies SNMP alerts.**

```
# alerts=on
```

- 3 **Remove the `#` symbol and change `on` to `off`.**

After editing, the line is as follows:

```
# alerts=off
```

- 4 **Save your changes and exit the file.**

- 5 **Use the `pkill` command to send a `SIGHUP` signal to the `sam-fsd` daemon.**

```
# pkill -HUP sam-fsd
```

This command restarts the `sam-fsd` daemon and enables the daemon to recognize the changes in the `defaults.conf` file.

## Adding the Administrator Group

By default, only the superuser can execute administrator commands. However, during installation you can supply an administrator group name. The `pkgaddprocess` prompts you for this group name during the installation.

Members of the administrator group can execute all administrator commands except for `star`, `samfsck`, `samgrowfs`, `sammkfs`, and `samd`. The administrator commands are located in `/opt/SUNWsamfs/sbin`.

After installing the package, you can use the `set_admin` command to add or remove the administrator group. This action performs the same function as selecting an administrator group during the package installation. You must be logged in as superuser to use the `set_admin` command. You can also undo the effect of this selection and make the programs in `/opt/SUNWsamfs/sbin` executable only by the superuser. For more information, see “[set\\_admin\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## ▼ How to Add the Administrator Group

- 1 Choose a group name, or select a group that already exists within your environment.
- 2 Use the `groupadd` command, or edit the `/etc/group` file.

The following example is an entry from the group file designating an administrator group for the software. In this example, the `samadm` group consists of both the `adm` and `operator` users.

```
samadm : 1999 : adm,operator
```

## ▼ How to Enable System Logging

The software logs errors, cautions, warnings, and other messages using the standard Oracle Solaris `syslog` interface. By default, the Sun SAM facility is `local7`.

- 1 Open the `/etc/syslog.conf` file.
- 2 In the `/opt/SUNWsamfs/examples/syslog.conf_changes` file, locate the logging line, which is similar to the following example:

```
local7.debug /var/adm/sam-log
```

---

**Note** – The preceding entry is all one line and has a TAB character (not a space) between the fields.

---

The default facility is `local7`. If you set logging to something other than `local7` in the `/etc/syslog.conf` file, edit the `defaults.conf` file and reset it there, too. For more information, see “[defaults.conf\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

- 3 **Append the logging line from `/opt/SUNWsamfs/examples/syslog.conf_changes` to the `/etc/syslog.conf` file.**

For example:

```
# cp /etc/syslog.conf /etc/syslog.conf.orig
# cat /opt/SUNWsamfs/examples/syslog.conf_changes >> /etc/syslog.conf
```

- 4 **Create an empty log file and send the `syslogd` a HUP signal.**

For example, to create a log file in `/var/adm/sam-log` and send the HUP to the `syslogd` daemon, type the following:

```
# touch /var/adm/sam-log
# pkill -HUP syslogd
```

For more information, see the `syslog.conf(4)` and `syslogd(1M)` man pages.

- 5 **(Optional) Use the `log_rotate.sh` command to enable log file rotation.**

Log files can become very large, and the `log_rotate.sh` command can help in managing log files. For more information, see the `log_rotate.sh(1M)` man page.

## Configuring Other Sun Storage Products

The Sun SAM-QFS installation and configuration process is complete. You can configure related storage products at this time. For example, to configure the Sun SAM-Remote software, see [Chapter 18, “Using the Sun SAM-Remote Software.”](#)



## Creating Parameters Files for Network-Attached Automated Libraries

---

If you have network-attached automated libraries in your Sun Storage Archive Manager (SAM-QFS) environment, you must have a parameters file for each library. You must configure your storage devices first, as described in [Chapter 2, “Configuring Storage Devices for Archiving.”](#)

You can include automated libraries in a SAM-QFS environment either by directly attaching them to the server or by attaching them to the environment’s network. Libraries that are attached through a SCSI or Fibre Channel (FC) attachment are direct-attached libraries. Libraries that are attached through a network attachment are network-attached libraries. This chapter describes how to create a parameters file for each network-attached library to be included in your environment.

---

**Note** – The examples and the discussions in the following sections mention both the parameters files for network-attached automated libraries and the `mc f` file. The `mc f` file is the main configuration file for the SAM-QFS software. For more information about creating an `mc f` file, see [Chapter 7, “Configuring the File System Environment,”](#) in *Sun QFS and Sun Storage Archive Manager 5.3 Installation Guide*. The parameters file and the `mc f` file are both mentioned in this section because the two files reference each other.

---

## Creating Parameters Files for Network-Attached Automated Libraries

You must create a parameters file for each network-attached library to be included in your environment.

## ▼ How to Create a Parameters File for Network-Attached Automated Libraries

### 1 Change to the `/etc/opt/SUNWsamfs` directory.

Although the parameters file can be written to any directory, the most common location is `/etc/opt/SUNWsamfs`.

---

**Note** – When you created your `mcf` file, you included the full path name to the parameters files. Ensure that the `mcf` file points to the correct location for the parameters files that you create.

---

### 2 Create a new file with a name that corresponds to the library that you are configuring.

For example, for a Sony network-attached library, you might name the file `sonyparams`.

### 3 Type a list of parameter entries in the file.

See the sections in this chapter for details about the parameter entries that should be included for specific library types.

## Configuring a StorageTek ACSLS-Attached Automated Library Parameters File

In many respects, the way in which SAM-QFS systems interoperate with StorageTek ACSLS-attached automated libraries is very similar to the way in which they interoperate with direct-attached automated libraries. However, the installation and configuration procedure of a StorageTek ACSLS-attached automated library requires additional steps.

The StorageTek ACSLS software package controls the automated library. Daemon software controls the StorageTek automated library through the ACSAPI interface.

---

**Note** – The SAM-QFS Manager supports the automatic discovery and configuration of ACSLS network-attached libraries. You do not need to configure the parameters file before configuring the library in SAM-QFS Manager. For more information, see the SAM-QFS Manager online help.

---

Before you begin, ensure that the following are true and that the StorageTek ACSLS-attached automated library is prepared for inclusion in a SAM-QFS environment:

- The StorageTek ACSLS automated library is operational.
- The StorageTek ACSLS software package is installed and working.

For instructions describing how to create an empty parameters file, see [“How to Create a Parameters File for Network-Attached Automated Libraries”](#) on page 40.

Type a list of parameter entries in the StorageTek parameters file.

The following table shows the keywords to use.

Parameter	Description
<code>access = user-id</code>	(Optional) Specifies the user identification value that is used by the StorageTek software for access control. If this parameter is not supplied, the access control string is a null string, indicating that there is no user - id.
<code>hostname = hostname</code>	Specifies the host name of the server that runs the StorageTek ACSLS interface.
<code>portnum = portnum</code>	Specifies the port number that is used for communication between ACSLS and the SAM-QFS software. For information about the <i>portnum</i> argument, see “ <a href="#">stk(7)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<code>ssihost = hostname</code>	Specifies the name of the SAM-QFS server on the LAN that connects to the ACSLS host. Specify this parameter only if you are including a multihomed SAM-QFS server in your environment. The default is the name of the local host.
<code>ssi_inet_port = ssi_inet-port</code>	Specifies the fixed port number for incoming responses and specifies the port that the SSI uses for incoming ACSLS responses in a firewall environment. Specify either 0 or a value from 1024 to 65535. A non-zero value forces SSI to use this port for incoming ACSLS responses.
<code>csi_hostport = csi-port</code>	Specifies the port on the ACSLS server to which the StorageTek SSI daemon sends its ACSLS requests. Specify either 0 or a value from 1024 to 65535, inclusive. If set to 0, or left unset, the system queries the port mapper on the ACSLS server.
<code>capid = ( acs = acsnum, lsm = lsmnum, cap = capnum )</code>	Specifies the cartridge access point (CAP), in terms of the StorageTek library, to be used when the <code>export -f</code> command is specified. The <i>capid</i> description starts with an open parenthesis followed by three name-value pairs followed by a closing parenthesis. Use a comma, a colon, or a space to separate the name-value pairs. For <i>acsnum</i> , specify the asynchronous communications server (ACS) number for this CAP as configured in the StorageTek library. For <i>lsmnum</i> , specify the length subnet mask (LSM) number for this CAP as configured in the StorageTek library. For <i>capnum</i> , specify the CAP number for this CAP as configured in the StorageTek library.
<code>capacity = ( index = value, [index = value]... )</code>	Specifies the capacities of the supported cartridges. Use a comma to separate the name-value pairs, and enclose the string in parentheses.  For <i>index</i> , specify the index of the supplied <i>media_type</i> file, which is located in the following ACSLS directory: <code>/export/home/ACSSS/data/internal/mixed_media/media_types.dat</code>  For <i>value</i> , specify the capacity of the cartridge type in units of 1024 bytes. In general, supplying a capacity entry is necessary only for an index of new cartridge types or to override the supported capacity.

Parameter	Description
device-path-name = (acs = value, lsm = value, panel = value, drive = value ) [shared]	<p>Specifies the path to the device on the client. Specify one <i>device-path-name</i> = entry for each drive that is attached to this client. This parameter describes the drive within the StorageTek automated library. This description starts with an open parenthesis followed by four <i>keyword = value</i> pairs and a closing parenthesis. Use a comma, a colon, or a space to separate the <i>keyword = value</i> pairs.</p> <p>The shared keyword is optional and it specifies that the drive can be shared between two or more SAM processes from two or more hosts.</p> <p>For more information about implementing shared drives, see <a href="#">“About Shared Drives” on page 48</a> and the <code>stk(7)</code> man page.</p> <p>For the <i>value</i> specifications, use the following information that is supplied by the ACSLS query drive command:</p> <ul style="list-style-type: none"> <li>■ <code>acs</code> – ACS number of the drive as configured in the StorageTek library</li> <li>■ <code>lsm</code> – LSM number of the drive as configured in the StorageTek library</li> <li>■ <code>panel</code> – PANEL number of the drive as configured in the StorageTek library</li> <li>■ <code>drive</code> – DRIVE number of the drive as configured in the StorageTek library</li> </ul>

The following example shows a parameters file for a StorageTek ACSLS-attached automated library:

```
#
# This is file: /etc/opt/SUNWsamfs/stk50
#
hostname = baggins
portnum = 50014
access = some_user # No white space allowed in user_id
ssi_inet_port = 0
csi_hostport = 0
capid = (acs=0, lsm=1, cap=0)
/dev/rmt/0cbn = (acs=0, lsm=1, panel=0, drive=1) shared
/dev/rmt/1cbn = (acs=0, lsm=1, panel=0, drive=2)
```

**Note** – [“Configuration Example for a Shared File System on an Oracle Solaris OS Platform” in Sun QFS File System 5.3 Configuration and Administration Guide](#) shows an example StorageTek ACSLS-attached automated library parameters file. The example `mcf` file points to file `stk50` in the `/etc/opt/SUNWsamfs` directory.

## Configuring an ADIC/Grau Automated Library Parameters File

The ADIC/Grau automated library operates within SAM-QFS environments through the `grauaci` interface. This interface uses the DAS/ACI 3.12 interface that is supplied by ADIC/Grau. For more information about the DAS/ACI interface, see your ADIC/Grau documentation.

---

**Note** – ADIC/Grau network attached libraries are not supported by the SAM-QFS software on an x64 hardware platform.

---

Before you begin, ensure that the following are true and that the ADIC/Grau automated library is prepared for inclusion in a SAM-QFS environment:

- The ADIC/Grau automated library is operational.
- The ADIC/Grau library is operating on the Distributed AML Server (DAS).
- Both the `avc` (avoid volume contention) and the `dismount` parameters are set to `true` in the DAS configuration file for this client.

For instructions describing how to create an empty parameters file, see [“How to Create a Parameters File for Network-Attached Automated Libraries”](#) on page 40.

Type a list of parameter entries in the ADIC/Grau parameters file.

The parameters are specified as name-value pairs, such as *keyword=value*. The various parameters identify the ADIC/Grau automated libraries, the drives associated with the libraries, and the server name. All parameter entries are case-sensitive, so enter them exactly as specified in the DAS configuration file and in the `mc f` file.

The following table shows the parameters that must appear in the ADIC/Grau parameters file.

Parameter	Description
<code>client = client-id</code>	The name of the client as defined in the DAS configuration file. This parameter is required.
<code>server = server-id</code>	The host name of the server that runs the DAS server code. This parameter is required.
<code>acidrive drive-id = path</code>	The name of the drive as configured in the DAS configuration file. <i>path</i> specifies the path to the drive as entered in the Equipment Identifier field of the <code>mc f</code> file. Include one <code>acidrive</code> line for every drive assigned to the client.

Comments can appear anywhere on any line, and they must begin with a hash character (`#`). The system ignores characters to the right of the hash.

If the ADIC/Grau library contains various media types, a media changer exists for each media type. Each media changer has a unique client name in the DAS configuration, a unique library catalog, and a unique parameters file.

The following sample ADIC/Grau parameters file defines one ADIC/Grau automated library that supports DLT tape and one ADIC/Grau automated library that supports a Hewlett-Packard optical drive.

```
# This is file: /etc/opt/SUNWsamfs/grau50
#
client = DASclient
server = DAS-server
#
# the name "drive1" is from the DAS configuration file
#
acidrive drive1 = /dev/rmt/0cbn
#
# the name "drive2" is from the DAS configuration file
#
acidrive drive2 = /dev/rmt/1cbn
```

---

**Note** – “[Configuration Example for a Shared File System on an Oracle Solaris OS Platform](#)” in *Sun QFS File System 5.3 Configuration and Administration Guide* shows an example ADIC/Grau network-attached automated library parameters file. The example `mcf` file points to the `grau50` file in the `/etc/opt/SUNWsamfs` directory.

---

The `/var/opt/SUNWsamfs/.grau` directory contains diagnostic information that can be useful for troubleshooting.

The system creates files in this directory that are named `grau $log$ - $eq$` , where  $eq$  is the Equipment Ordinal as defined in the `mcf` file. For more information, see “[grauaci\(7\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* and “[mcf\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Configuring an IBM 3494 Automated Library Parameters File

The IBM 3494 automated tape library operates in SAM-QFS environments with the assistance of the IBM `lmcpd` daemon package. You can obtain the IBM `lmcpd` daemon package from IBM.

---

**Note** – IBM 3494 network attached libraries are not supported by the SAM-QFS software on an x64 hardware platform.

---

Before you begin, ensure that the following are true and the IBM 3494 automated library is prepared for inclusion in a SAM-QFS environment:

- The IBM 3494 automated library is operational.
- The IBM `lmcpd` daemon package is installed and working.
- The `/etc/ibmatl.conf` file is configured and working.
- The IBM 3494 automated library can be used as a single physical library or as multiple logical libraries. If you divide this library into multiple logical libraries, create a parameters file for each logical library.

For instructions describing how to create an empty parameters file, see [“How to Create a Parameters File for Network-Attached Automated Libraries” on page 40](#).

Type a list of parameter entries (`keyword = value` and `pathname = value`) in the IBM 3494 parameters file.

All arguments are case sensitive. The following table shows how to specify the parameters.

Parameter	Description
<code>name = name</code>	The name assigned by you as system administrator, and specified in the <code>/etc/ibmatl.conf</code> file. This name is also the symbolic name of the library. This parameter must be supplied. There is no default value.
<code>category = hexnumber</code>	A hexadecimal number between <code>0x0001</code> and <code>0xfeff</code> . By default, the SAM-QFS software sets this value to 4 for media under its control. If you have divided your physical library into multiple logical libraries, ensure that the value of <code>hexnumber</code> is different in each logical library. This parameter specifies which tapes are assigned to which library. When you import media into the library, they are added to the catalog, and their <code>category = value</code> is changed to the value specified by this <code>category = hexnumber</code> parameter.
<code>access = permission</code>	Valid values are <code>shared</code> or <code>private</code> . <ul style="list-style-type: none"> <li>■ Specify <code>private</code> if you are using the library as one physical library. This is the default value.</li> <li>■ Specify <code>shared</code> if you are dividing the library into multiple logical libraries.</li> </ul>
<code>device-pathname = device-number</code>	For <code>device-pathname</code> , specify the path of a drive. You must have a <code>device-pathname</code> entry for every drive in the library that is attached to this machine. Each <code>device-pathname</code> entry must match the Equipment Identifier value of the corresponding entry in the <code>mcf</code> file. For <code>device-number</code> , the device number is described in the IBM documentation. You can obtain this number by running the IBM <code>mtLib</code> utility.

Comments can appear anywhere on any line and must begin with a hash character (`#`). The system ignores characters to the right of the hash sign.

The following sample `/etc/ibmatl.conf` file uses information obtained from the `mtlib` utility supplied by IBM.

```
#
# This is file: /etc/ibmatl.conf
# Set this file up according the documentation supplied by IBM.
3493a 198.174.196.50 test1
```

After the `lmcpsd` daemon is running, use the IBM `mtlib` utility to obtain the device numbers. The following example shows output from `mtlib`.

```
# mtLib -l 3493a -D
0, 00145340 003590B1A00
1, 00145350 003590B1A01
```

The following sample parameters file is for an IBM 3494 library.

```
#
# This is file: /etc/opt/SUNWsamfs/ibm50
#
name = 3493a # From /etc/ibmatl.conf
/dev/rmt/1bn = 00145340 # From mtlib output
/dev/rmt/2bn = 00145350 # From mtlib output
access=private
category = 5
```

---

**Note** – “[Configuration Example for a Shared File System on an Oracle Solaris OS Platform](#)” in *Sun QFS File System 5.3 Configuration and Administration Guide* shows an example IBM 3494 network-attached automated library parameters file. The example `mcf` file points to file `ibm50` in the `/etc/opt/SUNWsamfs` directory.

---

## Configuring a Sony Network-Attached Automated Library Parameters File

The Sony network-attached automated library operates within the SAM-QFS environment through the DZC-8000S Application Interface Library package. This software package provides the application programming interface (API) to the PetaSite Controller (PSC). For more information about the DZC-8000S interface, see the Sony PetaSite Application Interface Library DZC-8000S, which is available from Sony.

---

**Note** – Sony network-attached libraries are not supported by the SAM-QFS software on an x64 hardware platform.

---

Before you begin, ensure that the following are true and that the Sony network-attached automated library is prepared for inclusion in a SAM-QFS environment:

- The Sony network attached automated library is operational.
- The Sony PSC configuration file is installed and working.

Type a list of parameter entries in the Sony parameters file.

For instructions describing how to create an empty parameters file, see “[How to Create a Parameters File for Network-Attached Automated Libraries](#)” on page 40.

The various parameter values identify the Sony automated libraries, the drives associated with the libraries, and the host name. All parameter and value entries are case-sensitive, so type them exactly as they are specified in the configuration file and in the `mcf` file.

---

**Note** – The information in this section applies only to Sony automated libraries that are network attached through a Sony DZC-8000S interface. Sony direct-attached B9 and B35 automated libraries or Sony direct-attached 8400 PetaSite automated libraries do not require a parameters file.

---

The following table shows the parameters that must appear in the Sony parameters file. All parameters are required.

Parameter	Description
<code>userid = <i>user-id</i></code>	A number from 0 to 65535, inclusive. If you specify a number other than 0, it must be the PSC ID. The <code>user-id</code> parameter identifies the user during initialization of the PetaSite automated library functions.
<code>server = <i>server-id</i></code>	The host name of the server that runs the PSC server.
<code>sonydrive <i>drive-id</i> = <i>path</i></code>	For <i>drive-id</i> , the drive bin number as configured in the PSC configuration file. Include one <code>sonydrive</code> line for every drive defined in the <code>mc f</code> file. For <i>path</i> , specify the path to the drive as entered in the Equipment Identifier field of the <code>mc f</code> file.

Comments can appear anywhere on any line, but they must begin with a hash character (#). The system ignores characters to the right of the hash sign.

The following example shows a parameters file for a Sony network-attached automated library.

```
#
# This is file: /etc/opt/SUNWsamfs/sonyfile
#
# The userid identifies the user during initialization of
# the PetaSite library functions
#
userid = 65533
#
# europa is the hostname for the server running
# the DZC-8000S server code.
#
server = europa
#
# The bin numbers 1001 and 1002 are from the PSC
# configuration file.
#
sonydrive 1001 = /dev/rmt/1cbr
sonydrive 1002 = /dev/rmt/2cbr
```

---

**Note** – “[Configuration Example for a Shared File System on an Oracle Solaris OS Platform](#)” in *Sun QFS File System 5.3 Configuration and Administration Guide* shows an example Sony network-attached automated library parameters file. The example `mcf` file points to file `sonyfile` in the `/etc/opt/SUNWsamfs` directory.

---

## About Shared Drives

Typically, the SAM-QFS processes have exclusive control over a library's drives as declared in the host system's `mcf` file. In many cases, however, drives are defined in individual `mcf` files that are used by independent copies of SAM-QFS processes. If a process is not using a drive, the drive remains idle.

The shared-drives capability enables two or more `mcf` files to define the same drive, which makes the drive available to multiple SAM-QFS processes. However, these multiple processes cannot share media. Each SAM-QFS process must still maintain its own set of VSNs.

The shared-drives feature can be useful, for example, when a library is attached to more than one host system in a SAM-QFS environment. The SAM-QFS processes coordinate the use of a drive and keep the drives in a library busy.

You can configure some network-attached libraries to share one or all of the media drives between multiple SAM-QFS processes on multiple host systems. All of the StorageTek ACSLS-attached libraries support shared drives in SAM-QFS environments.

To implement one or more shared drives, specify the `shared` keyword in the parameters file for each drive that is to be shared. The placement of the `shared` keyword is specific to each manufacturer's library, so see the vendor-specific sections for more information.

---

**Note** – By default, a cartridge in a shared drive can be idle for 60 seconds before being unloaded. To change this timing, change the `shared_unload` value in the `defaults.conf` file to the new value, in seconds.

---

## Checking the Drive Order in Libraries

---

If your automated library contains more than one drive, the order of the drives in the `mc f` file must be the same as the order of the drives shown by the automated library's controller. The drive that the library controller identifies as the first drive must be the first drive entry for that library in the `mc f` file, and so on. This order can be different from the order of the devices as reported in the `/var/adm/messages` file.

This chapter provides procedures for checking the drive order. The procedure varies depending on whether your automated library has a front panel, whether it has tape or magneto-optical drives, and whether it is direct attached or network attached. Each procedure maps the library drives to SCSI target IDs, and then maps the SCSI target IDs to remote tape devices.

### Checking the Drive Order in Libraries

#### ▼ **How to Check the Drive Order of Libraries With a Front Panel**

Some libraries have a panel that display the drive information.

The following procedure is a general plan. The actual steps depend on your specific library product, so consult your vendor documentation for information about drive identification and target identification.

- 1 **Verify the order of the drives according to the vendor's documentation.**
- 2 **In the front panel, check each drive's SCSI target ID or World Wide Name (WWN).**
- 3 **Record the order in which each drive and drive target is reported.**

- 4 In the `mc f` file, ensure that the order of the drive targets is the same order in which the drives are shown by the automated library's controller.
- 5 If you made any changes, verify the `mc f` file and test the drives. Then, propagate the change to the rest of the system.

To determine whether the drives become active when loaded with a cartridge, you can visually inspect the drives or use the `samu` utility's `r display`. For more information, see the [Sun QFS File System 5.3 Configuration and Administration Guide](#).

## ▼ How to Check the Drive Order of a Tape Library Without a Front Panel

- 1 Stop the SAM-QFS software so that no drives are used during the procedure.
- 2 Obtain a listing of devices in `/dev/rmt/`.

```
# ls -l /dev/rmt/?
lrwxrwxrwx 1 root root 42 Jan 10 2000 /dev/rmt/0 ->
../.. /devices/pci@1f,4000/scsi@2,1/st@2,0:
lrwxrwxrwx 1 root root 42 Jan 10 2000 /dev/rmt/1 ->
../.. /devices/pci@1f,4000/scsi@4,1/st@5,0:
lrwxrwxrwx 1 root root 42 Jan 10 2000 /dev/rmt/2 ->
../.. /devices/pci@1f,4000/scsi@4,1/st@6,0:
lrwxrwxrwx 1 root other 40 Dec 13 2000 /dev/rmt/3 ->
../.. /devices/pci@1f,4000/scsi@4/st@1,0:
lrwxrwxrwx 1 root root 40 Jun 20 2001 /dev/rmt/4 ->
../.. /devices/pci@1f,4000/scsi@4/st@2,0:
lrwxrwxrwx 1 root root 40 Jun 20 2001 /dev/rmt/5 ->
../.. /devices/pci@1f,4000/scsi@4/st@3,0:
lrwxrwxrwx 1 root root 40 Jun 20 2001 /dev/rmt/6 ->
../.. /devices/pci@1f,4000/scsi@4/st@4,0:
lrwxrwxrwx 1 root root 40 Sep 14 2001 /dev/rmt/7 ->
../.. /devices/pci@1f,4000/scsi@2/st@2,0:
lrwxrwxrwx 1 root root 40 Sep 14 2001 /dev/rmt/8 ->
../.. /devices/pci@1f,4000/scsi@2/st@3,0:
lrwxrwxrwx 1 root root 40 Sep 14 2001 /dev/rmt/9 ->
../.. /devices/pci@1f,4000/scsi@2/st@4,0:
```

- 3 Load a tape into the library's Drive 1.

Ensure that the other drives are empty by running the `samload` command.

- 4 Obtain information about the drive and tape position by running the following command with each `/dev/rmt/entry`:

```
# mt -f /dev/rmt/x status
```

The `/dev/rmt/x` entry returns information corresponding to the library's Drive 1. The following example shows `mt` command output that indicates that a tape is in the drive.

```
# mt -f /dev/rmt/0 status
DLT 7000 tape drive tape drive:
  sense key(0x2)= Not Ready  residual= 0  retries= 0
  file no= 0  block no= 0
```

### 5 Repeat the steps for each library drive.

Create a table that shows which library drive corresponds to each `/dev/rmt/` entry.

For example:

```
drive 1 = /dev/rmt/4 -> ../../devices/pci@1f,4000/scsi@4/st@2,0:
drive 2 = /dev/rmt/7 -> ../../devices/pci@1f,4000/scsi@2/st@2,0:
...
```

### 6 Update the `mcf` file to list the drives in the order in which they are shown by the automated library's controller.

In this case, the `mcf` file starts with the following items:

```
# Equipment      Eq  Eq  Family  Device  Additional
# Identifier      Ord Type Set    State  Parameters
#-----
/dev/rmt/4        31  li  ibm3580  on
/dev/rmt/7        32  li  ibm3580  on
...
```

### 7 Verify the `mcf` file and test the drives.

### 8 Propagate the change to the rest of the system.

For more information, see the *Sun QFS File System 5.3 Configuration and Administration Guide*

## ▼ How to Check the Drive Order of Magneto-Optical Libraries Without a Front Panel

### 1 Stop the SAM-QFS software so that no drives are used during the procedure.

### 2 Obtain a listing of devices in `/dev/samst/`.

```
# ls -l /dev/samst/
```

### 3 Load a magneto-optical cartridge manually through the library front panel into the library's Drive 1.

Ensure the other drives are empty.

- 4 Obtain information about the drive and tape position by running the following command with each `/dev/samst/` entry:

```
# dd if=/dev/samst/x bs=2k iseek=3374 of=/tmp/foo count=10
```

The `/dev/samst/` entry that returns information corresponds to the library's Drive 1. The following example shows a status message that indicates that an optical cartridge is in the selected device.

```
# dd if=/dev/samst/c0t3u0 bs=2k iseek=3374 of=/tmp/junk count=10
10+0 records in
10+0 records out
```

- 5 Repeat the steps for each library drive.

Create a table that shows which library drive corresponds to each `/dev/samst/` entry:

```
drive 1 = /dev/samst/4 -> ../../devices/pci@1f,4000/scsi@4/st@2,0:
drive 2 = /dev/samst/7 -> ../../devices/pci@1f,4000/scsi@2/st@2,0:
...
```

- 6 Update the `mc f` file to list the drives in the order in which they are shown by the automated library's controller.

In this case, the `mc f` file starts with the following items:

# Equipment # Identifier #-----	Eq Ord	Eq Type	Family Set	Device State	Additional Parameters
-----	----	----	----	----	-----
/dev/samst/4	31	li	ibm3580	on	
/dev/samst/7	32	li	ibm3580	on	
...					

- 7 Verify the `mc f` file and test the drives.

- 8 Propagate the change to the rest of the system.

For more information, see the [Sun QFS File System 5.3 Configuration and Administration Guide](#).

## ▼ How to Check the Drive Order of Network-Attached Libraries

- 1 Stop the SAM-QFS software so that no drives are used during the procedure.
- 2 Obtain a listing of devices in `/dev/rmt/`.

```
# ls -l /dev/rmt/*[0-9] | awk '{print $9, $10, $11}'
/dev/rmt/0 -> /devices/pci@8,700000/SUNW,qlc@4,1/fp@0,0/st@w500104f0006041f0,0:
/dev/rmt/1 -> /devices/pci@8,700000/SUNW,qlc@4,1/fp@0,0/st@w500104f0006041f3,0:
/dev/rmt/2 -> /devices/pci@8,700000/SUNW,qlc@4,1/fp@0,0/st@w500104f00043cbb8,0:
```

```

/dev/rmt/3 -> /devices/pci@8,700000/SUNW,qlc@5,1/fp@0,0/st@w500104f0006041ea,0:
/dev/rmt/4 -> /devices/pci@8,700000/SUNW,qlc@5,1/fp@0,0/st@w500104f0006041ed,0:
/dev/rmt/5 -> /devices/pci@8,700000/SUNW,qlc@4/fp@0,0/st@w500104f00060420e,0:
/dev/rmt/6 -> /devices/pci@8,700000/SUNW,qlc@4/fp@0,0/st@w500104f000604211,0:
/dev/rmt/7 -> /devices/pci@8,700000/SUNW,qlc@4/fp@0,0/st@w500104f000604214,0:
/dev/rmt/8 -> /devices/pci@8,700000/SUNW,qlc@5/fp@0,0/st@w500104f000604208,0:
/dev/rmt/9 -> /devices/pci@8,700000/SUNW,qlc@5/fp@0,0/st@w500104f00060420b,0:

```

Next, use `luxadm` output and output from an `ACSL`s display command to associate the serial numbers of each drive with the physical location in the library.

### 3 Display the serial number for each device.

```
# luxadm display /dev/rmt/x
```

### 4 Use the `ACSL`s display to show the drive identifier for each serial number.

```

ACSSA> display drive * -f serial_num
2007-10-11 10:49:12          Display Drive
Acs  Lsm  Panel  Drive  Serial_num
0    2    10    12    331000049255
0    2    10    13    331002044567
0    2    10    14    331002057108
0    2    10    15    331002042417
0    2    10    16    331002031352
0    2    10    17    HU92K00200
0    2    10    18    HU92K00208
0    3    10    10    1200019405
0    3    10    11    1200019442
0    3    10    12    1110150718

```

### 5 Create a table to show the relationships between the identifiers.

Device	SSN	Drive Identifier
/dev/rmt/0	-> 331000049255	-> (acs=0, lsm=2, panel=10, drive=12)
/dev/rmt/1	-> 331002044567	-> (acs=0, lsm=2, panel=10, drive=13)
/dev/rmt/2	-> 331002057108	-> (acs=0, lsm=2, panel=10, drive=14)

### 6 In the `mcf` file, verify that the order of the drives matches the table.

### 7 Verify the `mcf` file and test the drives.

### 8 Propagate the change to the rest of the system.

For more information, see the [Sun QFS File System 5.3 Configuration and Administration Guide](#).



## Populating the Catalog

---

After you mount a file system, the SAM-QFS software creates catalogs for each automated library that is configured in the `mc f` file. However, if you have a network-attached automated library, you must populate the library's catalog. The appropriate method depends on the number of volumes that you include in the catalog.

### Populating the Catalog

The appropriate method to use to populate a library's catalog depends on the number of volumes that you include in the catalog.

#### ▼ How to Populate an Automated Library With Many Volumes

Use this procedure for StorageTek ACSLS-Attached, ADIC/Grau, Sony network-attached, and IBM 3494 automated libraries.

Note the following when creating the input file:

- The file has four fields in each row. Each row identifies a volume. For each volume, specify the slot number, the VSN, the bar code, and the media type.

---

**Note** – The slot position of a tape in a network-attached automated library has no relationship to the slot number of the volume in a Sun Storage Archive Manager (SAM-QFS) library catalog.

---

- Use a space character or a tab character to separate the fields in this file.
- If a VSN contains one or more space characters, enclose the VSN name in quotation marks (" ").

- 1 **Create an input file that contains the slot number, the volume's VSN, the barcode number, and the media type.**

The following example shows the sample file `input_vsns`.

```
0 TAPE01 "TAPE 01" lt
1 TAPE02 TAPE02 lt
2 TAPE03 TAPE03 lt
```

- 2 **Use the `build_cat` command to create the catalog.**

```
# build_cat input-file catalog-file
```

Header	Header
<i>input-file</i>	Specifies the name of an input file, which is typically a file that contains a list of VSNs.
<i>catalog-file</i>	Specifies the full path to the library catalog. By default, the SAM-QFS software creates a catalog and writes it to <code>/var/opt/SUNWsamfs/catalog/family-set-name</code> . <i>family-set-name</i> is derived from the <code>mcf</code> file entry for this automated library. Alternatively, if you have specified a catalog name in the Additional Parameters field of the <code>mcf</code> file, use that catalog file name for <i>catalog-file</i> .

For example, you might specify the following `build_cat` command:

```
# build_cat input_vsns /var/opt/SUNWsamfs/catalog/grau50
```

## ▼ How to Populate an Automated Library With a Small Number of Volumes

Use this procedure for ADIC/Grau, Sony network-attached, StorageTek ACSLS-Attached, and IBM 3494 automated libraries.

Perform this procedure for each cartridge that you want to include in the catalog. The cartridge must be physically present in the automated library. If the cartridge is not present, the entry is recorded in the historian. The historian keeps track of cartridges that are exported from an automated library or a manually mounted device. For more information about the historian, see [“Tracking Exported Media - The Historian” on page 73](#).

- **Use the `samimport` command to import catalog entries into the default catalog.**

```
# samimport -v VSN eq
```

Argument	Content
VSN	Specifies the VSN identifier for a volume. If a VSN name contains one or more space characters, enclose the VSN name in double quotes (" ").

Argument	Content
<i>eq</i>	Specifies the Equipment Ordinal as specified for the device in the <code>mc f</code> file.

For example:

```
# samimport -v TAPE01 50
```

## ▼ How to Populate an IBM 3494 Automated Library

Use this procedure only if you are using an IBM 3494 library as one physical library where `access=private` is specified in the `mc f` file. Do not use this procedure if you divided the library into multiple logical libraries.

---

**Note** – If you have an IBM 3494 library that is divided into multiple logical libraries where `access=shared` is specified in the IBM 3494 parameters file, use one of the previous methods to populate the catalog: [“How to Populate an Automated Library With Many Volumes” on page 55](#) or [“How to Populate an Automated Library With a Small Number of Volumes” on page 56](#).

---

- **Insert the media cartridge into the mail slot.**

The library automatically builds a catalog that includes the media cartridge.

## ▼ How to Populate a StorageTek ACSLS-Attached Library Quickly

This procedure is a faster method of populating a library catalog than the methods described in [“How to Populate an Automated Library With Many Volumes” on page 55](#) or [“How to Populate an Automated Library With a Small Number of Volumes” on page 56](#).

- **Use the `samimport` command with the `-c` and `-s` options to import from a pool of VSNs.**

For more information, see [“samimport\(1M\)” in \*Sun QFS and Sun Storage Archive Manager Reference Manual\*](#).

# StorageTek ACSLS-Attached Automated Libraries: Common Problems and Error Messages

If errors exist in the configuration files for a StorageTek ACSLS-attached automated library, the system generates several error messages. These examples show common problems and the messages that the system generates.

## StorageTek ACSLS Parameters File Errors

```
May 23 09:26:13 baggins stk-50[3854]: initialize:
Syntax error in stk configuration file line 4.
May 23 09:26:13 baggins stk-50[3854]: initialize:
Syntax error in stk configuration file line 5.
```

Check your StorageTek parameters file for syntax errors. Ensure that each line begins with a keyword or a comment. For more information about the StorageTek parameters file, see “[stk\(7\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## StorageTek ACSLS Library Errors

The following example shows drives that are frozen in the initializing state.

```
May 23 09:29:48 baggins stk-50[3854]: main: Waiting for 2 drive(s) to initialize
May 23 09:29:59 baggins stk-50[3854]: main: Waiting for 2 drive(s) to initialize
May 23 09:30:39 baggins stk-50[3854]: main: Waiting for 2 drive(s) to initialize
.
.
.
May 23 09:31:19 baggins stk-50[3854]: main: 2 drive(s) did not initialize.
```

The following example shows the `samu` utility's `:r` output.

```
ty eq status act use state vsn
sg 51 -----p 0 0% off drive set off due to ACS reported state
sg 52 -----p 0 0% off drive set off due to ACS reported state
lt 61 -----p 0 0% off drive set off due to ACS reported state
tp 62 ----- 0 0% off empty
```

A drive that is frozen in an initializing state or that does not initialize indicates a configuration error.

- Verify that the ACSLS software is running.
- Verify the host name.
- Use the `ping` command to determine whether you can reach the host.
- Check the `portnum` specification in the StorageTek parameters file. In ACSLS 5.3, for example, the default port number, 50004, is used for a different application. Try a higher port number, such as 50014.

## VSN Import Errors

This example shows messages that are generated after the `samimport` command was used to import a VSN to the library catalog, but the VSN was not in the StorageTek automated library. The cartridge must be present in the ACSLS-managed automated library before the `samimport` command can be successful.

```
May 20 15:09:33 baggins stk-50[6117]: view_media  
returned:STATUS_VOLUME_NOT_IN_LIBRARY  
May 20 15:09:33 baggins stk-50[6117]: add_to_cat_req: view_media:  
failed:STATUS_VOLUME_NOT_IN_LIBRARY. A
```

The `sam-stkd` daemon uses the `ssi.sh` script to ensure that a copy of the SSI daemon, `ssi_so`, is running. If `ssi_so` exits, the daemon starts another. If your site has its own version of `ssi.sh`, modify it to wait for a `SIGTERM` signal and then exit. The daemon sends a `SIGTERM` signal to stop the process. The `/opt/SUNWsamfs/examples/ssi.sh` file contains an example `ssi.sh` script. The system copies the `ssi.sh` script to `/etc/opt/SUNWsamfs/scripts/ssi.sh` during installation if one does not already exist.



# Managing Automated Libraries and Manually Loaded Drives

---

An *automated library* is a robotically controlled device that is designed to load and unload removable cartridges without operator intervention. Automated libraries are also known as media changers, jukeboxes, robots, libraries, or media libraries.

This section describes aspects of using automated libraries and manually loaded drives in a SAM-QFS environment. In addition, this section describes the operator-oriented load notification facility that alerts an operator when a requested volume is not in a library.

---

**Note** – The SAM-QFS software interoperates with automated libraries from many manufacturers. Contact Oracle Support for information pertinent to library model numbers, firmware levels, and other compatibility information.

---

## Vendor-Specific Library Operational Procedures

Certain operations for some automated libraries may differ from those described in this section. To determine whether your automated library has additional vendor-specific operating instructions in a SAM-QFS environment, see [Chapter 8, “Managing Vendor-Specific Libraries.”](#)

### ▼ How to Start Removable Media Operations

Typically, removable media operations commence when a file system is mounted.

- **Start removable media operations manually without mounting any file systems.**

```
# samd start
```

If removable media operations are already running when you run the `samd start` command, you will see the following message:

```
SAM-FS sam-amld daemon already running
```

## ▼ How to Stop Removable Media Operations

You can stop removable media operations and leave the file system mounted. You might do this, for example, if you want to manipulate cartridges manually.

### 1 Issue the `idle` command to enable the archiver, stager, and other processes to complete current tasks.

You can also idle drives by using the `samu` operator utility or by using SAM-QFS Manager

```
# samcmd aridle
# samcmd stidle
```

---

**Note** – Failing to follow these procedures could result in tape media issues.

Issue the `sam unload` command for each idle tape drive. If you attempt to restart an idle drive without unloading it, you might experience unpredictable events when archiving, staging, and other activities are resumed.

---

### 2 Monitor the tape drive activity with the `samcmd r` command.

#### a. Wait until all the tape drives have stopped.

#### b. Unload the tape drives.

```
# samcmd unload eq
```

*eq* specifies the equipment number of the drive being unloaded as defined in the `mc f` file.

To unload multiple drives, issue a `samcmd unload` for each drive.

#### c. When the drive is empty, stop removable media operations by running the `sam stop` command.

When you restart operations, pending stages are reissued and archiving is resumed.

## ▼ How to Turn On an Automated Library

A library in the on state is under the control of the SAM-QFS system and can proceed with general operations. When you turn on a library, the SAM-QFS software performs the following actions:

- Queries the device about its state, for example where tapes are, whether barcodes are used, and so on.
- Updates the catalog and other internal structures.

### ● Start an automated library.

```
# samcmd on eq
```

*eq* specifies the equipment number of the automated library as defined in the *mcf* file.

You can also perform this task by using *samu* or SAM-QFS Manager.

## ▼ How to Turn Off an Automated Library

Placing a library in the *off* state stops I/O operations and removes the automated library from SAM-QFS control. No automatic movement of cartridges occurs, and the drives in the automated library remain in the *on* state. You turn off an automated library to perform the following tasks:

- Stop SAM-QFS operations for this automated library only.
- Power down the automated library.

### ● Turn off an automated library.

```
# samcmd off eq
```

*eq* specifies the equipment number of the automated library being addressed as defined in the *mcf* file.

You can also perform this task by using *samu* or SAM-QFS Manager.

## ▼ How to Load a Cartridge Manually

When a cartridge is loaded, it moves from a storage slot to a drive and it is made ready to receive data. A cartridge is loaded automatically when a volume serial name (VSN) is requested for archiving or staging. You can load a cartridge at any time by issuing one of the following commands. You might do this, for example, during a disaster recovery operation or to analyze a tape.

```
1 # samcmd load eq:slot[:partition]
```

```
2 # samcmd load media-type.vsn
```

Argument	Definition
<i>eq</i>	Specifies the equipment number of the drive as defined in the <i>mcf</i> file.
<i>slot</i>	Specifies the number of a storage slot as recognized in the library catalog.
<i>media-type</i>	Specifies the media type. For a list of valid media types, see “ <a href="#">mcf(4)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>partition</i>	Specifies a side of a magneto-optical disc. The partition must be 1 or 2. This argument is not applicable to tape cartridges.

Argument	Definition
<i>vsn</i>	Specifies the volume serial name assigned to the volume.

You can also perform this task by using `samu` or SAM-QFS Manager.

When you load a cartridge manually, it is loaded in the next available drive in the library. If you want to make a drive unavailable for this purpose, use the `samu` utility's `:unavail` command or change the state of the device using SAM-QFS Manager.

---

**Note** – SAM-QFS does not support mixed media in direct attached libraries. If the library is partitioned, each partition must contain only one media type.

---

## ▼ How to Unload a Cartridge Manually

When a cartridge is unloaded, it is removed from a drive. Unloading occurs automatically when a volume is no longer needed. Unload a drive at any time, even if the drive is in `unavail` status.

- **#samcmd unload *eq***

*eq* specifies the equipment number of the drive as defined in the `mc f` file.

You can also perform this task by using `samu` or SAM-QFS Manager.

## Labeling and Unlabeling Cartridges

If you have stand-alone tape or optical devices or if your automated library has no barcode reader, you must label cartridges as described in this section. To label cartridges, use the `tplabel` command for tapes or use the `odlabel` command for optical discs. These commands create a cartridge label that the SAM-QFS software can read.

If your library uses barcodes, `labels = barcodes` is set by default. The result is that the first six characters are used for the volume serial number (VSN).

If your library uses barcodes and you want the last six characters to become the VSN for the cartridge, edit the `/etc/opt/SUNWsamfs/defaults.conf` file and include the following line:

```
labels = barcodes_low
```

When the software loads a barcoded cartridge for a write operation, it writes a label on the cartridge before the write begins. The cartridge must be write enabled, be unlabeled, and have a readable barcode.

## ▼ How to Label or Relabel a Tape



**Caution** – Labeling and relabeling a cartridge makes the data currently on the cartridge inaccessible to any software. Relabel a cartridge only if you are certain that you do not need the data that is stored on the cartridge.

- **Label the tape.**
  - **To label a new tape, use the following `tpLabel` command.**  
`# tpLabel -new -vsn vsn eq:slot`
  - **To relabel an existing tape, use the following `tpLabel` command.**  
`# tpLabel -old vsn -vsn vsn eq:slot`

Argument	Definition
<i>vsn</i>	Specifies the volume serial name. If you are relabeling, the new VSN can be identical to the old VSN.
<i>eq</i>	Specifies the equipment number of the drive as defined in the <code>mcf</code> file.
<i>slot</i>	Specifies the number of the tape's storage slot as recognized in the library catalog. This argument is not applicable for manually loaded drives.

After the command to label or relabel a tape is issued, the tape is loaded and positioned, and the tape label is written.

You can also perform this task by using SAM-QFS Manager.

### Example 7-1 Label a Tape

```
# tpLabel -vsn TAPE01 -new 50:0
```

## ▼ How to Label or Relabel an Optical Disc

- **Label the disc.**
  - **To label a new optical disc, use the following `odLabel` command.**  
`# odLabel -new -vsn vsn eq:slot:partition`
  - **To relabel an existing optical disc, use the following `odLabel` command.**  
`# odLabel -old vsn -vsn vsn eq:slot:partition`

Argument	Definition
<i>vsn</i>	Specifies the volume serial name. If you are relabeling, the new VSN can be identical to the old VSN.
<i>eq</i>	Specifies the equipment number of the drive as defined in the <i>mcf</i> file.
<i>slot</i>	Specifies the number of the disk's storage slot as recognized in the library catalog. This argument is not applicable to manually loaded drives.
<i>partition</i>	Specifies a side of a magneto-optical disc. The partition value must be 1 or 2. This argument is not applicable to tape cartridges.

After the command to label or relabel an optical disc is issued, the optical disc is loaded and positioned, and the optical disc label is written.

You can also perform this task by using SAM-QFS Manager.

### Example 7-2 Label an Optical Disc

```
# odlabel -vsn OPTIC01 -new 30:1:1
```

## ▼ How to Audit a Volume

Occasionally, the library catalog needs to be updated with the reported space remaining on a tape or optical disc. The `auditslot` command loads the cartridge that contains the volume, reads the label, and updates the library catalog entry for the slot.

- Use the following command to update the amount of remaining space:

```
# auditslot [-e] eq:slot[:partition]
```

Argument	Definition
<code>-e</code>	Updates the remaining space if the media is tape. Otherwise, it is not changed.
<i>eq</i>	Specifies the equipment number of the drive as defined in the <i>mcf</i> file.
<i>slot</i>	Specifies the number of the storage slot as recognized in the library catalog. This argument is not applicable to manually loaded drives.
<i>partition</i>	Specifies a side of a magneto-optical disc. The partition value must be 1 or 2. This argument is not applicable to tape cartridges.

For more information, see “`auditslot(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

You can also perform this task by using the `samu` utility's `:audit` command or by using SAM-QFS Manager.

## ▼ How to Audit a Direct Attached Automated Library

A full audit loads each cartridge into a drive, reads the label, and updates the library catalog. Audit a library in the following situations:

- After moving cartridges in the automated library without using SAM-QFS commands
  - If you are in doubt about the status of the library catalog (for example, after a power outage)
  - If you have added, removed, or moved cartridges in an automated library that has no mailbox
- **Perform a full audit on a direct attached automated library.**

```
# samcmd audit eq
```

*eq* specifies the equipment number of the automated library as defined in the *mc f* file.

You can also perform this task by using the *samu* utility's `:audit` command or by using SAM-QFS Manager.

## Using a Cleaning Cartridge

The SAM-QFS environment supports the use of cleaning tapes if cleaning tapes are supported by the hardware. If a tape drive requests a cleaning, the system automatically loads a cleaning tape.

If your system uses barcoded labels, cleaning tapes must have a VSN of CLEAN or a VSN starting with the letters CLN in the barcode label. Alternatively, you can use the `chmed` command to mark a VSN as a cleaning tape and set the count. Multiple cleaning tapes are allowed in a system.

Cleaning practices differ from manufacturer to manufacturer. See [Chapter 8, “Managing Vendor-Specific Libraries,”](#) to determine whether specialized procedures are recommended for your equipment.

## ▼ How to Use a Cleaning Cartridge With a Barcode

If the cleaning cartridge is barcoded, you can import it by using the `samimport` command. This command moves the cartridge from the mailbox to a storage slot and updates the library catalog. In addition, the cleaning media flag is set, and the access count is set to the appropriate number of cleaning cycles, based on the media type. Each time the cartridge is used to clean a drive, the access count is decremented.

- 1 **Ensure that the cleaning cartridge has a barcode of CLEAN or starts with the letters CLN.**
- 2 **Import the cleaning cartridge into the automated library.**

```
# samimport eq
```

*eq* specifies the equipment number of the automated library as defined in the *mc f* file.

You can also perform this task by using *samu* or SAM-QFS Manager.

## ▼ How to Use a Cleaning Cartridge Without a Barcode

If the cartridge is not barcoded, it is not identified as a cleaning cartridge. After you import the cartridge, you must identify it.

### 1 Import the cartridge into the automated library.

```
# samimport eq
```

*eq* specifies the equipment number of the automated library as defined in the *mc f* file.

### 2 Change the type to a cleaning cartridge.

```
# chmed +C eq:slot
```

*eq* specifies the equipment number of the automated library and *slot* specifies the slot in which the cleaning cartridge has been loaded.

In the following example, the automated library is equipment number 50 and the cleaning cartridge is in slot 77:

```
# chmed +C 50:77
```

### 3 Set the cleaning cycle count.

```
# chmed -count count-number eq:slot
```

*eq* specifies the equipment number of the automated library and *slot* specifies the slot in which the cleaning cartridge has been loaded.

The following example command sets the cleaning count on the cartridge to 20.

```
# chmed -count 20 50:77
```

## ▼ How to Reset the Number of Cleaning Cycles

Cleaning cartridges are useful for a limited number of cleaning cycles. The SAM-QFS system ejects the cartridge when the number of remaining cycles equals zero. Each time a cleaning tape is imported, the cleaning cycle is reset to the highest number of cycles for that type of tape. For example, a DLT cleaning tape has 20 cycles and an Exabyte cleaning tape has 10 cycles. You can view the number of remaining cycles with the *samu* utility's *:v* display or by using SAM-QFS Manager.

If automatic cleaning is available but all cleaning cartridges in the automated library have a cleaning cycle count of zero, the drive state is set to *off* and a message is issued in the SAM-QFS log.

- **Reset the cycles on a cleaning tape.**

```
# chmed -count count media-type.vsn
```

Argument	Definition
<i>count</i>	Specifies the number of cleaning cycles to which you want the cleaning tape reset.
<i>media-type</i>	Specifies the media type. For a list of valid media types, see “mcf(4)” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>vsn</i>	Specifies the volume serial name (VSN) assigned to the volume.

## ▼ How to Limit the Number of Cleaning Cycles

Certain drive errors can result in the repeated loading of cleaning cartridges until all cleaning cycles are consumed.

- **Limit the number of cleaning cycles on cleaning cartridges.**

```
# chmed -count count-number eq:slot
```

*eq* specifies the equipment number of the automated library, and *slot* specifies the slot in which the cleaning cartridge has been loaded.

### Example 7-3 Example

For example, the following command sets the cleaning cycle count to 20 for the cleaning cartridge in slot 77 of the automated library with the equipment number of 50.

```
# chmed -count 20 50:77
```

## ▼ How to Clean a Tape Drive Automatically

Beginning with the Sun Storage Archive Manager 4.4 release, the default setting for software-initiated tape drive cleaning is off. You can enable automatic cleaning in one of the following ways:

- Use the media changer's auto-cleaning feature, which might require specific placement of cleaning cartridges. See the manufacturer's documentation for directions.
- Enable the SAM-QFS auto-cleaning feature:

- 1 **Disable the media changer's cleaning feature according to the manufacturer's documentation.**
- 2 **Edit the `defaults.conf` file to add the following line:**

```
tapeclean = all autoclean on logsense on
```

The logsense option prevents a drive from using expired cleaning media. To use only sense data for determining the status of cleaning media, add the following line to the `defaults.conf` file:

```
tapeclean = all autoclean on logsense off
```

---

**Note** – When using the auto-cleaning feature that has a library with more than two drives, use at least two cleaning cartridges for each catalog. If not enough cleaning cartridges are available, any drive that requires cleaning is put into a DOWN state.

---

## ▼ How to Clean a Tape Drive Manually

When automatic cleaning is not available and the system uses barcodes, you can request that a drive be cleaned at any time.

- **Clean the tape drive.**

```
# cleandrive eq
```

*eq* specifies the equipment number of the automated library as defined in the `mcf` file.

This drive is loaded with the cleaning cartridge.

## ▼ How to Clear Media Errors




---

**Caution** – Removing the error flag can cause problems. If you are uncertain about what caused the error and whether the flag can be removed safely, do not use this procedure. Contact Oracle Technical Support.

---

When a hardware or software error is encountered on a cartridge, the SAM-QFS system sets the media error flag in the VSN catalog. The media error flag is shown in the `samu` utility's `v` display and in SAM-QFS Manager.

You can clear the error to reset the flag and you can then attempt to use the cartridge.

- 1 **Clear the media error flag on a cartridge.**

```
# chmed -E media-type.vsn
```

Argument	Definition
<i>media-type</i>	Specifies the media type. For a list of valid media types, see “ <code>mcf(4)</code> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .

Argument	Definition
<i>vsn</i>	Specifies the volume serial name (VSN) assigned to the volume.

## 2 Update the library catalog with the space remaining information.

```
# auditslot -e eq:slot[:partition]
```

Argument	Definition
-e	If the -e option is specified and the media is tape, the remaining space is updated. Otherwise, it is not changed.
<i>eq</i>	The equipment number of the automated library or manually loaded drive as defined in the <i>mc f</i> file.
<i>slot</i>	The number of the storage slot in the automated library as recognized in the library catalog. This argument is not applicable to manually loaded drives.
<i>partition</i>	A side of a magneto-optical disc. The partition must be 1 or 2. This argument is not applicable to tape cartridges.

For more information, see “[auditslot\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

You can also perform this task by using the *samu* utility's `:audit` command or SAM-QFS Manager.

## ▼ How to Remove a Stuck Cartridge From a Drive

### 1 Turn off the drive in the automated library.

```
# samcmd off eq
```

*eq* specifies the equipment number of the drive as defined in the *mc f* file.

You can also perform this step by using *samu* or SAM-QFS Manager.

### 2 Turn off the automated library.

```
# samcmd off eq
```

*eq* specifies the equipment number of the library as defined in the *mc f* file.

You can also perform this step by using *samu* or SAM-QFS Manager.

### 3 Physically remove the cartridge from the drive.

Be careful not to damage either the cartridge or the drive.

**4 Turn on the automated library and the drive.**

Issue this command once for the drive and once for the library.

```
# samcmd on eq
```

*eq* specifies the equipment number of the library and then of the drive as defined in the *mc f* file.

If the automated library performs an audit when it is turned on, you are done.

**5 Follow these steps if the automated library does not perform an audit:****a. Put the cartridge back into its storage slot.****b. Use the *chmed* command to adjust the library catalog to set the occupied flag for the damaged tape.**

```
# chmed +o eq:slot
```

Argument	Definition
<i>eq</i>	Specifies the equipment number of the automated library or drive as defined in the <i>mc f</i> file.
<i>slot</i>	Specifies the number of the storage slot in the library as recognized in the library catalog. This argument is not applicable for manually loaded drives.

If you keep the cartridge out of its slot and you want to put it back in later, you must import the cartridge into the automated library.

## Catalog Operations, Importing Cartridges, and Exporting Cartridges

The physical addition (import) of cartridges to and removal (export) of cartridges from an automated library enables you to perform several functions, including the following:

- Replacing cartridges.
- Relocating cartridges to offsite storage to use later for disaster recovery purposes.  
Use the *chmed -I* command to specify additional information, such as the storage location of the cartridge.

When you import and export cartridges, you also update the library catalog.

The library catalog is the central repository of all information that the SAM-QFS environment needs for finding cartridges in an automated library. The library catalog file is a binary UNIX file system (UFS)-resident file. The information in this file includes the following:

- One or more VSNs that are associated with the cartridge stored in the slot

- The capacity and space that remains on that cartridge
- Flags that indicate read-only, write-protect, recycling, and other status information for the cartridge

The SAM-QFS environment treats catalogs based on how the automated library is attached to the server, as follows:

- If the automated library is direct-attached, the library catalog is a one-to-one mapping between library catalog entries and physical slots in the automated library. The first entry in the library catalog is for the first slot in the automated library. When a cartridge is needed, the system consults the library catalog to determine which slot contains the VSN. Then, the system issues a command to load the cartridge from that slot into a drive.

---

**Note** – SAM-QFS does not support mixed media in direct-attached libraries. If the library is partitioned, each partition must contain only one media type.

---

- If the automated library is network-attached, the library catalog is not a direct mapping to the slots. Instead, it is a list of the VSNs that are known to be present in the automated library. When a cartridge is requested, the system sends a request to the vendor's software to load the VSN into a drive. The vendor's software locates the VSN's storage slot.

Each automated library handles cartridge import and export based to system characteristics and the vendor-supplied software. For example, on the ACL 4/52 library, you must issue a move command to move cartridges into the import or export unit before you export cartridges from the automated library.

Network-attached automated libraries import and export cartridges using their own utilities. So, the `samimport` and `samexport` commands only update the library catalog entries that are used by the SAM-QFS systems. For information about importing and exporting cartridges on a network-attached library, see [Chapter 8, “Managing Vendor-Specific Libraries.”](#)

## Tracking Exported Media - The Historian

The SAM-QFS historian tracks the cartridges that are exported from an automated library or a manually mounted device. The historian acts like a virtual library, but it has no defined hardware devices. It is similar to an automated library in the following ways:

- Is configured in the `mc f` file
  - Use a device type of `hy`. If you do not configure the historian in the `mc f` file, it is created as follows:

```
historian n+1 hy - on /var/opt/SUNWsamfs/catalog/historian
```

In the preceding entry, `n+1` is the last equipment number in the `mc f` file plus 1. To use a different equipment number or path name for the catalog, define the historian in the `mc f`.

- Has a catalog that records entries for all cartridges that are associated with it

The historian library catalog is initialized with 32 entries when the historian first starts. Ensure that the catalog resides on a file system that is large enough to hold the entire catalog. Your site might want to track existing SAM-QFS cartridges that have been exported from the library. In this case, you must build a historian catalog from the existing cartridges as described in “[build\\_cat\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

The following configuration directives in the `defaults.conf` file affect the behavior of the historian:

- `exported_media = unavailable` flags any cartridges that are exported from an automated library as unavailable to the historian. Requests for these cartridges generate EIO errors.
- `attended = no` declares to the historian that no operator is available to handle load requests. Requests to load cartridges that are not already loaded generate EIO errors.  
For more configuration information, see “[historian\(7\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* and “[defaults.conf\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.
- Can import and export cartridges  
Importing and exporting practices differ from manufacturer to manufacturer. To determine if specialized procedures are recommended for your equipment, see [Chapter 8, “Managing Vendor-Specific Libraries.”](#)
- Is shown in SAM-QFS Manager as another automated library

## About Importing and Exporting From an Automated Library

A *mailbox* is an area in an automated library for adding and removing cartridges from the automated library. The `samimport` command moves a cartridge from the mailbox to a storage slot. The `samexport` command moves the cartridge from a storage slot to the mailbox. For most libraries, when a cartridge is present in the mailbox at SAM-QFS software startup, the software imports the cartridge automatically.

## ▼ How to Import a Cartridge From a Library With a Mailbox

### 1 Open the mailbox.

Use the manufacturer's suggested operation, which is usually a button near the mailbox. Sometimes the mailbox is a one-slot mailbox referred to as a *mail slot* in the vendor's documentation.

### 2 Manually place the cartridge in the mailbox.

### 3 Close the mailbox.

### 4 Import the cartridge.

```
# samimport eq
```

*eq* specifies the equipment number of the library as defined in the *mcf* file.

The system moves the cartridge from the mailbox to a storage slot and updates the library catalog for the cartridge.

You can also perform this task by using the *samu* command or SAM-QFS Manager.

## ▼ How to Export a Cartridge From a Library With a Mailbox

### ● Move a cartridge from a storage slot to the mailbox.

Use one of the following formats:

- # samexport *eq:slot*
- # samexport *media-type.vsn*

Argument	Definition
<i>eq</i>	Specifies the equipment number of the automated library as defined in the <i>mcf</i> file.
<i>slot</i>	Specifies the number of the storage slot in the automated library as recognized in the library catalog.
<i>media-type</i>	Specifies the media type of the cartridge. For a list of valid media types, see “ <i>mcf(4)</i> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>vsn</i>	Specifies the volume serial name (VSN) assigned to the volume.

You can also perform this step by using *samu* or SAM-QFS Manager.

## ▼ How to Import a Cartridge From a Library Without a Mailbox

### 1 Unload the cartridge.

```
# samcmd unLoad eq
```

*eq* specifies the equipment number of the library as defined in the *mc f* file. Wait until the system completes its current task, sets the status to *off*, and transfers the current active catalog to the historian.

### 2 Unlock and open the door to the automated library.

### 3 Load cartridges into the available slots.

### 4 Close and lock the door to the automated library.

The automated library reinitializes and scans the cartridges in the library. The SAM-QFS software updates the library catalog by adding the VSNs of the imported cartridges to the catalog. The automated library state is set to *on*.

## ▼ How to Export a Cartridge From a Library Without a Mailbox

### 1 Use the following command:

```
# samcmd unLoad eq
```

*eq* specifies the equipment number of the library being addressed as defined in the *mc f* file. Wait until the system completes its current task, sets the status to *off*, and transfers the current active catalog to the historian.

### 2 Unlock and open the door to the automated library.

### 3 Remove the cartridges from their respective slots.

### 4 Close and lock the door to the automated library.

The automated library reinitializes and scans the cartridges in the automated library. The system updates the library catalog with the VSNs of the cartridges that are currently in library slots. The VSNs of the removed cartridges are removed from the library catalog and are now recorded only in the historian file. The automated library state is set to *on*.

## ▼ How to Enable Load Notification

The SAM-QFS software requests cartridges to be loaded regularly to satisfy archiving and staging needs. If the request is for a cartridge that resides inside a library, the request is handled automatically. If the request is for a cartridge that resides outside the library, operator action is required. If enabled, the `load_notify.sh` script sends email when a cartridge must be obtained from outside the library.

- 1 **Become superuser.**
- 2 **Copy the load notification script from its installed location to its operable location.**

For example:

```
# cp /opt/SUNWsamfs/examples/load_notify.sh
  /etc/opt/SUNWsamfs/scripts/load_notify.sh
```

- 3 **Examine the `defaults.conf` file.**

Ensure that the following default directives are in the file and have not been changed.

```
exported_media=available
attended=yes
```

- 4 **Modify the `load_notify.sh` script to send notices to the operator.**

By default, the script sends email to root, but it can be edited to send email to another person, to dial a pager, or to provide some other means of notification.

## Using Drives With Encryption Capability

If you are archiving files to drives with encryption capability, plan your archive operations according to the following considerations:

- Do not mix non-encrypted and encryption-capable drives in a library.
- After a drive has encryption enabled, it cannot be disabled.
- Do not mix encrypted and non-encrypted files on a tape.
- An encrypted drive cannot append to a tape that contains non-encrypted data.
- An encryption-enabled drive can read non-encrypted data.

## Manually Loaded Drive Operations

This section describes operations that differ if you have a manually loaded, stand-alone drive rather than an automated library. Each manually loaded drive has its own one-slot library catalog.

## ▼ How to Load a Cartridge Into a Manually Loaded Device

- Place the cartridge in the drive according to the manufacturer's instructions.

The SAM-QFS system recognizes that the cartridge is loaded, reads the label, and updates the one-slot catalog. No further action is necessary.

## ▼ How to Unload a Cartridge

- Idle the drive.

```
# samcmd idle eq
```

*eq* specifies the equipment number of the drive as defined in the *mc f* file.

This command ensures that no archive or stage processes are active. The drive switches from idle to off when all I/O activity is complete, and the tape ejects.

If the cartridge is a tape, the tape rewinds and is ready to be removed. An optical disc ejects automatically. See the manufacturer's instructions for removing the specific cartridge. You can also perform this task by using *samu* or SAM-QFS Manager.

## ▼ How To View a Library Catalog

- View the library catalog by using the *samu* command.

```
# samu :v eq
```

*eq* specifies the equipment number of the library as defined in the *mc f* file.

## Managing Vendor-Specific Libraries

---

You can include libraries from many different manufacturers in a SAM-QFS environment. For most libraries, use the operational procedures that are described in [Chapter 7, “Managing Automated Libraries and Manually Loaded Drives.”](#) The following libraries, however, have vendor-specific operational procedures:

- “StorageTek ACSLS-Attached Automated Libraries” on page 79
- “ADIC/Grau Automated Libraries” on page 81
- “IBM 3584 UltraScalable Tape Libraries” on page 82
- “IBM 3494 Libraries” on page 84
- “Sony Direct-Attached 8400 PetaSite Automated Libraries” on page 85
- “Sony Network-Attached Automated Libraries” on page 87

---

**Note** – Consult your Oracle sales representative or your authorized service provider for information about library model numbers, firmware levels, and other compatibility information.

---

### StorageTek ACSLS-Attached Automated Libraries

The following sections describe aspects of this library's operations that differ from the procedures described in [Chapter 7, “Managing Automated Libraries and Manually Loaded Drives.”](#)

Some StorageTek automated libraries, such as the StorageTek 9730 library use a mail slot to import and export only one cartridge at a time. A mailbox is an area used for putting cartridges into and removing cartridges from the automated library. Examples of StorageTek automated libraries that have a mailbox are the StorageTek 9714 and the StorageTek 9710 libraries.

In StorageTek documentation, the mailbox and mailslot are both referred to as the *cartridge access port* (CAP).

When importing and exporting cartridges from any ACSLS-attached automated library, you must keep the ACSLS inventory and the SAM-QFS catalog in agreement.

- When importing cartridges, the `samimport` command does not insert cartridges physically into the automated library. You must also issue ACSLS commands to complete the operation.
- When exporting cartridges, issue the `samexport -f` command to direct the SAM-QFS system to put the cartridge in the CAP and to update the catalog. Without the `-f` option, the cartridge is not in the CAP so you must then use ACSLS commands to complete the operation.

You can also perform the import and export procedures by using `samu` or SAM-QFS Manager. For more information, see the SAM-QFS Manager online help.

## ▼ How to Import Tapes

### ● Import the tape.

```
# samimport -v vsn eq
```

Argument	Definition
<i>vsn</i>	Specifies the volume serial name (VSN) assigned to the volume.
<i>eq</i>	Specifies the equipment ordinal of the device being addressed as defined in the <code>mcf</code> file.

The `samimport` command causes the new VSN to appear in the library catalog. If the VSN was in the historian, the SAM-QFS software moves the VSN information from the historian to the library catalog.

## ▼ How to Export Tapes Using a Mailbox

### ● You can export tape cartridges by using slot or VSN information.

Export the tape using one of the following formats:

- `samexport [-f] eq:slot`
- `samexport [-f] media-type.vsn`

Argument	Definition
<code>-f</code>	Specifies that the SAM-QFS system put the volume in the cartridge access port (CAP) and update the catalog accordingly.
<i>eq</i>	Specifies the equipment ordinal of the device being addressed as defined in the <code>mcf</code> file.

Argument	Definition
<i>slot</i>	Specifies the number of a storage slot in an automated library as recognized in the library catalog.
<i>media-type</i>	Specifies the media type. For a list of valid media types, see “mcf(4)” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>vsn</i>	Specifies the volume serial name (VSN) assigned to the volume.

## ADIC/Grau Automated Libraries

---

**Note** – ADIC/Grau network-attached libraries are not supported by SAM-QFS software on an x64 hardware platform.

---

The following sections describe aspects of this library's operations that differ from the procedures described in [Chapter 7, “Managing Automated Libraries and Manually Loaded Drives.”](#)

Because you use vendor-supplied utilities to physically add and remove cartridges in the ADIC/Grau automated library, the SAM-QFS interfaces (`samimport`, `samexport`, and SAM-QFS Manager) only affect the library catalog.

### ▼ How to Import a Cartridge

- 1 Use ADIC/Grau commands to physically move the cartridge into the library.
- 2 Use the `samimport` command to update the library catalog.

```
# samimport -v volser eq
```

Argument	Definition
<i>volser</i>	Specifies the <i>volser</i> to be added. The <code>grauaci</code> interface verifies that the ADIC/Grau automated library has the <i>volser</i> information before updating the library catalog with the new entry.
<i>eq</i>	Specifies the equipment ordinal of the device being addressed as defined in the <code>mcf</code> file.

## ▼ How to Export a Cartridge

- 1 Use the `samexport` command to remove the entry from the library catalog.

```
# samexport eq:slot
# samexport media-type.vsn
```

Argument	Definition
<code>eq</code>	Specifies the equipment ordinal of the device being addressed as defined in the <code>mcf</code> file.
<code>slot</code>	Specifies the number of a storage slot in an automated library as recognized in the library catalog.
<code>media-type</code>	Specifies the media type. For a list of valid media types, see “ <code>mcf(4)</code> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<code>vsn</code>	The volume serial name (VSN) assigned to the volume.

The `samexport` command updates the library catalog as each VSN is exported. It also moves the library catalog entry for each VSN from the library catalog to the historian.

- 2 Use `ADIC/Grau` commands to physically move the cartridge out of the library.

## IBM 3584 UltraScalable Tape Libraries

---

**Note** – IBM 3584 UltraScalable libraries are not supported by the SAM-QFS software on an x64 hardware platform.

---

The following sections describe aspects of this library’s operations that differ from the procedures described in [Chapter 7, “Managing Automated Libraries and Manually Loaded Drives.”](#)

### Importing Cartridges

When the SAM-QFS software is started, cartridges that are in the mailbox are not automatically imported.

## Cleaning Drives

Disable automatic cleaning and enable hosted cleaning. This process is described in the *IBM 3584 UltraScalable Tape Library Planning and Operator Guide*, IBM publication GA32-0408-01. This process is also described in “[ibm3584\(7\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Partitioning

This library accommodates several tape drives. If you use multiple drives, you can divide this one physical library into two, three, or four logical libraries. If you do divide your library into two or more logical libraries, ensure that these logical libraries are operating properly before you add the IBM 3584 library to the SAM-QFS environment.

When a cartridge is exported from a partitioned library, only the logical library from which it was exported can obtain access to that drawer slot. If the cartridge is removed and reinserted manually, any logical library can obtain access to that drawer slot.

For more information about using this library as a logically partitioned library in a SAM-QFS environment, see your IBM documentation or “[ibm3584\(7\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

### ▼ How To Remove a Cartridge From a Logical Library

- 1 Open the door.
- 2 Remove the cartridge.
- 3 Close the door.
- 4 Wait for the door to lock and then unlock.
- 5 Open the door.
- 6 Replace the cartridge.
- 7 Close the door.

# IBM 3494 Libraries

---

**Note** – IBM 3494 network-attached libraries are not supported by the SAM-QFS software on an x64 hardware platform.

---

The following sections describe aspects of this library's operations that differ from the procedures described in [Chapter 7, “Managing Automated Libraries and Manually Loaded Drives.”](#)

## ▼ How to Import a Cartridge

**1 Place the new media into the I/O slots.**

**2 Close the door.**

The library locks the door and moves the media into the storage area. You can import only 100 volumes at one time.

- **The library locks the door and moves the media into the storage area. You can import only 100 volumes at one time.**
- **If the library is configured with `access=shared`, issue the `samimport` command to add the media to the catalog.**

## ▼ How to Export a Cartridge

**1 Use the `export` command to export cartridges.**

This command moves the media to the I/O area and turns on the output mode light on the operator panel.

**2 Physically remove the media from the I/O area.**

---

# Sony Direct-Attached 8400 PetaSite Automated Libraries

---

**Note** – Sony 8400 PetaSite libraries are not supported by the SAM-QFS software on an x64 hardware platform.

---

The Sony 8400 PetaSite Series automated library is different from other Sony models because it has an eight-slot import and export mailbox (slots 400-407). Because the mailbox slots can be used as storage slots, the SAM-QFS library catalog tracks the mailbox slots. This automated library uses a barcode reader.

---

**Note** – The information in this section applies only to Sony direct-attached 8400 PetaSite automated libraries. This information does not pertain to the Sony direct-attached B9 and B35 automated libraries, nor does it pertain to the [“Sony Network-Attached Automated Libraries” on page 87](#).

---

## ▼ How to Import Tapes

- 1 **Open the door of the automated library by pushing the Open/Close button on the front panel of the automated library.**
- 2 **Load the cartridges into the mailbox slots.**
- 3 **Push the Open/Close button on the front panel of the automated library.**
- 4 **Manually close the door to the mailbox.**

The automated library checks the mailbox slots for the cartridge barcodes. If the library detects a problem with a barcode, both the in and out lights flash for that slot.

- 5 **Use the `samimport` command to enable the SAM-QFS system to recognize the imported cartridges.**

`# samimport eq`

`eq` specifies the equipment ordinal of the device being addressed as defined in the `mcf` file.

You can also perform this step by using SAM-QFS Manager. For more information, see SAM-QFS Manager online help.

## ▼ How to Export a Tape Without Using Mailbox Slots as Storage Slots

- 1 Move the cartridge to a mailbox slot (slots 400-407).

```
# move source-slot destination-slot eq
```

Argument	Definition
<i>source-slot</i>	Specifies the number of the slot in which the cartridge currently resides.
<i>destination-slot</i>	Specifies the number of the slot into which the cartridge should be moved.
<i>eq</i>	Specifies the equipment ordinal of the device being addressed as defined in the mc f file.

- 2 Push the Open/Close button on the front panel of the automated library to open the door.
- 3 Remove the cartridge from the mailbox slot.
- 4 Push the Open/Close button on the front panel of the automated library.
- 5 Manually close the door to the mailbox.
- 6 Issue the `samexport` command to enable the SAM-QFS system to recognize the exported cartridge.

```
# samexport eq
```

*eq* specifies the equipment ordinal of the device being addressed as defined in the mc f file.

You can also perform this step by using SAM-QFS Manager. For more information, see the SAM-QFS Manager online help.

## ▼ How to Export a Tape Using Mailbox Slots as Storage Slots

- 1 Push the Open/Close button on the front panel of the automated library to open the door.
- 2 Remove the cartridge from the mailbox slot.
- 3 Push the Open/Close button on the front panel of the automated library.
- 4 Manually close the mailbox door.

- 5 Issue the `samexport` command to enable the SAM-QFS system to recognize the exported cartridge.

```
# samexport eq
```

`eq` specifies the equipment ordinal of the device being addressed as defined in the `mc f` file.

You can also perform this step by using SAM-QFS Manager. For more information, see the SAM-QFS Manager online help.

## ▼ How to Move a Cartridge to a Different Slot

- 1 Ensure that the source slot is occupied and that the destination slot is empty.
- 2 Issue the `move` command.

```
# move eq:source-slot destination-slot
```

Argument	Definition
<code>eq</code>	Specifies the equipment ordinal of the device being addressed as defined in the <code>mc f</code> file.
<code>source-slot</code>	Specifies the number of the slot in which the cartridge currently resides.
<code>destination-slot</code>	Specifies the number of the slot into which the cartridge should be moved.

You can also perform this step by using SAM-QFS Manager. For more information, see the SAM-QFS Manager online help.

## Sony Network-Attached Automated Libraries

---

**Note** – Sony network-attached libraries are not supported by the SAM-QFS software on an x64 hardware platform.

---

The following sections describe aspects of this library's operations that differ from the procedures described in [Chapter 7, “Managing Automated Libraries and Manually Loaded Drives.”](#)

Because you use vendor-supplied utilities to physically add and remove cartridges, the SAM-QFS interfaces (`samimport`, `samexport`, and SAM-QFS Manager) affect only the library catalog.

## ▼ How to Import a Cartridge

- 1 Use Sony commands to physically move the cartridge into the library.
- 2 Use the `samimport` command to update the library catalog.

```
# samimport -v "volser" eq
```

Argument	Definition
<i>volser</i>	Specifies the <i>volser</i> to be added. The PSC API interface verifies that the Sony automated library has the <i>volser</i> information before updating the library catalog with the new entry. If the cartridge does not physically exist in the library, the entry is placed in the historian catalog.  The <i>volser</i> value must be enclosed in quotation marks if it contains spaces.
<i>eq</i>	Specifies the equipment ordinal of the library being addressed as defined in the <code>mcf</code> file.

## ▼ How to Export a Cartridge

- 1 Remove the entry from the library catalog.

The `samexport` command updates the library catalog as each VSN is exported, and it moves the library catalog entry for each VSN from the library catalog to the historian.

```
# samexport eq:slot
# samexport media-type.vsn
```

Argument	Definition
<i>eq</i>	Specifies the equipment ordinal of the device being addressed as defined in the <code>mcf</code> file.
<i>slot</i>	Specifies the number of a storage slot in an automated library as recognized in the library catalog.
<i>media-type</i>	Specifies the media type. For a list of valid media types, see “ <code>mcf(4)</code> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>vsn</i>	Specifies the volume serial name (VSN) assigned to the volume.

- 2 Use Sony commands to physically move the cartridge out of the library.

## About Archiving

---

Archiving is the process of copying a file from a file system to one of the following:

- A volume that resides on a removable media cartridge
- A disk partition of another file system

When using SAM-QFS, you can specify that files be archived immediately, specify that files never be archived, and perform other tasks.

The term *archive media* refers to the various cartridges or disk slices to which archive volumes are written. This section provides general guidelines for developing the archive policies for your site and describes the archiver's theory of operations. See [Chapter 10, “Configuring the Archiver,”](#) for details about how to configure the archiver.

## Planning Archiving Operations

The archiver automates storage management operations by using information in the `archiver.cmd` file. Before writing this file, review the following general guidelines:

- **Save your archive logs** The archive logs are essential to recovering data even if the SAM software is unavailable. Keep these logs in a safe place in the event of a catastrophic disaster.
- **Use regular expressions for volume ranges** Enable the system to put files on many different volumes. Volume ranges allow the system to run continuously. If you use specific volume names, the archive set copies can fill a volume rapidly, which can cause workflow problems as you replace media.
- **Set an optimal archive interval** The archive interval is the time between file system scans. Set your archive interval based on how often files are created and modified, and whether you want to save all modified copies. An archive interval that is too short keeps the archiver scanning almost continuously.
- **Consider the number of file systems you are using.** Multiple file systems increase the performance of the archiver over a single file system. Multiple file systems can be scanned in less time than a single file system.

- **Use directory structures to organize your files in a file system** To maximize performance, do not place more than 10,000 files in a directory.
- **Always make a minimum of two file copies on two separate volumes** Putting data on a single media type puts your data at risk if physical problems with the media occur.
- **Back up your metadata (directory structure, file names, and so on)** Use `samfsdump` on a regular basis. The metadata is stored in an archive set that has the same name as the file system. You can use this information to recover a file system in the event of a disaster. If you do not want the archiver to back up your metadata, set `archivemeta=off` in the `archiver.cmd` file and schedule the `samfsdump` command to run in a `cron` file.

These guidelines can improve the performance of your file system and the archiver, which ensures that your data is stored in the safest possible way.

## Preview Queue

The archiver and stager processes can both request that media be loaded and unloaded. If the number of requests exceeds the number of drives available for media loads, the excess requests are sent to the preview queue.

By default, preview requests are satisfied in first-in-first-out (FIFO) order. You can override the FIFO default by entering directives in the preview command file, which is written to `/etc/opt/SUNWsamfs/preview.cmd`. For more information about using this file, see [“Prioritizing Preview Requests” on page 179](#).

## Archiver Daemons

SAM-QFS has the following archiver daemon and processes:

- The `sam-archiverd` daemon schedules the archiving activity.
- The `sam-arfind` process assigns files to be archived to archive sets.
- The `sam-arcopy` process copies the files to be archived to the selected volumes.

When SAM-QFS is initiated, its `sam-fsd` daemon starts the `sam-archiverd` daemon.

The `sam-archiverd` daemon executes the `archiver` command to read the `archiver.cmd` file and then builds the tables necessary to control archiving.

The `sam-archiverd` daemon starts a `sam-arfind` process for each mounted file system. When a file system is unmounted, it stops the associated `sam-arfind` process.

The `sam-archiverd` process monitors `sam-arfind` and processes signals from an operator or other processes.

# Archiving Operations

Operator actions are not required to archive files. The archiver writes files to a volume on the archive media. Archive media can contain one or more volumes. Each volume is identified by a unique identifier called a volume serial name (VSN).

By default, the archiver makes one copy of each archive set, but you can request up to four copies. An archive set and a copy number become synonyms for a collection of volumes. The copies of the archive set provide duplication of files on separate volumes.

To be a candidate for archiving or re-archiving, the data in a file must change. A file is not archived if it is accessed but not changed. For example, issuing a `touch` command on a file does not cause it to be archived or re-archived. If the `mv` command is used to rename a file, the file might move to a different archive set. In this case, the archiver software re-evaluates the archive copies to determine whether the file needs to be archived or re-archived.

---

**Note** – Because issuing an `mv` command alters the file name but not the file data, this action can have ramifications for disaster recovery if you plan to restore from tar files.

---

The archive operation is affected by the following factors:

- **Archive age** The period of time that has passed since the file was last modified. The archive age can be defined for each archive copy.

Use the `touch` command to change the default time references on their files to values in the past or future. Note that this practice can cause unexpected archiving results. To avoid problems, the archiver adjusts the references so that they are always at a point between the file creation time and the present time.

- **Archive priority** This value is computed from file property characteristics and from file property multipliers that are associated with the archive set, as follows:

$archive-priority = file-property-value \times property-multiplier$

- Most *file-property-value* numbers are 1 (for true) or 0 (for false). For instance, the value of the property copy 1 is 1 if archive copy 1 is being made. The values of copy 2, copy 3, and copy 4 are therefore 0. Other properties, such as archive age and file size, can have values other than 0 or 1.
- The *property-multiplier* value is determined from the `-priority` parameters for the archive set. Various aspects of a file, such as age or size, can be given values to determine the archive request's priority. The default value for all property multipliers is 0.0. For more information about the `-priority` parameter, see [“archiver.cmd\(4\)” in Sun QFS and Sun Storage Archive Manager Reference Manual](#).

The archive-priority and the property-multiplier values are floating-point numbers.

The following sections describe the steps taken by the archiver from the initial file scan to the file copy process:

- “Step 1: Identifying Files to Archive” on page 92
- “Step 2: Composing Archive Requests” on page 94
- “Step 3: Scheduling Archive Requests” on page 95
- “Step 4: Archiving the Files in an Archive Request” on page 97
- “Sample Default Output From the archiver -l Command” on page 98

## Step 1: Identifying Files to Archive

A separate `sam-archind` process monitors each mounted file system to determine which files need archiving. The file system notifies this `sam-archind` process whenever a file is changed in a manner that affects its archival state. Such changes are file modification, re-archiving, unarchiving, and renaming.

The `sam-archind` process examines the file to determine the archive set to which the file belongs and what action to take.

The `sam-archind` process uses the following file property descriptions to determine a file's archive set:

- The directory path portion of the file's name and, optionally, the complete file name using a regular expression
- The user name of the file's owner
- The group name of the file's owner
- A minimum file size
- A maximum file size

The `sam-archind` process performs the following evaluation to determine the archive action:

- If the archive age of the file for one or more copies has been met or exceeded, the `sam-archind` process adds the file to one or more archive requests for the archive set. An archive request is a collection of files that belong to the same archive set. The archive request resides in the `/var/opt/SUNWsamfs/archiver/file_sys/ArchReq` directory. The files in this directory are binary files. To display them, use the `showqueue` command. Separate archive requests are used for files that are not yet archived and for files that are being re-archived. This enables you to control scheduling independently for these two types of files.
- If the archive age of the file for one or more copies has not been met, the directory in which the file resides and the time at which the archive age is reached is added to a scan list. Directories are scanned as the scan list times are reached. Files that have reached their archive age are added to archive requests.
- If a file is offline, the `sam-archind` process selects the volumes to be used as the source for the archive copy. If the file copy is being re-archived, the `sam-archind` process selects the volume that contains the archive copy that is being rearchived.

- If a file is segmented, only those segments that have changed are selected for archiving. The index of a segmented file contains no user data, so it is treated as a member of the file system archive set and is archived separately.

The archive action is accomplished using the following methods:

- [“Continuous Archiving” on page 93](#)
- [“Scanned Archiving” on page 93](#)

## Continuous Archiving

With the continuous archiving method, the archiver works with the file system to determine which files need to be archived.

Continuous archiving is the default archiving method (the `archiver.cmd` file parameter is `examine=noscan`) and operates with the following default start conditions:

- The archive starts every two hours.
- The archive waits until at least 90% of the `archmax` value of data is ready to archive.
- The archive waits until at least 500,000 files to archive.

When any of the scheduling start conditions is reached, the `sam-archfind` process sends each archive request to the archiver daemon, `sam-archiverd`, to set the schedule for copying the file to archive media.

To control the schedule of the archive operation, specify the start conditions for each archive set by using the `-startage`, `-startcount`, and `-startsize` parameters. These conditions enable you to optimize archive timeliness in relation to archive work done. For example:

- If creating files that you want archived together takes an hour, set the `-startage` parameter to one hour (`-startage 1h`) to ensure that all files are created before the archive request occurs.
- If you want all archive requests to be no less than 150 gigabytes of data, specify a `-startsize` of 150 gigabytes (`-startsize 150g`) to direct the archiver to wait until 150 gigabytes of data are ready to be archived.
- If you know that 3000 files are available for archival, specify `-startcount 3000` to ensure that the files are archived together.

For more information about archiving parameters, see [“Global Archiving Directives” on page 125](#).

## Scanned Archiving

With the scanning method, the archiver checks the file systems periodically and selects files for archiving. To enable the scanning method and to disable continuous scanning, specify `examine=scan` in the `archiver.cmd` file.

The `sam-arfind` process scans each file system periodically to determine which files need archiving and to place them in archive requests. The first scan is a directory scan, in which `sam-arfind` recursively descends through the directory tree. The process examines each file and sets the file status flag to `archdone` if the file does not need archiving. During successive scans, the `sam-arfind` process scans the `.inodes` file and examines only inodes without the `archdone` flag.

When the scan is complete, the `sam-arfind` process sends each archive request to the archiver daemon, `sam-archiverd`, to be scheduled for file copying to archive media. The `sam-arfind` process then sleeps for the duration specified by the `interval=time` directive. At the end of the interval, the `sam-arfind` process resumes scanning.

For information about controlling the setting of the `archdone` flag, see [“setarchdone Directive: Controlling the Setting of the archdone Flag”](#) on page 132.

## Step 2: Composing Archive Requests

Regardless of the archiving method used, the `sam-arfind` process sends each archive request to the archiver daemon, `sam-archiverd` for composing. This section describes the composing process.

Composing is the process of selecting the files from the archive request to be archived at one time. Because of the capacity of the archive media or of the controls specified in the archiver command file, the files in an archive request might not all be archived simultaneously. When the archive copy operation is complete for an archive request, the archive request is recomposed if files remain to be archived.

The `sam-archiverd` daemon places the files in the archive request according to certain default and site-specific criteria. The default operation is to archive all the files in an archive request by their full path name so that files in the same directories are also together on the archive media. The site-specific criteria enable you to control the order in which files are archived and how they can be distributed on volumes. These criteria, called *archive set parameters*, are evaluated in the following order: `-reserve`, `-sort`, `-rsort` (reverse sort), and `-drives`. For more information, see [“archiver.cmd\(4\)”](#) in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

When the `-reserve owner` parameter is specified, the `sam-archiverd` daemon orders the files in the archive request according to the file's directory path, user name, or group name. The files that belong to the *owner* are selected for archiving. The remaining files are archived later.

When the `-sort` or `-rsort` parameter is specified, the `sam-archiverd` daemon orders the files according to the specified sort method such as age, size, or directory location.

When an archive request contains both online and offline files, the online files are selected for archiving first.

In the absence of a specified sort method, the offline files are ordered by the volume on which the archive copies reside. This rule ensures that all files in each archive set on the same volume are staged simultaneously in the order in which they were stored on the media. When more than one archive copy of an offline file is being made, the offline file is not released until all required copies are made. All the files to be staged from the same volume as the first file are selected for archiving.

---

**Note** – Using the `-sort` and `-rsort` parameters can have a negative effect on performance during archiving of offline files. This effect occurs if the order of the files to be archived does not match the order of the volumes that is needed for the offline files. Use these parameters only for the first archive copy to be made. Other copies must maintain the order of the first copy if enough archive media space is available when the copies are started.

---

After being composed, the archive requests are entered in the `sam-archiverd` daemon's scheduling queue.

### Step 3: Scheduling Archive Requests

The scheduler in the `sam-archiverd` daemon executes on demand when one of the following conditions exists:

- An archive request is entered in the scheduling queue.
- The archiving for an archive request has been completed.
- A change in media status is received from the catalog server.
- A message is received that changes the state of the archiver.

The archive requests in the scheduling queue are ordered by priority. Each time the scheduler runs, it examines all archive requests to determine whether they can be assigned to a `sam-arcopy` process to have their files copied to archive media.

The following must be true for archive requests to be scheduled:

- Drive usage must be available for making file copies.
- Volume usage must be available and have sufficient space to hold the files in the archive request.

### Drive Usage

If the archive set has the `-drives` parameter specified, the `sam-archiverd` daemon divides the selected files in the archive request among the multiple drives. If the number of drives available is fewer than that specified by the `-drives` parameter, the smaller number of drives is used.

If the total size of files in the archive request is less than the `-drivemin` value, one drive is used. The `-drivemin` value is either the value specified by the `-drivemin` parameter or the `archmax`

value. The `archmax` value is specified by the `-archmax` parameter or the value defined for the media. For more information about the `-archmax` parameter and the `archmax=` directive, see “[archiver.cmd\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

If the total size of files in the archive request is more than the `-drivemin` value, the number of drives used is determined by the total size of the files divided by the `-drivemin` value. If the number of drives available is fewer than that specified by the `-drives` parameter, the smaller number of drives is used.

Drives can take varying amounts of time to archive files. The `-drivemax` parameter specifies the maximum number of bytes to be written to a drive before that drive is rescheduled for more data. You can use the `-drivemax` parameter to obtain better drive utilization.

## Volume Usage

For archiving to occur, at least one volume must have enough space to hold at least some of the files in the archive request. The volume that has most recently been used for the archive set is the one scheduled, if it has enough space. This volume must not be in use by the archiver.

If a volume usable for the archive set is busy, another is selected unless the `-fillsns` parameter is specified. In this case, the archive request cannot be scheduled.

If an archive request is too big for one volume, the files that can fit on the volume are selected to be archived to the volume. If the archive request contains files that are too big to fit on one volume and volume overflow for the archive request is not selected, the files cannot be archived. An appropriate message for this condition is sent to the log.

You can specify volume overflow for the archive set by using the `-ovflmin` parameter, or for the media by using the `ovflmin=` directive. For more information about the `-ovflmin` parameter and the `ovflmin=` directive, see “[archiver.cmd\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*. The `ovflmin` specification determines the file size threshold above which additional volumes or media are assigned for archiving. An `ovflmin` value specified for the archive set takes precedence over an `ovflmin` value specified for the media.

If the size of the files is less than the value of `ovflmin`, the files cannot be archived. An appropriate message for this condition is sent to the log. If the size of the files is more than the value of `ovflmin`, additional volumes are assigned as required. Volumes are selected in order of decreasing size to minimize the number of volumes required. If no usable volumes can be found for the archive request, the archive request waits until volumes become available.

Certain properties, such as whether the file is online or offline, are used in conjunction with the archive priority to determine the scheduling priority for a particular archive request. For more information about customizing the priority multiplier, see the `-priority` parameters described on the `archiver.cmd(4)` man page.

For each archive request, the `sam-archiverd` daemon computes the scheduling priority by adding the archive priority to multipliers that are associated with various system resource

properties. These properties are associated with the number of seconds for which the archive request has been queued, whether the first volume to be used in the archiving process is loaded into a drive, and so on.

Using the adjusted priorities, the `sam-archiverd` daemon assigns each ready archive request to be copied.

## Step 4: Archiving the Files in an Archive Request

When an archive request is ready to be archived, the `sam-archiverd` daemon marks the archive file boundaries so that each archive file's size is less than the specified `-archmax` value. If a single file is larger than this value, it becomes the only file in an archive file.

For each archive request and each drive to be used, the `sam-archiverd` daemon assigns the archive request to a `sam-arcopy` process to copy the files to the archive media. The archive information is entered into the inode.

If archive logging is enabled, an archive log entry is created.

For each file that was staged, the disk space is released until all files in the list have been archived.

A variety of errors and file status changes can prevent a file from being successfully copied. Errors include read errors from the cache disk and write errors to the volumes. Status changes include modification since selection, a file that is open for writing, or a file that has been removed.

When the `sam-arcopy` process exits, the `sam-archiverd` daemon examines the archive request. If any files have not been archived, the archive request is recomposed.

You can also specify timeouts for archiving operations in the `archiver.cmd` file. The directive is as follows:

```
timeout = [operation | media] time
```

*operation* is one of the following:

- `read` – Read the file from the disk. The default timeout is 1 minute.
- `request` – Request the archive media. The default timeout is 15 minutes.
- `stage` – Stage the file to be archived. The default timeout is 0 minutes.
- `write` – Write to the archive media. The default timeout is 15 minutes.

The `time` value for the write operation can also be specified for individual media.

You can configure the `-queue_time_limit time` archive set parameter in the `archiver.cmd` file to notify the administrator when an archive request has been in the schedule queue longer than a certain amount of time. If the archive request remains in the queue at the end of this time, an email is sent to the system administrator.

## Sample Default Output From the archiver -l Command

The following example shows sample output from the `archiver -l` command.

```
# archiver -l
Archive media:
default:mo
media:mo archmax:5000000
media:lt archmax:50000000
Archive devices:
device:mo20 drives_available:1 archive_drives:1
device:lt30 drives_available:1 archive_drives:1
Archive file selections:
Filesystem samfs1:
samfs1 Metadata
copy:1 arch_age:240
big path:. minsize:512000
copy:1 arch_age:240

all path:
copy:1 arch_age:30
Archive sets:
all
copy:1 media:mo
big
copy:1 media:lt
samfs1
copy:1 media:mo
```

## Log Files and Event Logging for Archive Operations

The log file is a continuous record of archival action. You can use the log file to locate earlier copies of files for traditional backup purposes. The `sam-archfind` and `sam-archcopy` processes use the `syslog` facility and `archiver.sh` to log warnings and informational messages in a log file that contains information about each archived or automatically unarchived file.

The log file is disabled by default. Use the `logfile=` directive in the `archiver.cmd` file to enable logging and to specify the name of the log file. For more information about the log file, see [“About the archiver.cmd File” on page 101](#) and [“archiver.cmd\(4\)” in \*Sun QFS and Sun Storage Archive Manager Reference Manual\*](#).

The following example shows sample lines from an archiver log file.

### EXAMPLE 9-1 Archiver Log File Lines

```
A 2001/03/23 18:42:06 mo 0004A arset0.1 9a089.1329 samfs1 118.51 162514 t0/fdn f 0 56
A 2001/03/23 18:42:10 mo 0004A arset0.1 9aac2.1 samfs1 189.53 1515016 t0/fae f 0 56
A 2001/03/23 18:42:10 mo 0004A arset0.1 9aac2.b92 samfs1 125.53 867101 t0/fai f 0 56
A 2001/03/23 19:13:09 lt SLOT22 arset0.2 798.1 samfs1 71531.14 1841087 t0/fhh f 0 51
```

**EXAMPLE 9-1** Archiver Log File Lines (Continued)

```

A 2001/03/23 19:13:10 lt SLOT22 arset0.2 798.e0e samfs1 71532.12 543390 t0/fhg f 0 51
A 2003/10/23 13:30:24 dk DISK01/d8/d16/f216 arset4.1 810d8.1 qfs2 119571.301 1136048
  t1/fileem f 0 0
A 2003/10/23 13:30:25 dk DISK01/d8/d16/f216 arset4.1 810d8.8ad qfs2 119573.295 1849474
  t1/fileud f 0 0
A 2003/10/23 13:30:25 dk DISK01/d8/d16/f216 arset4.1 810d8.16cb qfs2 119576.301 644930
  t1/fileen f 0 0
A 2003/10/23 13:30:25 dk DISK01/d8/d16/f216 arset4.1 810d8.1bb8 qfs2 119577.301 1322899
  t1/fileeo f 0 0

```

The following table defines each field in the log.

**TABLE 9-1** Archiver Log File Fields

Field	Example Value	Content
1	A	Archive activity, as follows: <ul style="list-style-type: none"> <li>■ A — Archived</li> <li>■ R — Re-archived</li> <li>■ U — Unarchived</li> </ul>
2	2001/03/23	Date of the archive action, in <i>yyyy/mm/dd</i> format.
3	18:42:06	Time of the archive activity, in <i>hh:mm:ss</i> format.
4	mo	Archive media type. For information about media types, see “ <a href="#">mcf(4)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
5	0004A	VSN. For removable media cartridges, the volume serial name. For disk archives, the disk volume name and archive <code>tar(1)</code> file path.
6	arset0.1	Archive set and copy number.
7	9a089.1329	Physical position of the start of the archive file on media (tar file) and file offset within the archive file, in hexadecimal format.
8	samfs1	File system name.
9	118.51	Inode number and generation number. The generation number is used in addition to the inode number for uniqueness because inode numbers are reused.
10	162514	Length of the file if the file is written on only one volume. Length of the section if the file is written on multiple volumes.
11	t0/fdn	Path and name of the file relative to the file system's mount point.

TABLE 9-1 Archiver Log File Fields (Continued)

Field	Example Value	Content
12	f	Type of file, as follows: <ul style="list-style-type: none"> <li>■ d — Directory</li> <li>■ f — Regular file</li> <li>■ l — Symbolic link</li> <li>■ R — Removable media file</li> <li>■ I — Segment index</li> <li>■ S — Data segment</li> </ul>
13	0	Section of an overflowed file or segment. If the file is an overflowed file, the value is nonzero. For all other file types, the value is 0.
14	56	Equipment ordinal of the drive on which the file was archived.

## Data Verification

You can enable data verification for archive copies. This feature checks for data corruption on any data that is copied to secondary and/or tertiary media.

The data verification process performs a read-after-write verification test and records a confirmation of data validity in the metadata properties for that file. The process uses the `ssum` option to mark files and directories to be verified. The normal checksum method is employed to verify copies written to tape or disk archive.

Use the `ssum -e` command to set data verification for a file or directory. A child directory inherits the data verification properties of its parent. This command forces the generation and use of checksums for archiving and staging. It also prevents the release of the file until all archive copies have been created and their checksums verified. Only superuser can set this attribute on a file or directory.

---

**Note** – Data verification places an additional burden on stager resources because data verification requests are placed on the stager queue in addition to normal stage requests. Data verification also leads to additional tape mounts and therefore affects archiver and stager performance. Because a file cannot be released until all archive copies are made, using data verification might also require additional disk cache.

---

## Configuring the Archiver

---

This chapter describes how to configure the archiver by modifying the `archiver.cmd` file.

### About the `archiver.cmd` File

The `archiver.cmd` file controls the archiver's behavior.

The archiver copies files from a file system to volumes on removable media cartridges or to disk partitions in another file system. You can tune the archiver operations to suit the types of files at your site and to suit your site's data protection needs by creating an archiver command file named `/etc/opt/SUNWsamfs/archiver.cmd`. You are not required to create an `archiver.cmd` file, but the efficiency and performance of the archiver is improved if you tune the archiver to your site.

By default, the archiver starts whenever the `sam-fsd` process is started and a file system is mounted.

If the `archiver.cmd` file does not exist, the archiver is put in a `wait` state. To restart the archiver, use the `samcmd arun` command. If no `archiver.cmd` file is found after the restart, the archiver continues to run, using the following default settings:

- Archives all files to all available volumes in all configured libraries.
- Makes one copy of each file.
- Sets the archive age for all files to 4 minutes. The archive age is the amount of time since a file's last modification.
- Sets the archive interval to 10 minutes. The archive interval is the amount of time that elapses between complete archiving processes.

To tune the actions of the archiver for your site, set directives in the `archiver.cmd`. A directive acts like a command parameter and consist of lines of text in the `archiver.cmd` file. The following rules apply to the `archiver.cmd` file:

- Each directive line contains one or more fields separated by spaces or tabs.
- Any text that appears after the hash character (#) is treated as a comment and is not examined.
- Lines that end with a backslash character (\) are joined to the next line.

The archiver .cmd file uses two types of directives.

For comprehensive information about the archiver .cmd directives, see “[archiver.cmd\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

Whenever you make changes to the archiver .cmd file, check for syntax errors using the archiver(1M) command. This command produces a list of all options and writes a list of the volumes, file system content, and errors to the standard output file (stdout). If you encounter errors, correct them in the file and run the archiver(1M) command again to verify your corrections. The archiver does not archive any files if it finds errors in the archiver .cmd file.

## Archive Directives

Archive directives specify the general archive operations and consist of two main areas in the archiver .cmd file:

- At the top of the file, *global directives* affect all file systems defined in your mc f file.
- The lower part of the file contains *file system-specific directives*. These directives must come after the global directives. For any file system, these directives override the global directives. The file-system-specific directives start with an fs= *name* directive that identifies the file system.

Within a file system's directives, you can also specify archive copy directives to customize the number and frequency of the archive copies.

See [Chapter 11, “Archive Directives \(archiver .cmd\)”](#) for detailed information.

## Archive Set Directives

An archive set identifies a group of files to be archived regardless of their file system. A file in a file system can be a member of only one archive set. Files in an archive set share common criteria that pertain to the size, ownership, group, or directory location.

The archive set controls the destination of the archive copy, how long the copy is kept archived, and how long the software waits before archiving the data. All files in an archive set are copied to the volumes associated with that archive set.

The directives for an archive set include:

- Assignment directives – Define archive sets.

- Copy parameters – Define how each archive set is archived.
- VSN association directives – Assign volumes to archive sets.
- VSN pools directives – Define a collection of volumes.

See [Chapter 12, “Archive Set Directives \(archiver.cmd\)”](#) for detailed information.

## ▼ How to Create an archiver . cmd File Using the Command Line

- If your site has an `/etc/opt/SUNWsamfs/archiver . cmd` file and your system is already archiving files, do not make changes to it. Copy the file to a location where you can edit and test it. When it is verified, replace the existing file with the new one.
- If your site does not have a `archiver . cmd` file, you can edit the file in the `/etc/opt/SUNWsamfs` directory.

### 1 Edit the archiver . cmd file to add or change the directives that control archiving at your site.

For information about the directives you can include in this file, see [Chapter 11, “Archive Directives \(archiver . cmd\)”](#) and [Chapter 12, “Archive Set Directives \(archiver.cmd\)”](#) for details.

### 2 Save and close the archiver . cmd file.

### 3 Verify the file for the current SAM-QFS environment.

If you are working with a test `archiver . cmd` file, use the `-c` option with the `archiver(1M)` command and supply the file name.

```
# archiver -lv
```

### 4 If you are working with a test file, move it to `/etc/opt/SUNWsamfs/archiver . cmd`.

### 5 Use the `samd config` command to propagate the file changes.

```
# samd config
```

## ▼ How to Create an archiver . cmd File Using SAM-QFS Manager

When you create or edit an archive policy for a file system within the SAM-QFS Manager interface, the `archiver . cmd` file is automatically created or edited.

### 1 On the Servers page, select the name of the server for which you want to create a policy.

The File Systems Summary page is displayed.

**2 Select the Archive Administration node in the navigation tree.**

The Archive Policies Summary page is displayed.

**3 Click New.**

The New Archive Policy wizard is displayed.

**4 Follow the steps in the wizard.**

For detailed instructions about using the New Archive Policy wizard, see the SAM-QFS Manager online help.

When you save the new archive policy, it is automatically written to the archiver .cmd file.

## Examples of archiver .cmd Files

This section provides some examples of archiving configurations.

**EXAMPLE 10-1** Simple archiver.cmd File

The following example shows a simple archiver .cmd file that you can modify. Add directives only to accommodate more archive sets, copies, and VSN usage.

```
archiver.cmd# One file system = samfs1
archmax = sg 2G
examine = noscan
fs = samfs1
logfile = /var/opt/SUNWsamfs/log/samfs1.log
all_archset .
  1 -norelease 10m
  2 -norelease 10m
params
allsets -sort path -offline_copy stageahead -reserve set
allsets -recycle_hwm 50 -recycle_mingain 90 \
-recycle_vsncount 5 -recycle_dataquantity 40G
allsets.1 -startage 6h -startsize 6G -startcount 30000
allsets.2 -startage 10h -startsize 12G \
-startcount 60000 -archmax 12G
endparams
vsn
all.1 li .*
all.2 li .*
endvsn
```

**EXAMPLE 10-2** Advanced archiver .cmd File

The following example shows a complex archiver .cmd file. The comments indicate the various types of directives.

```
# Global directives

archmax = li 8G
```

## EXAMPLE 10-2 Advanced archiver.cmd File (Continued)

```

examine = noscan
scanlist_squash = on

# Limit the drives

drives = stk50 3

# File selection

fs = samfs1
logfile = /var/adm/samfs1.log
archive_archset .
    1 -norelease 5m
    2 -norelease 5m

fs = samfs2
logfile = /var/adm/samfs2.log
archive_archset .
    1 -norelease 5m
    2 -norelease 5m

fs = samfs3
logfile = /var/adm/samfs3.log
archive_archset .
    1 -norelease 5m
    2 -norelease 5m

fs = samfs4
logfile = /var/adm/samfs4.log
archive_archset .
    1 -norelease 5m
    2 -norelease 5m

fs = samfs5
logfile = /var/adm/samfs5.log
archive_archset .
    1 -norelease 5m
    2 -norelease 5m

# The following information is for tape archiving.
# The recycler is not set up to actually recycle. It is set up for
# data checks and file recovery.

params allsets -sort path -offline_copy stageahead -reserve set
allsets -recycle_ignore allsets -recycle_hwm 50 -recycle_mingain 90 -recycle_vsncount 1
allsets.1 -startage 6h -startsize 8G -startcount 90000 -drives 3 -drivemin 10G
allsets.2 -startage 10h -startsize 12G -startcount 90000
-archmax 12G -drives 3 -drivemin 10G
endparams

# Define VSNS for archive sets

vsns
archive.1 li .*
archive.2 li .*
endvsns

```

**EXAMPLE 10-3** No archiver .cmd File

This example illustrates the action of the archiver when no archiver .cmd file is used in a SAM-QFS environment that has one file system, an optical automated library with two drives, and six cartridges.

The following archiver -lv output shows that the default media selected by the archiver is type mo. Only the mo media are available.

```
# archiver -lv
Notify file: /etc/opt/SUNWsamfs/scripts/archiver.sh
Archive media:
media:lt archmax: 512.0M Volume overflow not selected
media:mo archmax: 4.8M Volume overflow not selected
```

The following archiver -lv output indicates that the archiver uses two drives. It lists the 12 volumes, storage capacity, and VSNs with available space.

```
Archive libraries:
Device:hp30 drives_available:2 archive_drives:2
Catalog:
mo.optic00 capacity: 1.2G space: 939.7M -il-o-----
mo.optic01 capacity: 1.2G space: 934.2M -il-o-----
mo.optic02 capacity: 1.2G space: 781.7M -il-o-----
mo.optic03 capacity: 1.2G space: 1.1G -il-o-----
mo.optic10 capacity: 1.2G space: 85.5M -il-o-----
mo.optic11 capacity: 1.2G space: 0 -il-o-----
mo.optic12 capacity: 1.2G space: 618.9k -il-o-----
mo.optic13 capacity: 1.2G space: 981.3M -il-o-----
mo.optic20 capacity: 1.2G space: 1.1G -il-o-----
mo.optic21 capacity: 1.2G space: 1.1G -il-o-----
mo.optic22 capacity: 1.2G space: 244.9k -il-o-----
mo.optic23 capacity: 1.2G space: 1.1G -il-o-----
```

The following archiver -lv output shows that the archive set samfs includes both metadata and data files. The archiver makes one copy of the files when their archive age reaches the default four minutes (240 seconds).

```
Archive file selections:
Filesystem samfs Logfile:
samfs Metadata
copy:1 arch_age:240
samfs1 path:
copy:1 arch_age:240
```

The following archiver -lv output shows the files in the archive sets archived to the volumes in the indicated order.

```
Archive sets:
allsets
samfs.1
media: mo (by default)
Volumes:
optic00
```

**EXAMPLE 10-3** No archiver .cmd File (Continued)

```

optic01
optic02
optic03
optic10
optic12
optic13
optic20
optic21
optic22
optic23
Total space available: 8.1G

```

**EXAMPLE 10-4** Data Files Archived Separately From Metadata

This example shows how to separate data files into two archive sets separate from the metadata. The environment includes a manually mounted DLT tape drive in addition to an optical automated library. The big files are archived to tape, and the small files are archived to optical cartridges.

The following example shows the content of the archiver .cmd file.

```

# archiver -lv -c example2.cmd
Reading archiver command file "example2.cmd"
1: # Example 2 archiver command file
2: # Simple selections based on size
3:
4: logfile = /var/opt/SUNWsamfs/archiver/log
5: interval = 5m
6:
7: # File selections.
8: big . -minsize 500k
9: all .
10: 1 30s
11:
12: vsns
13: samfs.1 mo .*[0-2] # Metadata to optic00 - optic02
14: all.1 mo .*[3-9] .*[1-2][0-9] # All others for files
15: big.1 lt .*
16: endvsns

```

The following archiver -lv output shows the media and drives to be used.

```

Notify file: /etc/opt/SUNWsamfs/scripts/archiver.sh
Archive media:
media:lt archmax: 512.0M Volume overflow not selected
media:mo archmax: 4.8M Volume overflow not selected
Archive libraries:
Device:hp30 drives_available:0 archive_drives:0
Catalog:
mo.optic00 capacity: 1.2G space: 939.7M -il-o-----
mo.optic01 capacity: 1.2G space: 934.2M -il-o-----
mo.optic02 capacity: 1.2G space: 781.7M -il-o-----

```

**EXAMPLE 10-4** Data Files Archived Separately From Metadata (Continued)

```

mo.optic03 capacity: 1.2G space: 1.1G -il-o-----
mo.optic04 capacity: 1.2G space: 983.2M -il-o-----
mo.optic10 capacity: 1.2G space: 85.5M -il-o-----
mo.optic11 capacity: 1.2G space: 0 -il-o-----
mo.optic12 capacity: 1.2G space: 618.9k -il-o-----
mo.optic13 capacity: 1.2G space: 981.3M -il-o-----
mo.optic20 capacity: 1.2G space: 1.1G -il-o-----
mo.optic21 capacity: 1.2G space: 1.1G -il-o-----
mo.optic22 capacity: 1.2G space: 244.9k -il-o-----
mo.optic23 capacity: 1.2G space: 1.1G -il-o-----
Device:lt40 drives_available:0 archive_drives:0
Catalog:
lt.TAPE01 capacity: 9.5G space: 8.5G -il-o-----
lt.TAPE02 capacity: 9.5G space: 6.2G -il-o-----
lt.TAPE03 capacity: 9.5G space: 3.6G -il-o-----
lt.TAPE04 capacity: 9.5G space: 8.5G -il-o-----
lt.TAPE05 capacity: 9.5G space: 8.5G -il-o-----
lt.TAPE06 capacity: 9.5G space: 7.4G -il-o-----

```

The following archiver -lv output shows the organization of the file system. Files bigger than 512000 bytes (500 kilobytes) are archived after four minutes. All other files are archived after 30 seconds.

```

Archive file selections:
Filesystem samfs Logfile: /var/opt/SUNWsamfs/archiver/log
samfs Metadata
copy:1 arch_age:240
big path:. minsize:502.0k
copy:1 arch_age:240
all path:.
copy:1 arch_age:30

```

The following archiver -lv output shows the division of the archive sets among the removable media.

```

Archive sets:
allsets
all.1
media: mo
Volumes:
optic03
optic04
optic10
optic12
optic13
optic20
optic21
optic22
optic23
Total space available: 6.3G
big.1
media: lt
Volumes:
TAPE01

```

**EXAMPLE 10-4** Data Files Archived Separately From Metadata (Continued)

```

TAPE02
TAPE03
TAPE04
TAPE05
TAPE06
Total space available: 42.8G
samfs.1
media: mo
Volumes:
optic00
optic01
optic02
Total space available: 2.6G

```

---

**Note** – The archiver(1M) -lv command shows only VSNs with space available.

---

**EXAMPLE 10-5** User and Data Files Archived to Various Media

In this example, user files and project data files are archived to various media. Files from the directory data are segregated by size to optical and tape media. Files assigned to the group ID pict are assigned to another set of volumes. Files in the directories tmp and users/bob are not archived. Archiving is performed at 15-minute intervals, and an archiving record is kept.

```

# archiver -lv -c example3.cmd
Reading archiver command file "example3.cmd"
1: # Example 3 archiver command file
2: # Segregation of users and data
3:
4: interval = 30s
5: logfile = /var/opt/SUNWsamfs/archiver/log
6:
7: no_archive tmp
8:
9: fs = samfs
10: no_archive users/bob
11: prod_big data -minsize 50k
12: 1 1m 30d
13: 2 3m
14: prod data
15: 1 1m
16: proj_1 projs/proj_1
17: 1 1m
18: 2 1m
19: joe . -user joe
20: 1 1m
21: 2 1m
22: pict . -group pict
23: 1 1m
24: 2 1m
25:
26: params

```

**EXAMPLE 10-5** User and Data Files Archived to Various Media *(Continued)*

```

27: prod_big.1 -drives 2
28: prod_big.2 -drives 2
29: endparams
30:
31: vsns
32: samfs.1 mo optic0[0-1]$
33: joe.1 mo optic01$
34: pict.1 mo optic02$
35: pict.2 mo optic03$
36: proj_1.1 mo optic1[0-1]$
37: proj_1.2 mo optic1[2-3]$
38: prod.1 mo optic2.$
39: joe.2 lt 0[1-2]$
40: prod_big.1 lt 0[3-4]$
41: prod_big.2 lt 0[5-6]$
42: endvsns
Notify file: /etc/opt/SUNWsamfs/scripts/archiver.sh
Archive media:
media:lt archmax: 512.0M Volume overflow not selected
media:mo archmax: 4.8M Volume overflow not selected
Archive libraries:
Device:hp30 drives_available:0 archive_drives:0
Catalog:
mo.optic00 capacity: 1.2G space: 939.7M -il-o-----
mo.optic01 capacity: 1.2G space: 934.2M -il-o-----
mo.optic02 capacity: 1.2G space: 781.7M -il-o-----
mo.optic03 capacity: 1.2G space: 1.1G -il-o-----
mo.optic04 capacity: 1.2G space: 983.2M -il-o-----
mo.optic10 capacity: 1.2G space: 85.5M -il-o-----
mo.optic11 capacity: 1.2G space: 0 -il-o-----
mo.optic12 capacity: 1.2G space: 618.9k -il-o-----
mo.optic13 capacity: 1.2G space: 981.3M -il-o-----
mo.optic20 capacity: 1.2G space: 1.1G -il-o-----
mo.optic21 capacity: 1.2G space: 1.1G -il-o-----
mo.optic22 capacity: 1.2G space: 244.9k -il-o-----
mo.optic23 capacity: 1.2G space: 1.1G -il-o-----
Device:lt40 drives_available:0 archive_drives:0
Catalog:
lt.TAPE01 capacity: 9.5G space: 8.5G -il-o-----
lt.TAPE02 capacity: 9.5G space: 6.2G -il-o-----
lt.TAPE03 capacity: 9.5G space: 3.6G -il-o-----
lt.TAPE04 capacity: 9.5G space: 8.5G -il-o-----
lt.TAPE05 capacity: 9.5G space: 8.5G -il-o-----
lt.TAPE06 capacity: 9.5G space: 7.4G -il-o-----
Archive file selections:
Filesystem samfs Logfile: /var/opt/SUNWsamfs/archiver/log
samfs Metadata
copy:1 arch_age:240
no_archive Noarchive path:users/bob
prod_big path:data minsize:50.2k
copy:1 arch_age:60 unarch_age:2592000
copy:2 arch_age:180
prod path:data
copy:1 arch_age:60
proj_1 path:projs/proj_1
copy:1 arch_age:60

```

**EXAMPLE 10-5** User and Data Files Archived to Various Media (Continued)

```

copy:2 arch_age:60
joe path:. uid:10006
copy:1 arch_age:60
copy:2 arch_age:60
pict path:. gid:8005
copy:1 arch_age:60
copy:2 arch_age:60
no_archive Noarchive path:tmp
samfs path:.
copy:1 arch_age:240
Archive sets:
allsets
joe.1
media: mo
Volumes:
optic01
Total space available: 934.2M
joe.2
media: lt
Volumes:
TAPE01
TAPE02
Total space available: 14.7G
pict.1
media: mo
Volumes:
optic02
Total space available: 781.7M
pict.2
media: mo
Volumes:
optic03
Total space available: 1.1G
prod.1
media: mo
Volumes:
optic20
optic21
optic22
optic23
Total space available: 3.3G
prod_big.1
media: lt drives:2
Volumes:
TAPE03
TAPE04
Total space available: 12.1G
prod_big.2
media: lt drives:2
Volumes:
TAPE05
TAPE06
Total space available: 16.0G
proj_1.1
media: mo
Volumes:

```

**EXAMPLE 10-5** User and Data Files Archived to Various Media (Continued)

```

optic10
Total space available: 85.5M
proj_1.2
media: mo
Volumes:
optic12
optic13
Total space available: 981.9M
samfs.1
media: mo
Volumes:
optic00
optic01
Total space available: 1.8G

```

**EXAMPLE 10-6** User and Data Files Archived to Optical Media

In this example, user files and project data files are archived to optical media.

Four VSN pools are defined. Three pools are used for user, data, and project, and one is a scratch pool. When `proj_pool` runs out of media, it relies on `scratch_pool` to reserve volumes. This example shows how to reserve volumes for each archive set based on the set component, owner component, and file system component. Archiving is performed at 10-minute intervals, and an archiving log is kept.

The following example shows the `archiver.cmd` file and archiver output.

```

Reading archiver command file "example4.cmd"
1: # Example 4 archiver command file
2: # Using 4 VSN pools
3:
4: interval = 30s
5: logfile = /var/opt/SUNWsamfs/archiver/log
6:
7: fs = samfs
8: users users
9: 1 10m
10:
11: data data
12: 1 10m
13:
14: proj projects
15: 1 10m
16:
17: params
18: users.1 -reserve user
19: data.1 -reserve group
20: proj.1 -reserve dir -reserve fs
21: endparams
22:
23: vsnpools
24: users_pool mo optic0[1-3]$
25: data_pool mo optic1[0-1]$

```

**EXAMPLE 10-6** User and Data Files Archived to Optical Media (Continued)

```

26: proj_pool mo optic1[2-3]$
27: scratch_pool mo optic2.$
28: endvsnpools
29:
30: vsn
31: samfs.1 mo optic00
32: users.1 mo -pool users_pool -pool scratch_pool
33: data.1 mo -pool data_pool -pool scratch_pool
34: proj.1 mo -pool proj_pool -pool scratch_pool
35: endvsns
Notify file: /etc/opt/SUNWsamfs/scripts/archiver.sh
Archive media:
media:mo archmax: 4.8M Volume overflow not selected
Archive libraries:
Device:hp30 drives_available:0 archive_drives:0
Catalog:
mo.optic00 capacity: 1.2G space: 939.7M -il-o-----
mo.optic01 capacity: 1.2G space: 934.2M -il-o-----
mo.optic02 capacity: 1.2G space: 781.7M -il-o-----
mo.optic03 capacity: 1.2G space: 1.1G -il-o-----
mo.optic04 capacity: 1.2G space: 983.2M -il-o-----
mo.optic10 capacity: 1.2G space: 85.5M -il-o-----
mo.optic11 capacity: 1.2G space: 0 -il-o-----
mo.optic12 capacity: 1.2G space: 618.9k -il-o-----
mo.optic13 capacity: 1.2G space: 981.3M -il-o-----
mo.optic20 capacity: 1.2G space: 1.1G -il-o-----
mo.optic21 capacity: 1.2G space: 1.1G -il-o-----
mo.optic22 capacity: 1.2G space: 244.9k -il-o-----
mo.optic23 capacity: 1.2G space: 1.1G -il-o-----
Archive file selections:
Filesystem samfs Logfile: /var/opt/SUNWsamfs/archiver/log
samfs Metadata
copy:1 arch_age:240
users path:users
copy:1 arch_age:600
data path:data
copy:1 arch_age:600
proj path:projects
copy:1 arch_age:600
samfs path:.
copy:1 arch_age:240
VSN pools:
data_pool media: mo Volumes:
optic10
Total space available: 85.5M
proj_pool media: mo Volumes:
optic12
optic13
Total space available: 981.9M
scratch_pool media: mo Volumes:
optic20
optic21
optic22
optic23
Total space available: 3.3G
users_pool media: mo Volumes:

```

**EXAMPLE 10-6** User and Data Files Archived to Optical Media (Continued)

```
optic01
optic02
optic03
Total space available: 2.7G
Archive sets:
allsets
data.1
reserve:/group/
media: mo
Volumes:
optic10
optic20
optic21
optic22
optic23
Total space available: 3.4G
proj.1
reserve:/dir/fs
media: mo
Volumes:
optic12
optic13
optic20
optic21
optic22
optic23
Total space available: 4.2G
samfs.1
media: mo
Volumes:
optic00
Total space available: 939.7M
users.1
reserve:/user/
media: mo
Volumes:
optic01
optic02
optic03
optic20
optic21
optic22
optic23
Total space available: 6.0G
```

## About Disk Archiving

Archiving is the process of copying a file from online disk to archive media. With disk archiving, the archive medium is online disks in a file system.

Disk archiving can be implemented so that the files from one file system are archived to another file system on the same host or to another file system on a different host. When disk archiving is

configured for two host systems, the systems act as a client and a server. The host system where the source files reside is the *client* and the host system where the archive copies are written is the *server*.

The file system to which the archive files are written can be any UNIX file system. However, if disk archive copies are written to a different host, that host must have at least one QFS or SAM-QFS file system installed on it.

The archiver treats files archived to disk volumes in the same way as files archived to volumes in a library. You can make one, two, three, or four archive copies. If you are making multiple archive copies, you can write one of the archive copies to disk volumes and write the other archive copies to removable media volumes. In addition, if you archive to disk volumes that are in an archived file system, the archive copies are also archived according to the `archiver.cmd` file rules for that file system.

When you plan disk archiving for your site, consider the following guidelines:

- Create disk volumes on a different disk from the one on which the original files reside.
- Make more than one archive copy and write to more than one type of archive media. For example, create archive copy 1 to disk volumes, copy 2 to tape, and copy 3 to magneto-optical disc.
- If you are archiving files to a file system on a server system, the archive copies can also be written to removable media cartridges in a library attached to that server system.

The following list summarizes some of the similarities and differences between archiving to disk volumes and archiving to removable media:

- Archive copies in disk volumes are not recorded in a catalog.
- Archive copies in disk volumes do not appear in the historian.
- Archiving to disk volumes does not rely on entries in the `mcf` file. Instead, you specify disk archive sets in the `archiver.cmd` file and you define disk volumes in `/etc/opt/SUNWsamfs/diskvols.conf`.
- To archive to disk volumes, you must define disk archive sets in the `archiver.cmd` file before mounting the file system. When you archive to removable media volumes, you can begin archiving after the file system is mounted without changing any of the default values in the `archiver.cmd` file.

## About the `diskvols.conf` File

Disk archiving does not rely on entries in the `mcf` file. You specify disk archive sets in the `archiver.cmd` file and you define disk volumes in `/etc/opt/SUNWsamfs/diskvols.conf`.

Create the `diskvols.conf` file on the system where the source files reside. Depending on where the archive copies are written, the `diskvols.conf` file contains the following information:

- If the archive copies are written to a file system on the same host system, the `diskvols.conf` file defines the VSNs and the paths to each VSN.
- If the archive copies are written to a different host system, the `diskvols.conf` file contains the host name of that server system. In this case, create another `diskvols.conf` file on the server system that specifies the host names of the client systems.



**Caution** – Be very careful when configuring the recycler if you are using disk archiving in an environment with multiple SAM-QFS servers. The `diskvols.conf` file for each SAM-QFS server must point to a unique set of disk volume resource specifications (disk archiving target directories). If any of the target directories are shared by the SAM-QFS servers, running the recycler from one SAM-QFS server destroys the disk archive data that is managed by the other SAM-QFS server.

The `diskvols.conf` file must contain the following information:

*VSN-name [host-name:] path*

Field	Content
<i>VSN-name</i>	A unique alphanumeric name of up to 31 characters for the disk VSN to receive the archive copies.
<i>host-name</i>	The name of the host, followed by a colon character (:), to which archive copies are written. If you are archiving to a disk on another host, specify the name of the destination server.  If you are archiving to a file system that resides on the same server as the source file system, do not specify the host name.
<i>path</i>	The full path, relative to the mount point, to the directory that receives the archive files. This directory must be in place before archiving can start, and the destination file system must be mounted. For example, if archive copies are written to the <code>vsns</code> directory in the <code>archivefs1</code> file system, specify <code>/archivefs1/vsns</code> in the path field. Create the destination directory with write permission granted only to root.

The following additional rules apply to the `diskvols.conf` file:

- Start each comment line with a hash character (#). All text to the right of the # is ignored.
- To continue a line, put an apostrophe character (') at the end.

For more information, see “[diskvols.conf\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Disk Archiving Directives

When archiving to disk volumes, the archiver recognizes the directives that define archive sets and recycling and ignores directives that pertain to removable media cartridges. The archiver recognizes the directives described in the following sections:

- “Archive Set Copy Parameters Exceptions” on page 117
- “Archive Directives Exceptions” on page 117
- “Recycling Directives Exceptions” on page 117
- “vsn Directives” on page 117
- “clients and endclients Directives” on page 118
- “-recycle minobs Recycler Directive” on page 118

### Archive Set Copy Parameters Exceptions

All the parameters described in “Archive Set Copy Parameters” on page 144 are valid except for the following:

- -reserve *method*
- -tapenonstop

To configure an archive set to write multiple, simultaneous disk archive streams, use the `-drives` parameter. In this configuration, volumes are selected in a round-robin manner starting with the volume that has the highest percentage of available space. However, if the parameter `-fillvsn` is specified, the volume with the least percentage of remaining space is selected first.

### Archive Directives Exceptions

All the directives described in Chapter 11, “Archive Directives (`archiver.cmd`),” are valid except for the following:

- `ovflmin` *min-size*

### Recycling Directives Exceptions

All of the directives described in Table 16–3 are valid except for the following:

- `-recycle_dataquantity` *size*
- `-recycle_vsncount` *count*
- `recycle_hwm`

### vsn Directives

The following directives are valid:

- `vsns` and `endvsns`
- `vsnpools` and `endvsnpools`

Disk volumes are supported in the VSN associations section and are defined with a `dk` media type. The volumes are identified by one or more VSN expression keywords. You can also specify VSN pools from which disk volumes are to be selected, as shown in the following example.

```
vsnpools
data_pool dk disk0[0-5]
endvsnpools

vsns
arset0.1 dk disk10 disk1[2-5]
arset1.1 dk -pool data_pool
endvsns
```

Disk archiving can be carried out on the StorageTek 5800 system. The StorageTek 5800 is an online storage appliance with an integrated hardware and software architecture in which the disk-based storage nodes are arranged in a symmetric cluster. The media abbreviation for StorageTek 5800 disk archives in the `vsns` directives is `cb`.

---

**Note** – If you are using the disk volumes on the StorageTek 5800 for archiving, be aware that the StorageTek 5800 is not a traditional file system and the security considerations are different from other types of disk storage. Read the StorageTek 5800 documentation for more information.

---

## clients and endclients Directives

The `clients` and `endclients` directives are valid. If you archive files from a client host to a server host, the server system must have a `diskvols.conf` file that contains the name of the client system. The format for these directives is shown in following example. For *client-system*, specify the host name of the client system that contains the source files.

```
clients
client-system1
client-system2
...
endclients
```

## -recycle minobs Recycler Directive

The `-recycle_minobs_percent_recycler` directive is valid. This option is used to set a threshold for the recycler's rearchiving process for disk archives. The default threshold is 50 percent. When the percentage of obsolete files within an archived tar file on the disk reaches this threshold, the recycler moves the valid files from the archive into a new tar file. When all of the valid files have been moved, the original tar file is marked as a candidate for removal from the disk archive. This option is ignored for removable media recycling.

## ▼ How to Enable Disk Archiving on the Client Host

Perform this procedure on the client host system that contains the files to be archived. As an alternative, you can use the SAM-QFS Manager interface to specify an archive policy that archives to disk volumes. This action updates both the `archiver.cmd` file and the `diskvols.conf` file.

---

**Note** – If you are configuring a Sun QFS file system for the first time at your site and have therefore not yet installed the SAM-QFS software on another host, you must write the archive copies to disk volumes in a file system that is on the same host as the source files. If you configure a Sun QFS file system on another host at a later time, you can modify your configuration files accordingly.

---

- 1 **Become superuser on the client host system.**
- 2 **Create or open the file `/etc/opt/SUNWsamfs/archiver.cmd`.**
- 3 **Add disk archive set directives as in the following example:**

```
#
vsns
archset1.1 dk disk01
archset2.1 dk disk02
archset3.1 dk disk03
endvsns
```

Disk archiving can also be carried out on the StorageTek 5800 system. The StorageTek 5800 is an online storage appliance with an integrated hardware and software architecture in which the disk-based storage nodes are arranged in a symmetric cluster. The media abbreviation for StorageTek 5800 disk archives in the `vsns` directives is `cb`.

For more information about specifying archive sets, see “[archiver.cmd\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* or see Chapter 12, “Archive Set Directives (`archiver.cmd`).”

- 4 **Save and close the `archiver.cmd` file.**
- 5 **Create a file named `diskvols.conf`.**
- 6 **Specify the directories to which the archive copies will be written.**

The following example shows a `diskvols.conf` file that archives files from three archive sets. The disk volumes named `disk01` and `disk02` reside in a file system on the server system named `otherserver`. Disk volume `disk03` resides on the same host as the files to be archived.

```
# This is file sourceserver:/etc/opt/SUNWsamfs/diskvols.conf
# on the client
#
# VSN_name [host_name:] path
```

```
#
disk01 otherserver:/sam/archset1
disk02 otherserver:/sam/archset2
disk03 /sam/archset3
```

- 7 **Save and close the `diskvols.conf` file.**
- 8 **Create directories in the file system to which the archive copies will be written.**

For example:

```
# mkdir sam
# cd sam
# mkdir archset1
# mkdir archset2
```

- 9 **Verify the syntax in the `archiver.cmd` file:**

```
# archiver -lv
```
- 10 **If any errors are found, correct them before proceeding.**

## ▼ **How to Configure Disk Archiving on the Server Host**

Perform this procedure only if you are writing your archive copies to a host system that is different from the host system upon which the source files reside. At least one QFS or SAM-QFS file system must be created on this host. If you create source files and write archive copies to the same host system, you do not need to perform this procedure.

---

**Note** – You can use the SAM-QFS Manager interface to enable disk archiving by specifying an archive policy that archives to disk VSNs. This action updates both the `archiver.cmd` file and the `diskvols.conf` file.

---

In this situation, you are creating a client/server environment:

- The client is the host that contains the source files.
- The server is the host to which the archive copies are written.

- 1 **Become superuser on the server.**
- 2 **Create or open the file `/etc/opt/SUNWsamfs/archiver.cmd`.**
- 3 **Edit the `archiver.cmd` file to add disk archive set directives as in the following example:**

```
#
vsn
archset1.1 dk disk01
archset2.1 dk disk02
archset3.1 dk disk03
endvsn
```

For more information about specifying archive sets, see “archiver.cmd(4)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* or see Chapter 12, “Archive Set Directives (archiver.cmd)”

**4 Save and close the file.**

**5 Change to the file system to which you want to write the archive copies.**

For example:

```
# cd /ufs1
```

**6 Create directories in the file system.**

For example:

```
# mkdir sam
# cd sam
# mkdir archset1
# mkdir archset2
```

**7 Create the file /etc/opt/SUNWsamfs/diskvols.conf.**

**8 Specify the clients and endclients directives and the name of the client.**

The name of the client in the following example is sourceserver.

```
# This is
# file destination_server:/etc/opt/SUNWsamfs/diskvols.conf
# on the server
#
clients
sourceserver
endclients
```

**9 Save and close the diskvols.conf file.**

## ▼ How to Enable Disk Archiving

You can enable disk archiving at any time. The procedure assumes that you are adding disk archiving to an existing archiving configuration.

**Before You Begin** Make certain that the server host to which you want to write your disk archive copies has at least one QFS or SAM-QFS file system installed on it.

**1 Become superuser on the host system that contains the files to be archived.**

**2 Enable or configure disk archiving on the client and server hosts.**

Follow the “How to Enable Disk Archiving on the Client Host” on page 119 procedure or “How to Configure Disk Archiving on the Server Host” on page 120 procedure.

- 3 On the client host, propagate the configuration file changes and restart the system.

```
# samd config
```

- 4 If you are archiving to disk on a different host, follow these steps:
  - a. Become superuser on the server host.
  - b. Use the `samd config` command to propagate the configuration file changes and restart the destination system.
- 5 If you are archiving to a StorageTek 5800 system, upgrade the StorageTek 5800 metadata schema configuration.

Follow the procedures documented in the StorageTek 5800 System Administration Guide and use the XML overlay in the following example to define the metadata that is used by SAM-QFS.

```
<<?xml version="1.0" encoding="UTF-8"?>

<metadataConfig>
<schema>
<namespace name="com">
<namespace name="sun">
<namespace name="samfs">
<field name="archiveId" type="string" indexable="true"/>
<field name="fileName" type="string" indexable="true"/>
<field name="modTime" type="time" indexable="true"/>
</namespace>
</namespace>
</namespace>
</schema>

<fsViews>
</fsViews>

</metadataConfig>
```

## Disk Archiving Examples

The following examples show disk archiving configurations.

### Example 1

In this example, VSNs identified as `disk01`, `disk02`, and `disk04` are written to `pluto`, the host system where the original source files reside. VSN `disk03` is written to a VSN on server system `mars`.

The following example shows the `diskvols.conf` file that resides on client system `pluto`.

```
# This is file /etc/opt/SUNWsamfs/diskvols.conf on pluto
# VSN Name [Host Name:]Path
#
```

```

disk01 /sam_arch1
disk02 /sam_arch2/proj_1
disk03 mars:/sam_arch3/proj_3
disk04 /sam_arch4/proj_4

```

The following example shows the `diskvols.conf` file on server system mars.

```

# This is file /etc/opt/SUNWsamfs/diskvols.conf on mars
#
clients
pluto
endclients

```

The following example shows a fragment of the `archiver.cmd` file on pluto.

```

vsns
arset1.2 dk disk01
arset2.2 dk disk02 disk04
arset3.2 dk disk03
endvsns

```

## Example 2

In this example, file `/sam1/testdir0/filea` is in the archive set for `arset0.1`, and the archiver copies the content of this file to the destination path `/sam_arch1`.

The following example shows the `diskvols.conf` file.

```

# This is file /etc/opt/SUNWsamfs/diskvols.conf
#
# VSN Name [Host Name:]Path
#
disk01 /sam_arch1
disk02 /sam_arch12/proj_1

```

The following example shows the `archiver.cmd` file lines that pertain to disk archiving:

```

.
vsns
arset0.1 dk disk01
endvsns .

```

The following example shows output from the `sls(1)` command for file `filea`, which was archived to disk. Note the following information about copy 1:

- `dk` is the media type for disk archive media
- `disk01` is the VSN
- `f192` is the path to the disk archive tar file

```

# sls -D /sam1/testdir0/filea
/sam1/testdir0/filea:
mode: -rw-r----- links: 1 owner: root group: other
length: 797904 admin id: 0 inode: 3134.49

```

```
archdone;
copy 1: ---- Dec 16 14:03 c0.1354 dk disk01 f192
access: Dec 19 10:29 modification: Dec 16 13:56
changed: Dec 16 13:56 attributes: Dec 19 10:29
creation: Dec 16 13:56 residence: Dec 19 10:32
```

### Example 3

In this example, file `/sam2/my_proj/fileb` is on client host `snickers` in archive set `arset0.1`, and the archiver copies the content of this file to the destination path `/sam_arch1` on server host `mars`.

The following example shows the `diskvols.conf` file on `snickers`.

```
# This is file /etc/opt/SUNWsamfs/diskvols.conf on snickers
#
# VSN Name [Host Name:]Path
#
disk01 mars:/sam_arch1
```

The following example shows the `diskvols.conf` file on `mars`.

```
# This is file /etc/opt/SUNWsamfs/diskvols.conf on mars
#
clients
snickers
endclients
```

The following example shows the directives in the `archiver.cmd` file that relate to this example.

```
.
vsns
arset0.1 dk disk01
endvsns .
```

# Archive Directives (`archiver.cmd`)

---

This chapter provides details about the archive directives.

## Global Archiving Directives

Global directives control the overall archiver operation and enable you to optimize operations for your site. You can add global directives directly to the `archiver.cmd` file, or you can specify them using the SAM-QFS Manager software. For more information about using SAM-QFS Manager to set global directives, see the SAM-QFS Manager online help.

Specify the global directives before you specify any file system directives (`fs=` directives). The archiver issues a message if it detects a global directive located after an `fs=` directive.

You can identify a global directive in the `archiver.cmd` file by either the presence of the equal sign (=) in the second field or the absence of additional fields. The following global directives are supported:

- “`archivemeta` Directive: Controlling Whether Metadata Is Archived” on page 126
- “`archmax` Directive: Controlling the Size of Archive Files” on page 126
- “`bufsize` Directive: Setting the Archiver Buffer Size” on page 127
- “`drives` Directive: Controlling the Number of Drives Used for Archiving” on page 128
- “`examine` Directive: Controlling Archive Scans” on page 128
- “`interval` Directive: Specifying an Archive Interval” on page 129
- “`logfile` Directive: Specifying an Archiver Log File” on page 129
- “`notify` Directive: Renaming the Event Notification Script” on page 130
- “`ovflmin` Directive: Controlling Volume Overflow” on page 130
- “`scanlist_squash` Directive: Controlling Scanlist Consolidation” on page 132
- “`setarchdone` Directive: Controlling the Setting of the `archdone` Flag” on page 132
- “`wait` Directive: Delaying Archiver Startup” on page 132

## archivemeta Directive: Controlling Whether Metadata Is Archived

The `archivemeta` directive controls whether file system metadata is archived. If files are often moved around and there are frequent changes to the directory structures in a file system, archive the file system metadata. In contrast, if the directory structures are very stable, you can disable metadata archiving and reduce the actions performed by removable media drives. By default, metadata is not archived.

This directive has the following format:

```
archivemeta=state
```

For *state*, specify either `on` or `off`. The default is `off`.

The archiving process for metadata depends on whether you are using a Version 1 or a Version 2 superblock, as follows:

- For Version 1 file systems, the archiver archives directories, removable media files, segment index inodes, and symbolic links as metadata.
- For Version 2 file systems, the archiver archives directories and segment index inodes as metadata. Removable media files and symbolic links are stored in inodes rather than in data blocks. They are not archived. Symbolic links are archived as data.

## archmax Directive: Controlling the Size of Archive Files

The `archmax` directive specifies the maximum size of an archive file. User files are combined to form the archive file. After the *target-size* value is met, no more user files are added to the archive file. Large user files are written in a single archive file.

To change the defaults, use the following directive:

```
archmax=media target-size
```

Argument	Definition
<i>media</i>	The media type. For the list of valid media types, see “ <code>mcf(4)</code> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>target-size</i>	The maximum size of the archive file. This value is media-dependent. By default, archive files written to optical discs are no larger than 5 megabytes. The default maximum archive file size for tapes is 512 megabytes.

Setting large or small sizes for archive files has advantages and disadvantages. For example, if you are archiving to tape and `archmax` is set to a large size, the tape drive stops and starts less often. However, when writing large archive files, a premature end-of-tape causes a large amount of tape to be wasted. As a best practice, do not set the `archmax` directive to be more than 5 percent of the media capacity.

The `archmax` directive can also be set for an individual archive set.

---

**Note** – The `archmax` directive is not a valid directive for archive sets that are archived to the StorageTek 5800 media type.

---

## bufsize Directive: Setting the Archiver Buffer Size

By default, a file being archived is copied to archive media using a memory buffer. You can use the `bufsize` directive to specify a nondefault buffer size and, optionally, to lock the buffer. These actions can improve performance. You can experiment with different *buffer-size* values. This directive has the following format:

```
bufsize=media buffer-size [lock]
```

Argument	Definition
<i>media</i>	The media type. For the list of valid media types, see “ <a href="#">mcf(4)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>buffer-size</i>	A number from 2 through 1024. The default is 4. This value is multiplied by the <code>dev_blksize</code> value for the media type, and the resulting buffer size is used. The <code>dev_blksize</code> value is specified in the <code>defaults.conf</code> file. For more information about this file, see “ <a href="#">defaults.conf(4)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<code>lock</code>	Indicates whether the archiver can use locked buffers when making archive copies. If <code>lock</code> is specified, the archiver sets file locks on the archive buffer in memory for the duration of the <code>sam-arcopy</code> operation. This action avoids the overhead associated with locking and unlocking the buffer for each I/O request and results in a reduction in system CPU time. The <code>lock</code> argument must be specified only on large systems with large amounts of memory. Insufficient memory can cause an out-of-memory condition. The <code>lock</code> argument is effective only if direct I/O is enabled for the file being archived. By default, <code>lock</code> is not specified and the file system sets the locks on all direct I/O buffers, including those for archiving. For more information about enabling direct I/O, see “ <a href="#">setfa(1)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> , “ <a href="#">sam_setfa(3)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> , or the <code>-O forcedirectio</code> option on “ <a href="#">mount_samfs(1M)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .

You can specify a buffer size and a lock for each archive set basis by using the archive set copy parameters, `-bufsize` and `-lock`. For more information, see “[Archive Set Copy Parameters](#)” on page 144.

## drives Directive: Controlling the Number of Drives Used for Archiving

By default, the archiver uses all of the drives in an automated library for archiving. To limit the number of drives used, use the `drives` directive. This directive has the following format:

```
drives=auto-lib count
```

Argument	Definition
<i>auto-lib</i>	The family set name of the automated library as defined in the <code>mc f</code> file.
<i>count</i>	The number of drives to be used for archiving activities.

Also see the descriptions of the archive set copy parameters `-drivemax`, `-drivemin`, and `-drives` in [“Specifying the Number of Drives for an Archive Request: `-drivemax`, `-drivemin`, and `-drives`” on page 146](#)

## examine Directive: Controlling Archive Scans

New files and files that have changed are candidates for archiving. The archiver finds such files through one of the following methods:

- Continuous archiving, in which the archiver works with the file system to detect file changes immediately
- Scan archiving, in which the archiver scans the file system periodically for files that need to have changed.

The `example` directive for scan archiving has the following format:

```
examine=method
```

<i>method</i> Value	Definition
<code>noscan</code>	Specifies continuous archiving. After the initial scan, directories are scanned only when the content changes and archiving is required. Directory and inode information is not scanned. This archiving method provides better performance than scan archiving, particularly for file systems with more than 1,000,000 files. Default.
<code>scan</code>	Specifies scan archiving. The initial file system scan is a directory scan. Subsequent scans are inode scans.
<code>scandirs</code>	Specifies scan archiving on directories only. If the archiver finds a directory with the <code>no_archive</code> attribute set, the directory is not scanned. If you have files that do not change, place them in this type of directory to reduce the amount of time spent on archiving scans.

<i>method</i> Value	Definition
scaninodes	Specifies scan archiving on inodes only.

## interval Directive: Specifying an Archive Interval

The archiver runs periodically to examine the status of all mounted archived-enabled file systems. The timing is controlled by the archive interval, which is the time between scan operations on each file system. To change the time, use the `interval` directive.

The `interval` directive initiates full scans only when continuous archiving is not set and no `startage`, `startsize`, or `startcount` parameters have been specified. If continuous archiving is set (`examine=noscan`), the `interval` directive acts as the default `startage` value. This directive has the following format:

```
interval=time
```

For *time*, specify the amount of time you want between scan operations on a file system. By default, *time* is interpreted in seconds and has a value of 600, which is 10 minutes. You can specify a different unit of time, such as minutes or hours.

If the archiver receives the `samu` utility's `:arrun` command, it begins scanning all file systems immediately. If the `examine=scan` directive is also specified in the `archiver.cmd` file, a scan is performed after `:arrun` or `:arscan` is issued.

If the `hwm_archive` mount option is set for the file system, the archive interval can be shortened automatically. This mount option specifies that the archiver commences its scan when the file system is filling up and the high-water mark is crossed. The `high=percent` mount option sets the high-water mark for the file system.

For more information about specifying the archive interval, see “[archiver.cmd\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*. For more information on setting mount options, see “[mount\\_samfs\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## logfile Directive: Specifying an Archiver Log File

The archiver can produce a log file that contains information about each file that is archived, re-archived, or automatically unarchived. The log file is a continuous record of archival action. By default, this file is not produced. To specify a log file, use the `logfile` directive. This directive has the following format:

```
logfile=pathname
```

For *pathname*, specify the absolute path and name of the log file. The `logfile` directive can also be set for an individual file system.

**EXAMPLE 11-1** Backing Up an Archiver Log File

Assume that you want to back up the archiver log file every day by copying the previous day's log file to an alternate location. Be sure to perform the copy operation when the archiver log file is closed, not while it is open for a write operation.

1. Use the `mv` command to move the archiver log file within a UNIX file system.  
This gives any `sam-arfind` or `sam-arcopy` operations time to finish writing to the archiver log file.
2. Use the `mv` command to move the previous day's archiver log file to the file system.

## notify Directive: Renaming the Event Notification Script

The `notify` directive sets the name of the archiver's event notification script file. This directive has the following format:

```
notify=filename
```

For *filename*, specify the name of the file containing the archiver event notification script or the full path to this file. The default file name is `/etc/opt/SUNWsamfs/scripts/archiver.sh`.

The archiver executes this script to process various events in a site-specific manner. The script is called with one of the following keywords for the first argument: `emerg`, `alert`, `crit`, `err`, `warning`, `notice`, `info`, and `debug`.

Additional arguments are described in the default script. For more information, see “[archiver.sh\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## ovflmin Directive: Controlling Volume Overflow

When volume overflow is enabled, the archiver can create archived files that span multiple volumes. When a file size exceeds the specified minimum size, the archiver writes the remaining portion of this file to another volume of the same type. The portion of the file written to each volume is called a *section*.

The `sls` command lists the archive copy, showing each section of the file on each volume.

---

**Note** – Use volume overflow with caution after assessing its effect on your site. Disaster recovery and recycling are much more difficult with files that span volumes.

---

The archiver controls volume overflow through the `ovflmin` directive. By default, volume overflow is disabled. To enable volume overflow, use the `ovflmin` directive in the `archiver.cmd` file. This directive has the following format:

`ovflmin` = *media* *minimum-file-size*

Argument	Definition
<i>media</i>	The media type. For a list of valid media types, see “ <a href="#">mcf(4)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>minimum-file-size</i>	The minimum file size that you want to trigger the volume overflow.

The `ovflmin` directive can also be set for an individual archive set.

Volume overflow files do not generate checksums. For more information on using checksums, see “[ssum\(1\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Examples of Volume Overflow

A site has many files with a significant `mo` media cartridge fraction length (such as 25 percent). These files leave unused space on each volume. To use volume space efficiently, set `ovflmin` for `mo` media to a size slightly smaller than the size of the smallest file. The following directive sets the value to 150 megabytes:

```
ovflmin=mo 150m
```

In this example, two volumes are loaded for archiving and staging the files because each file overflows onto another volume.

The following example shows the archiver log file when volume overflow is enabled. The file `file50` spans three volumes with VSNs of `DLT000`, `DLT001`, and `DLT005`. The position on the volume and the size of each section is indicated in the seventh and tenth fields respectively (`7eed4.1` and `477609472` for the first volume).

For a complete description of the archiver log entry, see “[archiver\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

```
A 97/01/13 16:03:29 lt DLT000 big.1 7eed4.1 samfs1 13.7 477609472 00 big/file50 0 0
A 97/01/13 16:03:29 lt DLT001 big.1 7fb80.0 samfs1 13.7 516407296 01 big/file50 0 1
A 97/01/13 16:03:29 lt DLT005 big.1 7eb05.0 samfs1 13.7 505983404 02 big/file50 0 2
```

This portion of the archiver log file matches the `sls -D` output for file `file50`, as shown in the following example.

```
# sls -D file50
file50:
mode: -rw-rw---- links: 1 owner: gmm group: sam
length: 1500000172 admin id: 7 inode: 1407.5
offline; archdone; stage -n
copy1: ---- Jan 13 15:55 lt
section 0: 477609472 7eed4.1 DLT000
section 1: 516407296 7fb80.0 DLT001
```

```
section 2: 505983404 7eb05.0 DLT005
access: Jan 13 17:08 modification: Jan 10 18:03
changed: Jan 10 18:12 attributes: Jan 13 16:34
creation: Jan 10 18:03 residence: Jan 13 17:08
```

## scanlist\_squash Directive: Controlling Scanlist Consolidation

The `scanlist_squash` parameter controls scanlist consolidation. The default setting is `off`. This parameter can be either global or file system-specific.

When this option is enabled, the scan list entries for files in two or more subdirectories with the same parent directory that need to be scanned by `sam-arfind` at a much later time are consolidated. These directories are combined upwards to the common parent, which results in a deep recursive scan of many subdirectories. This consolidation can cause a severe performance penalty if archiving on a file system that has a large number of changes to many subdirectories.

## setarchdone Directive: Controlling the Setting of the archdone Flag

The `setarchdone` parameter is a global directive that controls the setting of the `archdone` flag when the file is examined by `sam-arfind`. This directive has the following format:

```
setarchdone=on|off
```

When all archive copies for a file have been made, the `archdone` flag is set for that file to indicate that no further archive action is required. During directory scans, the `archdone` flag is also set for files that will never be archived. Because evaluating whether a file will ever be archived can affect performance, the `setarchdone` directive gives you control over this activity. This directive controls the setting of the `archdone` flag only on files that will never be archived. It does not affect the setting of the `archdone` flag after archive copies are made.

The default setting for the directive is `off` if the `examine` directive is set to `scandirs` or `noscan`.

## wait Directive: Delaying Archiver Startup

The `wait` directive causes the archiver to wait for a start signal from `samu(1M)` or `SAM-QFS Manager`. By default, the archiver begins archiving when started by `sam-fsd(1M)`. This directive has the following format:

```
wait
```

The `wait` directive can also be set for an individual file system.

## File System Directives

Archiving controls apply to all file systems. However, you can confine some controls to an individual file system. When the archiver encounters an `fs=` directive in the `archiver.cmd` file, all subsequent directives are applied to that specific file system. Therefore, place any `fs=` directives after the general directives.

You can specify `fs=` directives either by editing the `archiver.cmd` file as described in the following sections, or by using the SAM-QFS Manager software. See the SAM-QFS Manager online help for more information.

### fs Directive: Specifying the File System

Use the `fs=` directive to specify actions for a particular file system.

For instance, you can use this directive to create a different log file for each file system. This directive has the following format:

```
fs=fsname
```

For *fsname*, specify the file system name as defined in the `mcf` file.

The general directives and archive set association directives that occur after a `fs=` directive apply only to the specified file system until another `fs=` directive is encountered.

### Global Directives as File System Directives

Several directives can be specified both as global directives for all file systems and as directives specific to only one file system. These directives are described in the following sections:

- [“interval Directive: Specifying an Archive Interval” on page 129](#)
- [“logfile Directive: Specifying an Archiver Log File” on page 129](#)
- [“scanlist\\_squash Directive: Controlling Scanlist Consolidation” on page 132](#)
- [“wait Directive: Delaying Archiver Startup” on page 132](#)

# Archive Copy Directives

By default, the archiver writes a single archive copy for files in the archive set when the archive age of the file is four minutes. To change the default behavior, use archive copy directives. Archive copy directives must appear immediately after the archive set assignment directive to which they pertain.

The archive copy directives begin with a *copy-number* value of 1, 2, 3, or 4. The digit is followed by one or more arguments that specify archive characteristics for that copy. Each archive copy directive has the following format:

```
copy-number [ -release | -norelease ] [archive-age] [unarchive-age]
```

You can specify archive copy directives either by editing the `archiver.cmd` file as described in the following sections, or by using the SAM-QFS Manager software. For more information, see the SAM-QFS Manager online help.

The following sections describe the archive copy directive arguments.

## -release Directive: Releasing Disk Space After Archiving

To specify that the disk space for files is released after an archive copy is made, use the `-release` directive after the copy number. This directive has the following format:

```
-release
```

**EXAMPLE 11-2** `archiver.cmd` File Using the `-release` Directive

In the following example, files within the group `images` are archived when their archive age reaches 10 minutes. After archive copy 1 is made, the disk cache space is released.

```
ex_set . -group images  
1 -release 10m
```

## -norelease Directive: Delaying Disk Space Release

The `-norelease` option prevents the automatic release of disk cache until all copies marked with `-norelease` are made. The `-norelease` directive makes the archive set eligible to be released after all copies have been archived, but the files are not released until the releaser is invoked and selects them as release candidates. This directive has the following format:

```
-norelease
```

Using the `-norelease` directive on a single copy has no effect on automatic releasing.

**EXAMPLE 11-3** archiver.cmd File Using the -norelease Directive

The following example specifies an archive set named `vault_tapes`. Two copies are created and then the disk cache associated with this archive set is released.

```
vault_tapes
1 -norelease 10m
2 -norelease 30d
```

## Using -release and -norelease Together

To make sure that disk space is released immediately after all copies of an archive set have been archived, use the `-release` and `-norelease` directives together. The combination of `-release` and `-norelease` causes the archiver to release the disk space immediately after all copies having this combination are made, rather than waiting for the releaser to be invoked.

## Setting the Archive Age

Change the timing for archiving files by specifying the archive age. Specify the time with a suffix character such as `h` for hours or `m` for minutes, as shown in [Table 12-1](#).

**EXAMPLE 11-4** archiver.cmd File That Specifies the Archive Age

In the following example, the files in directory `data` are archived when their archive age reaches one hour.

```
ex_set data
1 1h
```

## Unarchiving Automatically

If you specify more than one archive copy of a file, you can unarchive all but one of the copies automatically. You can do this when the files are archived to various media using various archive ages.

**EXAMPLE 11-5** archiver.cmd File that Specifies the Unarchive Age

The following example shows the directive that specifies the unarchive age. The first copy of the files in the path `home/users` is archived six minutes after modification. When the files are 10 weeks old, the archiver creates the second and third archive copies and unarchives the first copy.

```
ex_set home/users
1 6m 10w
2 10w
3 10w
```

**EXAMPLE 11-5** archiver.cmd File that Specifies the Unarchive Age (Continued)

For more ways to control unarchiving, see [Chapter 12, “Archive Set Directives \(archiver.cmd\).”](#)

## Specifying More Than One Copy for Metadata

If more than one copy of metadata is required, place copy definitions in the archiver.cmd file immediately after the fs= directive.

**EXAMPLE 11-6** archiver.cmd File that Specifies Multiple Metadata Copies

In this example, one copy of the metadata for the samfs7 file system is made after 4 hours and a second copy is made after 12 hours.

```
fs = samfs7
1 4h
2 12h
```

File system metadata includes path names in the file system. If you have frequent changes to directories, the new path names cause the creation of new archive copies and result in frequent loading of the volumes specified for metadata.

# Archive Set Directives (archiver.cmd)

---

This chapter provides information about the archive set directives:

- “About Archive Set Directives” on page 137
- “Archive Set Copy Parameters” on page 144
- “VSN Association Directives” on page 156
- “VSN Pools Directives” on page 157

## About Archive Set Directives

The following archive sets are available by default:

- Reserved archive set: `no_archive` and `allsets`.
  - The `no_archive` archive set is defined by default. Files assigned to this archive set are never archived. For example, files in a temporary directory can be assigned to the `no_archive` archive set.
  - The `allsets` archive set defines parameters that apply to all archive sets.
- Each file system has a default archive set with the same name that cannot be changed. These archive sets are reserved for control structure information. For each file system, both the metadata and data files are archived. The file system archive set includes the directory and link information and any files that are not included in another archive set.

By default, files are archived as part of the archive set named for the file system. However, you can create archive sets for files that share common criteria such as size, ownership, group, or directory location. If a file does not match one of the specified archive sets, it is archived as part of the default archive set. A file in the file system can be a member of only one archive set. All files in an archive set are copied to the volumes associated with that archive set.

Archive files are compatible with the standard UNIX `tar` format for data compatibility with the Oracle Solaris OS and other UNIX systems. If a complete loss of your SAM-QFS environment occurs, the `tar` format allows file recovery using standard UNIX tools and commands.

The characteristics of archive set are controlled by the archive set directives. These directives are arranged in the following categories:

- The *Assignment directive* defines archive sets: the destination of the archive copy, how long the copy is kept archived, and how long the software waits before archiving the data
- *Copy parameters* define how each archive set is archived: The archiving process copies the data necessary for file system operations, including directories, symbolic links, the index of segmented files, and archive media information.
- *VSN association directives* assign volumes to archive sets.
- *VSN pools directives* define a collection of volumes.

You can create archive sets either by editing the `archiver.cmd` file as described in the following sections, or by using the SAM-QFS Manager software. In the SAM-QFS Manager, an *archive policy* defines an archive set. For more information, see the SAM-QFS Manager online help.

## Archive Set Assignment Directive

The archive set assignment directive selects files according to shared characteristics to include in archive sets. Each archive set assignment directive has the following format:

```
archive-set-name path [search-criterion1 search-criterion2 \
... ] [file-attribute1 file-attribute2 ...]
```

Argument	Definition
<i>archive-set-name</i>	<p>A site-defined name for the archive set. A best practice is to define a name that identifies the common characteristics of the files belonging to the archive set. The name has the following requirements:</p> <ul style="list-style-type: none"> <li>▪ Maximum of 29 characters</li> <li>▪ Uppercase and lowercase alphabetic characters, numbers 0-9, and the underscore character (<code>_</code>).</li> <li>▪ No other special characters or spaces are allowed.</li> <li>▪ The first character must be a letter.</li> <li>▪ You cannot create an archive set with the name of one of the reserved archive sets, <code>no_archive</code> or <code>allsets</code> To prevent archiving of the files in an archive set, specify the name as <code>no_archive</code></li> </ul>
<i>path</i>	<p>Specify the path relative to the mount point of the file system so that files in the directory specified by <i>path</i> and its subdirectories are included in this archive set. To include all of the files in a file system, use a period (<code>.</code>). A leading slash (<code>/</code>) is not allowed in the path.</p>

Argument	Definition
<i>search-criterion1</i> <i>search-criterion2</i>	Zero, one, or more search criteria can be specified to restrict the archive set to files that meet the criteria. Search criteria include file age, file size, file ownership, and file name.
<i>file-attribute1</i> <i>file-attribute2</i>	Zero, one, or more file attributes can be specified. These file attributes are set for files as the <code>sam-arfind</code> process scans a file system during archiving.

**EXAMPLE 12-1** Archive set assignment directives.

In this example, all files in the user account `hmk` are archived in a separate archive set. All files over 1 MB in size in the directories in the `xray` group are archived in the `datafiles` archive set. All other files are archived in the `system` archive set.

```
hmk_files net/home/hmk -user hmk
datafiles xray_group/data -size 1M
system .
```

The following example shows directives that prevent archiving of files in a `tmp` directory at any level and regardless of the directory in which the `tmp` directory resides within the file system.

```
fs = samfs1
no_archive tmp
no_archive . -name .*/tmp/
```

## File Age *search-criterion* Arguments: `-access` and `-nftv`

To use the last time a file was opened to define assignment to an archive set, use the `-access age` characteristic as one of the *search-criterion* arguments.

This characteristic causes files that have not been accessed within the value of *age* to be re-archived to different, less-expensive media. For *age*, specify an integer followed by one of the suffixes shown in the following table.

**TABLE 12-1** File Age Suffixes

Suffix	Definition
s	Seconds
m	Minutes
h	Hours
d	Days
w	Weeks
y	Years

When determining age, the software validates the access and modification times for files to ensure that these times are greater than or equal to the file creation time, and less than or equal to the time at which the file is examined. For files that have been migrated into a directory, this validation might not result in the desired behavior. Use the `-nftv` (no file time validation) parameter in these situations to prevent the validation of file access and modification times.

## File Age *search-criterion* Arguments: `-after`

Use the `-after` *date-time* characteristic to include files that have been modified or created recently into the same archive set. Only files created or modified after the date indicated are included in the archive set. Specify the date and time in the following format:

```
YYYY-MM-DD[Thh:mm>:ss][Z]
```

If the time is not specified, the default time is 00:00:00. If the Z is included, the time is Coordinated Universal Time (UTC). If the Z is not included, the time is local time.

## File Size *search-criterion* Arguments: `-minsize` and `-maxsize`

Use the `-minsize` *size* and `-maxsize` *size* characteristics to restrict membership in an archive set to those over or under a specified size. For *size*, specify an integer followed by one of the letters shown in the following table.

TABLE 12-2 `-minsize` and `-maxsize` *size* Suffixes

Letter	Definition
b	Bytes
k	Kilobytes
M	Megabytes
G	Gigabytes
T	Terabytes
P	Petabytes
E	Exabytes

**EXAMPLE 12-2** Using the `-minsize` and `-maxsize` Characteristics

This example specifies that all files of at least 500 kilobytes but less than 100 megabytes belong to the archive set `big_files`. Files bigger than 100 megabytes belong to the archive set `huge_files`.

EXAMPLE 12-2 Using the `-minsize` and `-maxsize` Characteristics (Continued)

```
big_files . -minsize 500k -maxsize 100M
huge_files . -minsize 100M
```

## Owner and Group *search-criterion* Arguments: `-user` and `-group`

To restrict membership in an archive group to ownership and group affiliation, use the `-user name` and `-group name` characteristics.

EXAMPLE 12-3 Using the `-user` and `-group` Directive

In the following example, all files belonging to user `sysadmin` belong to archive set `adm_set`, and all files with the group name of `marketing` are in the archive set `mktng_set`.

```
adm_set . -user sysadmin
mktng_set . -group marketing
```

## File Name *search-criterion* Arguments Using Pattern Matching: `-name regex`

To specify that file names are used for assignment to an archive set, use `-name regex` characteristic, which specifies that any complete path matching the regular expression *regex* is to be a member of the archive set.

All files beneath the selected directory (with their specified paths relative to the mount point of the file system) go through pattern matching. Therefore, you can specify patterns in the `-name regex` field to match both file names and path names.

The *regex* argument follows the conventions outlined in the `regex(5)` man page. Regular expressions do not follow the same conventions as UNIX wildcards.

EXAMPLE 12-4 `-name` Argument

The following directive restricts files in the archive set `images` to those files ending with `.gif`:

```
images . -name .gif$
```

The following directive selects files that start with the characters `GEO` for the `satellite` archive set:

```
satellite . -name /GEO
```

The following directive prevents any file ending with `.o` from being archived:

**EXAMPLE 12-4** -name Argument (Continued)

```
no_archive . -name .o$
```

**EXAMPLE 12-5** Pattern Matching With Regular Expression

The archive set assignment directive in the following example does not archive `fred.*` in the user directories or subdirectories.

```
# File selections.
fs = samfs1
1 ls
2 ls
no_archive share/marketing -name fred.
```

As a result, the following files are not archived:

```
/sam1/share/marketing/fred.*
/sam1/share/marketing/first_user/fred.*
/sam1/share/marketing/first_user/first_user_sub/fred.*
```

The following files are archived:

```
/sam1/fred.anything
/sam1/share/fred.*
/sam1/testdir/fred.*
/sam1/testdir/share/fred.*
/sam1/testdir/share/marketing/fred.*
/sam1/testdir/share/marketing/second_user/fred.*
```

**EXAMPLE 12-6** Pattern Matching With Regular Expression

The archive set assignment directive in the following example does not archive `fred.*` in the user home directories but it does archive `fred.*` in the user subdirectories and in the directory `share/marketing`. In this case, a user home directory is the part of the string from `share/marketing/` until the next slash character `/`.

```
# File selections.
fs = samfs1
1 ls
2 ls
no_archive share/marketing -name ^share/marketing/[^/]*fred.
```

The following files are not archived:

```
/sam1/share/marketing/first_user/fred.*
```

The following files are archived:

```
/sam1/share/fred.*
/sam1/share/marketing/fred.*
/sam1/share/marketing/first_user/first_user_sub/fred.*
```

**EXAMPLE 12-6** Pattern Matching With Regular Expression (Continued)

```

/sam1/fred.*
/sam1/testdir/fred.*
/sam1/testdir/share/fred.*
/sam1/testdir/share/marketing/fred.*
/sam1/testdir/share/marketing/second_user/fred.*
/sam1/testdir/share/marketing/second_user/sec_user_sub/fred.*

```

## Release and Stage *file-attributes* Argument: -release and -stage

You can set the release and stage attributes associated with files within an archive set by using the `-release` and `-stage` options, respectively. Both of these settings override any existing stage or release attributes.

The `-stage` option has the following format:

`-stage attribute`

Attribute	Definition
a	Stage the files in this archive set associatively.
d	Reset to default.
n	Never stage the files in this archive set.

**EXAMPLE 12-7** Archive Sets and File Attributes

The following example shows how you can use file name specifications and file attributes to partially release Macintosh resource directories.

```
MACS . -name .*/.rscs/ -release p
```

## Membership Conflicts in Archive Sets

When the selection of a file for inclusion in an archive set is ambiguous, the archiver uses the following rules:

- The membership definition occurring first in the archive set is chosen. Place the most restrictive assignment directives early in the `archiver.cmd` file.
- Membership definitions local to a file system are chosen before any global definitions.
- A membership definition that exactly duplicates a previous definition is noted as an error.

The archiver evaluates the file system-specific directives before evaluating the global directives. Therefore, files can be assigned to a local archive set (including the `no_archive` archive set) instead of being assigned to a global archive. This result has implications for global archive set assignments such as `no_archive`.

**EXAMPLE 12-8** `archiver.cmd` File With Membership Conflicts

In the following example, the administrator did not intend to archive any of the `.o` files across both file systems. However, because the local archive set assignment `allfiles` is evaluated before the global archive set assignment `no_archive`, the `.o` files in the `samfs1` and `samfs2` file systems are archived.

```
no_archive . -name *.o$
fs = samfs1
allfiles .
fs = samfs2
allfiles .
```

**EXAMPLE 12-9** `archiver.cmd` File Without Membership Conflicts

The following example shows the directives to use to ensure that no `.o` files are archived in the two file systems.

```
fs = samfs1
no_archive . -name *.o$
allfiles .
fs = samfs2
no_archive . -name *.o$
allfiles .
```

## Archive Set Copy Parameters

The archive set copy parameters define how each archive set is archived: the data files, directories, symbolic links, the index of segmented files, and archive media information.

The archive set copy parameters section of the `archiver.cmd` file begins with the `params` directive and ends with the `endparams` directive.

The following example shows the format for copy parameters for an archive set.

```
params
archive-set-name.copy-number[R] [-param1 -param2 ...]
.
.
.
endparams
```

TABLE 12-3 Arguments for the Archive Set Copy Parameters

Argument	Definition
<i>archive-set-name</i>	A site-defined name for the archive set. A best practice is to define a name that identifies the common characteristics of the files belonging to the archive set. The name has the following requirements: <ul style="list-style-type: none"> <li>■ Maximum of 29 characters</li> <li>■ Uppercase and lowercase alphabetic characters, numbers 0-9, and the underscore character (_).</li> <li>■ No other special characters or spaces are allowed.</li> <li>■ The first character must be a letter.</li> </ul>
<i>copy-number</i>	An integer that defines the archive copy number: 1, 2, 3, or 4.
R	Specifies that the parameters being defined are for re-archived copies of this archive set. For example, you can use the R and specify VSNs in the <code>-param1</code> argument to direct re-archived copies to specific volumes.
<i>-param1</i> <i>-param2</i>	One or more parameters such as maximum size, buffer size, number of drives, and so on. The following subsections describe the parameters that can be specified between the <code>params</code> and <code>endparams</code> directives.

To set default directives for all archive sets, specify directives for the archive set `allsets` archive set. The `allsets` directives must precede the directives for archive set copies because parameters set for individual archive set copies override parameters set for the `allsets` directive. For more information about the `allsets` archive set, see [“archiver.cmd\(4\)” in Sun QFS and Sun Storage Archive Manager Reference Manual](#).

You can specify archive set copy parameters by editing the `archiver.cmd` file as described in the following sections or by using the SAM-QFS Manager software. For more information, see the SAM-QFS Manager online help.

The following sections describe all archive set processing parameters with the exception of disk archiving parameters. For information about disk archiving parameters, see [“About Disk Archiving” on page 114](#).

## Controlling the Size of Archive Files: `-archmax` Parameter

The `-archmax` parameter sets the maximum file size for an archive set. This parameter has the following format:

```
-archmax target-size
```

This parameter is very similar to the `archmax` global directive. For information about that directive and the values to enter for *target-size*, see [“Controlling the Size of Archive Files: -archmax Parameter” on page 145](#).

## Setting the Archiver Buffer Size: -bufsize Parameter

By default, a file being archived is stored in memory in a buffer of a default size for the media type before being written to archive media. Use the `-bufsize` directive to specify a buffer size. A custom size can improve performance. This parameter has the following format:

```
-bufsize=buffer-size
```

The default buffer size is 4, indicating that the actual buffer size is 4 multiplied by the `dev_blksize` value for the media type. Specify a number from 2 to 32. The `dev_blksize` value is specified in the `defaults.conf` file.

For more information about this file, see [“defaults.conf\(4\)” in Sun QFS and Sun Storage Archive Manager Reference Manual](#).

**EXAMPLE 12-10** Buffer Size: -bufsize

```
myset.1 -bufsize=6
```

This parameter is similar to the `bufsize=media buffer-size` global directive. For more information about that directive, see [“Setting the Archiver Buffer Size: -bufsize Parameter” on page 146](#).

## Specifying the Number of Drives for an Archive Request: -drivemax, -drivemin, and -drives

By default, the archiver uses one media drive to archive the files of one archive set. When an archive set has many files or large files, using more than one drive is advantageous. In addition, if the drives in your automated library operate at different speeds, use of multiple drives can balance these variations and increase archiving efficiency. The drive directives have the following formats:

```
-drivemax max-size
-drivemin min-size
-drives number
```

Argument	Definition
<i>max-size</i>	The maximum amount of data to be archived using one drive.

Argument	Definition
<i>min-size</i>	The minimum amount of data to be archived using one drive. The default is the <code>-archmax target-size</code> value (if specified) or the default value for the media type. If you specify the <code>-drivemin min-size</code> directive, the SAM-QFS software uses multiple drives only when enough activity occurs to warrant it. As a guideline, set <i>min-size</i> to be large enough to cause the transfer time to be significantly longer than the cartridge change time (load, position, unload).
<i>number</i>	The number of drives to be used for archiving this archive set. The default is 1.

An archive request is evaluated against the parameters that are specified, as follows:

- If an archive request is less than the value of *min-size*, only one drive is used to write an archive request.
- If an archive request is larger than the value of *min-size*, the archive request is evaluated against *min-size* and the appropriate number of drives is scheduled up to the full number of drives specified.
- If the value of *min-size* is 0, an attempt is made to split the archive request among the full number of drives specified.

When you use the `-drives` parameter, multiple drives are used only if data that is more than the value of *min-size* is to be archived. The number of drives to be used in parallel is the lesser of the following two values:

- The size of the archive request divided by the value of *min-size*
- The number of drives specified by the `-drives` parameter

Use the `-drivemin` and `-drives` parameters when you want to divide an archive request among drives but do not want to have all the drives busy with small archive requests. This situation can occur with very large files.

To set these parameters, consider file creation rates, the number of drives, the time it takes to load and unload drives, and drive transfer rates. For example, a site splits an archive set named `bigfiles` across five drives. This archive set could be split as shown in the following table.

Archive Set Size	Number of Drives
< 20 gigabytes	1
> 20 gigabytes to < 30 gigabytes	2
> 30 gigabytes to < 40 gigabytes	3
> 40 gigabytes to < 50 gigabytes	4
> 50 gigabytes	5

**EXAMPLE 12-11** Directives Used to Split an Archive Request Over Multiple Drives

The following example shows how to split the archive requests of 10 GB or more over five drives.

```
params
bigfiles.1 -drives 5 -drivemin 10G
endparams
```

In addition, the following line ensures that two drives are used to archive the files when the total size of the files in archive set `huge_files.2` is equal to or greater than two times `drivemin` for the media.

```
huge_files.2 -drives 2
```

## Maximizing Space on a Volume: `-fillvsns` Parameter

By default, the archiver selects a volume with enough space for all files when it writes an archive copy. This action results in volumes not being filled to capacity. When `-fillvsns` is specified, the archiver separates the archive request into smaller groups and can use different volumes.

## Specifying Archive Buffer Locks: `-lock` Parameter

By default, a file is stored in a buffer before being written to archive media. If direct I/O is enabled, you can use the `-lock` parameter to lock this buffer. The `-lock` parameter indicates that the archiver must use locked buffers when making archive copies. If `-lock` is specified, the archiver sets file locks on the archive buffer in memory for the duration of the `sam-arcopy` operation. This action avoids paging of the buffer, and can improve performance.

This parameter has the following format:

```
-lock
```

Use the `-lock` parameter only on large systems with large amounts of memory. Insufficient memory can cause an out-of-memory condition.

The `-lock` parameter is effective only if direct I/O is enabled for the file. By default, `-lock` is not specified, and the file system sets locks on all direct I/O buffers, including those for archiving. For more information about enabling direct I/O, see “[setfa\(1\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*, “[sam\\_setfa\(3\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*, or `-O forcedirectio` option on “[mount\\_samfs\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

This parameter is similar to the `lock` argument to the `bufsize` global directive. For more information about this topic, see “[Setting the Archiver Buffer Size: `-bufsize` Parameter](#)” on page 146.

## Making Archive Copies of Offline Files: `-offline_copy` Parameter

A file is a candidate for being released after one archive copy is made. If the file is released and goes offline before any remaining archive copies are made, the archiver uses this parameter to determine the method use to make the other archive copies. When you specify the method, consider the number of drives available to the SAM-QFS system and the amount of disk cache available. This parameter has the following format:

`-offline_copy method`

<i>method</i> Value	Definition
none	Stages files as needed for each file before copying to the archive volume. Default.
direct	Copies files directly from the offline volume to the archive volume without using the cache. This method assumes that the source volume and the destination volume are different volumes and that two drives are available. Raise the value of the <code>stage_n_window</code> mount option to a value that is greater than its default of 256 kilobytes. For more information about mount options, see “ <code>mount_samfs(1M)</code> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
stageahead	Stages the next archive file while writing an archive file to its destination.
stageall	Stages all files to disk cache before archiving. This method uses one drive and assumes that room is available on disk cache for all files.

## Specifying Recycling

Use the recycling process to reclaim space on archive volumes in use by expired archive images. By default, no recycling occurs. You must specify directives in both the `archiver.cmd` file and the `recycler.cmd` file. For more information, see [Chapter 16, “Configuring the Recycler.”](#)

## Sorting Archive Files: `-sort` and `-rsort` Parameters

By default, files in an archive set are sorted by path before they are archived. You can specify that files are sorted by age, priority, or size, or not sorted (none). Only one sort method can be used per archive set.

You can use `-rsort` to reverse the order of sorting specified by method.

### EXAMPLE 12-12 Sorting Files in an Archive Set

The first example line sorts the archive set copy `cardiac.2` by the age of the file, oldest to youngest. The second line forces the archive set copy `cat scans` to be sorted by the size of the file, in reverse order, largest to smallest.

**EXAMPLE 12-12** Sorting Files in an Archive Set *(Continued)*

size.

```
cardiac.2 -sort age
catscans.3 -rsort size
```

## Controlling Unarchiving

Unarchiving is the process by which archive entries for files or directories are deleted. A file is unarchived based on the time since it was last accessed. This distinction means data that is accessed frequently can be stored on fast media such as disk and infrequently accessed data can be stored on tape. By default, files are never unarchived.

**EXAMPLE 12-13** Directives to Control Unarchiving

The following example directives specify that the `arset1` file remains on disk all the time, even if it is older than 60 days. The Copy 1 information is removed when the file has not been accessed for 60 days. After the Copy 1 information is removed, any access request is fulfilled by Copy 2 and is read from tape. The archiver makes a new Copy 1 on disk and the 60-day cycle starts again.

```
arset1 dir1
1 10m 60d
2 10m
3 10m
vsns
arset1.1 mo OPT00[0-9]
arset1.2 lt DLTA0[0-9]
arset1.3 lt DLTB0[0-9]
```

The example directives meet the requirements for both access and archiving in the following scenario.

A patient is in the hospital for four weeks. During this time, all the patient's files are on fast media and the data is being accessed frequently. This is Copy 1 (copy 1=mo). After two weeks, the patient is discharged from the hospital. The patient files are accessed less frequently and then not at all. When no data has been accessed for this patient 60 days, the Copy 1 entry in the inode is unarchived. Only Copy 2 and Copy 3 entries are available. The volume of fast media can now be recycled and used by current patients without having to increase the disk library. However, six months later, the patient returns to the hospital. The first access of the patient's file is from tape (Copy 2). To get the data on fast media, the archiver creates a new Copy 1 on disk, ready for new information.

## Controlling How Archive Files Are Written: -tapenonstop Parameter

By default, the archiver writes a tape mark, an end of file (EOF) label, and two more tape marks between archive files. When the next archive file is started, the driver backs up to the position after the first tape mark, causing a loss of performance. The `-tapenonstop` parameter directs the archiver to write only the initial tape mark. In addition, the archiver enters the archive information at the end of the copy operation.

For more information about the `-tapenonstop` parameter, see “[archiver.cmd\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Reserving Volumes: -reserve Parameter

By default, the archiver writes archive set copies to any volume specified by a regular expression as described in the volume associations section of the `archiver.cmd` file. However, if you require that an archive set volume contains files from only one archive set, you can reserve a volume for this purpose.

Note the following guidelines:

- A site that uses reserved volumes incurs more cartridge loads and unloads.
- A site that uses reserved volumes for file systems that have many directories of a few small files causes the archiver to write many small archive files to each reserved volume. These small archive files, each with its own tar header, slow performance.

The `-reserve` parameter specifies a volume for use by an archive set and gives it a unique identifier that ties the archive set to the volume. The volume identifier is not assigned to any other archive set copy, even if a regular expression matches it. The format for the `-reserve` parameter is as follows:

```
-reserve keyword
```

The value of *keyword* depends on the form you are using. You can specify one, two, or all three forms in combination.

Form	keyword	Reserved Name Examples	Notes
Archive Set	set	users.1// {}Data.1//	

Form	keyword	Reserved Name Examples	Notes
Owner	dir	proj.1/p105/ {}proj.1/p104/	The dir, user, and group keywords, which are mutually exclusive, specify the owner component in the reserved name. The dir keyword uses the directory path component immediately following the path specification of the archive set definition.
	user	users.1/user5/ {}users.1/user4/	
	group	data.1/engineering/	
File System	fs	proj.1/p103/samfs1{ }proj.1/p104/samfs1	The fs keyword specifies the file system component in the reserved name.

**EXAMPLE 12-14** Reserving Volumes by Archive Set

The following example specifies that the allsets archive set reserves a volume for each archive set.

```
params
allsets -reserve set
endparams
```

**EXAMPLE 12-15** Reserved Volume Name

The following example specifies that the arset.1 archive set reserves a volume and the volume identifier is created from an archive set, a group, and the file system.

```
params
arset.1 -reserve set -reserve group -reserve fs
endparams
```

Information about reserved volumes is stored in the library catalog. The lines in the library catalog that describe reserved volumes begin with #R characters and show the media type, the VSN, the reserve information, and the reservation date and time. The information also includes the archive set component, path name component, and file system component, separated by two slashes (/).

---

**Note** – The slash characters do not indicate a path name. They serve to separate the components of a reserved name.

---

**EXAMPLE 12-16** Library Catalog Showing Reserved Volumes

The lines have been truncated to fit on the page.

```
6 00071 00071 lt 0xe8fe 12 9971464 1352412 0x6a000000 131072 0x
# -il-o-b----- 05/24/00 13:50:02 12/31/69 18:00:00 07/13/01 14:03:00
```

**EXAMPLE 12-16** Library Catalog Showing Reserved Volumes (Continued)

```
#R lt 00071 arset0.3// 2001/03/19 18:27:31
10 ST0001 NO_BAR_CODE lt 0x2741 9 9968052 8537448 0x68000000 1310
# -il-o----- 05/07/00 15:30:29 12/31/69 18:00:00 04/13/01 13:46:54
#R lt ST0001 hgm1.1// 2001/03/20 17:53:06
16 SLOT22 NO_BAR_CODE lt 0x76ba 6 9972252 9972252 0x68000000 1310
# -il-o----- 06/06/00 16:03:05 12/31/69 18:00:00 07/12/01 11:02:05
#R lt SLOT22 arset0.2// 2001/03/02 12:11:25
```

One or more of the reserve information fields can be empty, depending on the options defined in the `archiver.cmd` file. A reservation line is appended to the file for each volume that is reserved for an archive set during archiving.

You can also use the `reserve` and `unreserve` commands to reserve and unreserve volumes. For more information about these commands, see “[reserve\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* and “[unreserve\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

A volume is unreserved when it is relabeled because the archive data has been effectively erased.

You can display the reserve information by using the `samu` utility's `v display` or by using the `archiver` or `dump_cat` command in one of the formats shown in the following example:

```
# archiver -lv
# dump_cat -V _catalog-name_
```

## Setting Archive Priorities: `-priority` Parameter

Archive-enabled file systems provide priorities for archiving files. Each file is assigned a priority computed from properties of the file and priority multipliers that can be set for each archive set in the `archiver.cmd` file. Properties include online/offline, age, number of copies made, and size.

By default, the files in an archive request are not sorted, and all property multipliers are zero. The result is that files are archived in first-found, first-archived order. To change the order in which files are archived, set priorities and sort methods. Examples of new priorities include:

- Select the `priority` sort method to archive files within an archive request in priority order.
- Change the `archive_loaded` priority to reduce media loads.
- Change the `offline` priority to cause online files to be archived before offline files.
- Change the `copy#` priorities to make archive copies in copy order.

TABLE 12-4 Archive Priorities

Archive Priority	Definition
-priority age <i>value</i>	Archive age property multiplier
-priority archive_immediate <i>value</i>	Archive immediate property multiplier
-priority archive_overflow <i>value</i>	Multiple archive volumes property multiplier
-priority archive_loaded <i>value</i>	Archive volume loaded property multiplier
-priority copies <i>value</i>	Copies-made property multiplier
-priority copy1 <i>value</i>	Copy 1 property multiplier
-priority copy2 <i>value</i>	Copy 2 property multiplier
-priority copy3 <i>value</i>	Copy 3 property multiplier
-priority copy4 <i>value</i>	Copy 4 property multiplier
-priority offline <i>value</i>	File offline property multiplier
-priority queuwait <i>value</i>	Queue wait property multiplier
-priority rearchive <i>value</i>	Rearchive property multiplier
-priority reqrelease <i>value</i>	Reqrelease property multiplier
-priority size <i>value</i>	File-size property multiplier
-priority stage_loaded <i>value</i>	Stage volume loaded property multiplier
-priority stage_overflow <i>value</i>	Multiple stage volumes property multiplier

For *value*, specify a floating-point number in the following range:

`-3.400000000E+38 <= _value_ <= 3.402823466E+38`

For more information about priorities, see “archiver(1M)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* and “archiver.cmd(4)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Scheduling Archiving: -startage, -startcount, and -startsize Parameters

As the archiver scans a file system, it identifies files to be archived. Files that are recognized as candidates for archiving are placed in a list known as an *archive request*. At the end of the file

system scan, the system schedules the archive request for archiving. The `-startage`, `-startcount`, and `-startsize` archive set parameters control the archiving workload and ensure the timely archival of files.

TABLE 12-5 `-startage`, `-startcount`, and `-startsize` Directives

Directive	Definition
<code>-startage</code> <i>time</i>	The amount of time that can elapse between the first file in a scan being marked for inclusion in an archive request and the start of archiving. Specify a time in the format described in “ <a href="#">Setting the Archive Age</a> ” on page 135. If this variable is not set, the <code>interval</code> directive is used.
<code>-startcount</code> <i>count</i>	The number of files to be included in an archive request. When the number of files in the archive request reaches the this value, archiving begins. By default, <i>count</i> is not set.
<code>-startsize</code> <i>size</i>	The minimum total size, in bytes, of all files to be archived in an archive request. Archiving work is accumulated, and archiving begins when the total size of the files reaches the this value. By default, <i>size</i> is not set.

The `examine=method` directive and the `interval=time` directives interact with the `-startage`, `-startcount`, and `-startsize` directives. The `-startage`, `-startcount`, and `-startsize` directives optimally balance archive timeliness and archive work done. These values override the `examine=method` specification, if any.

The `-startage`, `-startcount`, and `-startsize` directives can be specified for each archive copy. If more than one of these directives is specified, the first condition encountered starts the archive operation. If none of these directives is specified, the archive request is scheduled based on the `examine=method` directive:

- If `examine=noscan`, the default values of the directives are used: `startage` 10 minutes, `startcount` 10,000, and `startsize` 10 gigabytes. The archive request is scheduled according to the value of the `interval=` directive after the first file is entered in the archive request. This method is continuous archiving and is the default method.
- If `examine=scan|scaninodes|scandirs`, the archive request is scheduled for archiving after the file system scan.

The `archiver.cmd(4)` man page provides examples that show how to use these directives.

## VSN Association Directives

The VSN associations section of the `archiver.cmd` file assigns volumes to archive sets. This section starts with a `vsns` directive and ends with an `endvsns` directive.

VSN associations can also be configured with the SAM-QFS Manager software. See the SAM-QFS Manager online help for more information.

Collections of volumes are assigned to archive sets by directives of the following form:

```
archive-set-name.copy-num media-type vsn-expr ... [-pool /
vsn-pool-name ...]
```

An association requires at least three fields: *archive-set-name* and *copy-num*, *media-type*, and at least one volume. The *archive-set-name* and *copy-num* values are connected by a period (.).

Argument	Definition
<i>archive-set-name</i>	A site-defined name for the archive set.
<i>copy-num</i>	A digit followed by one or more arguments that specify archive characteristics for that copy. Archive copy directives begin with a digit. This digit (1, 2, 3, or 4) is the copy number.
<i>media-type</i>	The media type. For a list of valid media types, see “ <a href="#">mcf(4)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>vsn-expr</i>	A regular expression. See the <code>regex(5)</code> man page.
<code>-pool vsn-pool-name</code>	A named collection of VSNs.

---

**Note** – If your SAM-QFS environment is configured to recycle by archive set, do not assign a VSN to more than one archive set.

---

### EXAMPLE 12-17 VSN Specifications on Multiple Lines

The following example shows two lines of VSN specifications.

```
vsns
set.1 lt VSN001 VSN002 VSN003 VSN004 VSN005
set.1 lt VSN006 VSN007 VSN008 VSN009 VSN010
endvsns
```

### EXAMPLE 12-18 VSN Specifications With a Continued Line

The following example shows a VSN specification that uses a backslash character “\” to continue a line onto a subsequent line.

**EXAMPLE 12-18** VSN Specifications With a Continued Line *(Continued)*

```

vsn
set.1 lt VSN001 VSN002 VSN003 VSN004 VSN005 \
VSN006 VSN007 VSN008 VSN009 VSN010
endvsn

```

**EXAMPLE 12-19** VSN Specifications With Shorthand Notation

The following example specifies VSNs using a regular expression in a shorthand notation.

```

vsn
set.1 lt VSN0[1-9] VSN10
endvsn

```

When the archiver needs volumes for the archive set, it examines each volume of the selected media type in all automated libraries and manually mounted drives to determine whether the volume satisfies any VSN expression. It selects the first volume that matches an expression that contains enough space for the archive copy operation. For example:

- The following directive specifies that files belonging to archive set `ex_set` for copy 1 be copied to media type `mo` using any of the 20 volumes with the names `opt ic20` through `opt ic39`.

```
ex_set.1 mo optic[2-3][0-9]
```

- The following directive specifies that files belonging to archive set `ex_set` for copy 2 be copied to media type `lt` using any volume beginning with `TAPE`:

```
ex_set.2 lt ^TAPE
```

---

**Note** – Make sure you assign volumes to the archive set used for the file system's metadata when setting up the `archiver.cmd` file. Each file system has an archive set with the same name as the file system. For more information about preserving metadata, see [“samfsdump\(1M\)” in Sun QFS and Sun Storage Archive Manager Reference Manual](#).

---

## VSN Pools Directives

The VSN pools section of the `archiver.cmd` file starts with a `vsnpools` directive and ends either with an `endvsnpools` directive or with the end of the `archiver.cmd` file. This section names a collection of volumes.

VSN pools can also be configured with the SAM-QFS Manager software. See the SAM-QFS Manager online help for more information.

A VSN pool is a named collection of volumes. VSN pools are useful for defining volumes that can be available to an archive set. As such, VSN pools provide a useful buffer for assigning volumes and reserving volumes to archive sets. You can use VSN pools to define separate

groups of volumes by departments within an organization, by users within a group, by data type, and according to other convenient groupings.

If a volume is reserved, it is no longer available to the pool in which it originated. Therefore, the number of volumes within a named pool changes as volumes are used. You can view the VSN pools by issuing the `archiver` command in the following format:

```
# archiver -lv | more
```

The syntax of a VSN pool definition is as follows:

```
vsn-pool-name media-type vsn-expr
```

Argument	Definition
<i>vsn-pool-name</i>	The VSN pool.
<i>media-type</i>	The two-character media type. For a list of valid media types, see “ <a href="#">mcf(4)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>vsn-expr</i>	A regular expression. You can provide one or more <i>vsn-expr</i> arguments. See the <code>regcmp(3G)</code> man page.

The following example uses four VSN pools: `users_pool`, `data_pool`, `proj_pool`, and `scratch_pool`. A scratch pool is a set of volumes used when specific volumes in a VSN association are exhausted or when another VSN pool is exhausted. If one of the three specific pools is out of volumes, the archiver selects the scratch pool VSNs.

#### EXAMPLE 12-20 VSN Pools

The following example shows an `archiver.cmd` file that uses four VSN pools.

```
vsnpools
users_pool mo ^MO[0-9][0-9]
data_pool mo ^DA.*
scratch_pool mo ^SC[5-9][0-9]
proj_pool mo ^PR.*
endvsnpools
vsns
users.1 mo -pool users_pool -pool scratch_pool
data.1 mo -pool data_pool -pool scratch_pool
proj.1 mo -pool proj_pool -pool scratch_pool
endvsns
```

For more information about VSN associations, see “[VSN Association Directives](#)” on page 156.

# Data Integrity Validation in SAM-QFS

---

The Data Integrity Validation (DIV) feature provides end-to-end user data protection from the SAM disk cache to and from the tape archives.

## About DIV

DIV is supported on Oracle's StorageTek T10000C tape drives. When the DIV mode is enabled on the SAM-QFS file system for the archive copy, the SAM-QFS file system enables DIV in the `st` driver using the `mt DADP` command. The `st` driver then calculates a checksum (hardware or software generated depending on the server platform) for every logical block written by the archiver to the tape. The checksum is checked by the tape storage drive and it must match the data to be written to the tape. The StorageTek T10000C drive will put both the user data and the checksum on the media. If the checksum does not match, an error is returned to the archiver and the archiver does not mark the file as copy archived. The `st` driver retries the DIV errors during both the read and the write operation. If the errors are only temporary, then the `st` driver will successfully transfer the data without returning an error.

SAM-QFS uses the `st` driver to check the checksum for every logical block read by the stager from the tape. The checksum must match in order for data to be returned to the user. The stager will move to an alternate copy if an error is detected. DIV gives immediate feedback at the block level to guarantee that only good data is written to the tape and only good data is returned from the tape. The StorageTek T10000C drive enables a very fast user data verify option without having to transfer the data back to SAM-QFS. Because the checksum is stored on the media with the user data, it can be verified directly by the tape storage device.

SAM-QFS supports a new `tpverify` command for the StorageTek T10000C drive which will verify the data written to tape without staging the data. This check can be done periodically to verify that the data on the tape is correct. If the data is not correct, SAM-QFS stages the alternate copy and rearchives the file automatically. Also, the `tpverify` command will return errors if it detects incorrect data. Another archiver capability is `verify-after-write`, which causes the archiver to verify the entire file after it is written on tape. The `verify-after-write` capability ensures that the DIV Cyclic Redundancy Codes (CRC) is correctly stored on tape.

For the Sun SAM-Remote software, in order to use the DIV feature, the `div` settings in the `default.conf` file must be enabled for the SAM-Remote server as well as the client. For more information about the SAM-Remote Software, see [Chapter 18, “Using the Sun SAM-Remote Software.”](#)

## Minimum Hardware and Software Requirements for Using DIV in SAM-QFS

The minimum hardware and software requirements for enabling DIV in SAM-QFS are:

- StorageTek T10000C tape drives  
Minimum firmware release level is 1.53.315
- Oracle Solaris 11

## Configuring DIV in SAM-QFS

DIV can be enabled or disabled by changing the `div` values in the `/etc/opt/SUNWsamfs/default.conf` file. DIV is disabled by default. The following values can be set to enable or disable DIV in SAM-QFS:

- `div = off`  
Disables DIV. This is the default value.
- `div = on`  
Enables DIV for read, write, and verify operations.
- `div = verify`  
Enables DIV for read, write, and verify operations. Also, enables the archiver's verify after write mode.  
For more information, see “[defaults.conf\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

### ▼ How to Configure DIV in SAM-QFS

- 1 Use the `samd stop` command to stop all operations.

```
# samd stop
```

- 2 Set the `div` value in the `/etc/opt/SUNWsamfs/default.conf` file to either `on` or `verify`.

```
div = on | verify
```

**3 Propagate the configuration changes.**

```
# samd config
```

**4 Start all operations.**

```
# samd start
```

## Examples of Using DIV

The following examples show how to use the `samu` command to display the DIV settings in the `defaults.conf` file and to display various verify-after-write statuses.

For more information about displaying DIV settings, see “`samu(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

**EXAMPLE 13-1** Display the DIV Setting

Use the following syntax to display the DIV settings in the `defaults.conf` file:

```
# samu -L
div VERIFY
```

**EXAMPLE 13-2** Display Verify After Write Archiver Status

When DIV verify is enabled, the archiver can display the verifying status when the archive file is being verified. Use the following syntax to display the verify-after-write archive status:

```
# samu -a
sam-arcopy: qfs.arset1.2.344 ti.TKC960
Verifying archive file at position 1175
```

**EXAMPLE 13-3** Display Verify After Write Device Status

The `samu -s` command displays information about the tape drives that are performing the archive file verify operation. Use the following syntax to display the verify-after-write device status:

```
# samu -s
ti          91 on          /dev/rmt/4cbn          90 -l----oVr
Verify averaging 240.9 MB/s
```

You can use the `itemize -2 30` command to display the last verified time and position. For example:

```
# itemize -2 30
Robot VSN catalog: eq: 30 count: 2
slot access_time count use ty vsn lvtime lvpos
0 Jun 24 17:21 2 0% ti CET052 Jun 24 17:21 0
1 Jun 24 18:43 8 5% ti CET050 Jun 24 17:11 0x5ee
```

## Tape Verify (tpverify) Command

The `tpverify` command uses the SCSI `Verify` command with the DIV/DADP feature of the StorageTek T10000C tape drive to verify media.

The `tpverify` command loads the specified volume into a tape drive and verifies it. If the device is not specified, then the volume is loaded into an available drive in the media changer. The SAM-QFS file system chooses the drive into which the volume is loaded. You can cancel a verify operation if the resource is needed. If the `tpverify` command is canceled, then the last verified position (LVPos) is saved in the catalog as the starting position for the next verify operation.

---

**Note** – A verify operation on a tape drive that is in the “on” state can be cancelled by the SAM-QFS archiver or stager if the resource is needed. Use a drive in the “unavail” state to avoid the verify operation cancellation by SAM-QFS.

---

The `tpverify` command can be used with a non-DIV tape drive as long as the tape drive supports the SCSI `Verify` command in fixed block with a block count of either 0x1000, 0x100, or 0xff. For example, the HP LTO-4 tape drives support the SCSI `Verify` command and a block count of 0xff. The SCSI `Verify` errors for the non-DIV tape drives are handled the same way as the DIV media errors, in that the `tpverify` command sets bad media and terminates the verify run. Check the `/var/opt/SUNWsamfs/devlog/eqord` file for failure details.

The following table shows the options that are available for the `tpverify` command.

Options	Description
-a	Overrides the last verified position saved in the catalog to start the verify operation from the first archive file on media.
-c	Cancels a running <code>tpverify</code> command operation.
-w	Waits for the operation to complete before terminating it.

For more information about the tape verify command, see the `tpverify(1M)` man page.

## Tape Verify (tpverify) Command Return Values

The following table shows the return values for the `tpverify` command.

Return Values	Description
0	Success

---

Return Values	Description
249	Media error
250	Drive set to down
251	DIV error; bad media set
252	Drive needs cleaning
253	SAM-QFS canceled the verify operation
254	User canceled the verify operation
255	Verify failed

---

**EXAMPLE 13-4** Examples of Using the `tpverify` Command

The following examples show how to apply the `tpverify` command.

To load VSN CET050 into an available drive:

```
# tpverify ti.CET050
```

To load VSN CET048 into drive 31 and wait for the results:

```
# tpverify -w -a ti.CET048 31; echo $?
```

To cancel a verify run on VSN CET048 and wait for the results:

```
# tpverify -w -c ti.CET048 31; echo $?
```



# About Releasing

---

This chapter describes the releasing process and releaser operations.

Releasing is the process by which the releaser makes disk cache space available by identifying archived files and releasing their disk cache copy. This action makes room for other files to be created or staged from archive media. The releaser can release only archived files. A released file has no data on the disk cache.

The Sun Storage Archive Manager (SAM) software invokes the releaser process when a site-specified disk threshold is reached. As an alternative, you can use the `release` command to release a file's disk space immediately or to set releasing parameters for a file.

You can also specify that files are released immediately after archiving, that files are never released, or that files are partially released.

## Releaser Process

When file system utilization exceeds its configured high-water mark (HWM), the file system management software invokes the releaser, which does the following:

- Reads the `releaser.cmd` file and collects the directives that control the release process
- Scans the file system and collects information about each file
- Begins releasing files in priority order

A file system can contain thousands of files. Keeping track of the release priority for every file can be wasteful, because releasing only several large files might return the file system to its low-water mark (LWM). However, the releaser must examine the priority of each file or risk missing the best candidates for release. The releaser does this by identifying the first 10,000 candidates. It then discards subsequent candidates if they do not have a priority greater than the lowest-priority candidate among the first 10,000.

After the releaser has determined the priority of the first 10,000 candidates, it selects the files with the highest priority for release. After releasing each file, the releaser checks whether the file system cache utilization is below the low-water mark. If so, it stops releasing files. If not, it continues releasing the files in priority order.

If the releaser has released all 10,000 candidates and the file system is still above the low-water mark, it starts over and identifies 10,000 new candidates.

The releaser exits if it cannot find any viable candidates. This situation can occur, for example, if files do not yet have archive copies. In this case, the SAM software starts the releaser again after one minute has elapsed.

The high-water and low-water marks are set with the `high=percent` and `low=percent` file system mount options. For more information about these mount options, see [“mount\\_samfs\(1M\)”](#) in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Releaser Concepts

This section describes concepts that are basic to the releaser process:

- “Age” on page 166
- “Candidate” on page 166
- “Priority” on page 167
- “Weight” on page 167
- “Partial Release” on page 167

### Age

Age is the amount of time that elapsed from a given event to the present. A file's inode keeps track of the following times:

- Residence-change time
- Data-modified time
- Data-accessed time

You can view these times by using the `sls` command with the `-D` option. Each time has a corresponding age. For example, if it is 10:15 a.m., a file with a modify time of 10:10 a.m. has a data-modified age of five minutes. For more information, see [“sls\(1\)”](#) in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

### Candidate

A candidate is a file that is eligible to be released. A file is *not* a candidate under the following circumstances:

- The file is already offline.
- The file has not been archived.
- The archiver `.cmd` command file specifies the `-norelease` attribute for the file and the required copies have not yet been made.
- The file is marked as damaged.
- The file is not a directory, block, character-special file, or pipe.
- The archiver is staging the file to make an additional copy. The file becomes eligible for release after the archiver stages it.
- The age of the file is negative. This condition occurs for network file system (NFS) clients with inaccurate clock settings.
- The file is marked to never be released. You can use the `release -n` command to specify this.
- The file was staged at a time in the past that is less than the minimum residence time setting.
- The file was flagged for partial release, through the `release` command's `-p` option, and it is already partially released.
- The file is too small. Releasing it will not create much space.

## Priority

A priority is a numeric value that indicates the rank of a candidate file based on user-supplied weights that are applied to numeric attributes of that candidate. The overall priority is the sum of two types of priority: age priority and size priority. Candidate files with numerically larger priorities are released before candidates with numerically smaller priorities.

## Weight

A weight is a numeric value that biases the priority calculation to include file attributes in which you are interested and to exclude file attributes in which you are not interested. For example, if the size weight is set to 0, the size attribute of a file is excluded from the priority calculation. Weights are floating-point values from 0.0 to 1.0.

## Partial Release

With partial release, a beginning portion of the file remains in disk cache while the rest of the file is released. Partial release is useful with utilities such as `filemgr` that read the beginning of a file.

## About Partial Releasing and Partial Staging

Releasing and staging are complementary processes. Files can be completely released from online disk cache after they are archived, or a site can specify that the beginning of a file (the stub) remain in disk cache while the remainder of the file is released. Partially releasing a file provides immediate access to data in the file stub without requiring that the file be staged. You can specify both the default partial release size and the maximum size of the stub to remain online when a file system is mounted.

You can set the through the values for partial releasing and staging using the mount command or in the SAM-QFS Manager. See the SAM-QFS Manager online help for more information.

The mount command options are as follows:

- `-o partial=n` option - Sets the default size (*n*) of a file stub to remain online. The `-o partial=n` setting must be less than or equal to the `-o maxpartial=n` setting.
- `-o maxpartial=n` option - Sets the maximum size (*n*) of a file stub to remain online.

You can specify the default stub size for a file by specifying the `--p` option on the `release` command or the `-p` option on the `sam_release` library routine. To specify different-sized file stubs for different types of files or different applications, specify the `--s` option on the `release` command or the `-s` option on the `sam_release` library routine. The `-s` and `s` values must be less than the `-o maxpartial` value used with the mount command when the file system was mounted.

---

**Note** – A partially released file takes up space on the disk equal to one DAU. For example, if the partial release file stub is set to 16K and the DAU size is 256K, the actual space consumed by the file on the disk is 256K.

---

Use the mount option, `-o partial_stage=n`, to establish how much of a partial release stub must be read before the rest of the file is staged. Reading past the `-o partial_stage=n` size specification initiates the stage of the file.

By default, the `-o partial_stage=n` option is set to the size of the partial release stub. Changing this value affects file staging as follows:

- If the `-o partial_stage=n` option is set to the size of the partial release stub, the default behavior prevents the file from being staged until the application reaches the end of the partial release stub. Waiting until the end of the stub is reached causes a delay in accessing the rest of the file.
- If the `-o partial_stage=n` option is set to a value smaller than the partial release stub, the file is staged after the application crosses the threshold set by the `-o partial_stage=n` option. This reduces the chance of a delay in accessing the rest of the file data.

**EXAMPLE 14-1** Partial Staging

In this example, a site has set the following options:

- `-o partial_stage=16` (16 kilobytes)
- `-o partial=2097152` (2 gigabytes)
- `-o maxpartial=2097152` (2 gigabytes)

The `filemgr` program reads the first 8 kilobytes of a file. The file is not staged.

A video-on-demand application reads the same file. After it reads past the first 16 kilobytes of the file, the file is staged. The application continues reading while the archive tape is mounted and positioned.

When the video-on-demand application reads past 2 gigabytes of file data, it is reading immediately behind the staging activity. The application does not wait, because the tape mounting and positioning is done while the application reads the partial file data.

Several command-line options affect whether a file can be marked for partial release. Some options are enabled by the system administrator, and others can be enabled by individual users. The following sections describe the release characteristics that can be set by the various types of users.

- [“Summary of System Administrator Options” on page 169](#)
- [“Summary of User Options” on page 170](#)

## Summary of System Administrator Options

As a system administrator, you can change the maximum value and default value for partial release when the file system is mounted. The mount options in the following table affect partial release. For more information about the mount command, see [“mount\\_samfs\(1M\)” in \*Sun QFS and Sun Storage Archive Manager Reference Manual\*](#).

Option	Effect
<code>-o maxpartial= n</code>	Determines the maximum amount of space in kilobytes that can remain in disk cache when a file is marked for partial release. The maximum value is 2,097,152 kilobytes, which is 2 gigabytes. The minimum value is 0, which disables the partial release feature so that released files are released completely, and no portion of a file remains in disk cache. Users cannot override the value specified on this option after the file system is mounted. By default, the <i>n</i> argument is set to 16.

Option	Effect
-o partial= <i>n</i>	Sets a default amount of space in kilobytes that remains in disk cache when a user marks a file for partial release by using the <code>release</code> command's <code>-p</code> option. The <i>n</i> argument must be at least 8 and it can be as great as the value specified for the <code>-o maxpartial=<i>n</i></code> option. Because some applications do not need access to the entire file to complete their operations, this option ensures that applications have the beginnings of files available to them. Also, this option prevents files from being staged unnecessarily. By default, <i>n</i> is 16. A file that has been partially released from a disk takes up space on the disk equal to one DAU.
-o partial_stage= <i>n</i>	Specifies that when a partially released file is accessed, <i>n</i> bytes of the file must be read before the entire file is staged from the archive media. Set this value lower than the amount of the <code>-o partial</code> setting. For <i>n</i> , specify an integer from 0 to the <code>-o maxpartial</code> specification. By default, this value is 16 or the value specified for the <code>--o partial</code> option.
-o stage_n_window= <i>n</i>	Specifies the amount of data to be staged at any one time to <i>n</i> . Specify an integer from 64 to 2,048,000. The default is 256 kilobytes. This option applies only to files that have the <code>stage -n</code> attribute set.

## Summary of User Options

As a user, you can set maximum and default values for the size of a file stub that can remain in disk cache after the file is released. You can also determine whether the partial release feature is enabled for a particular file system.

By using the `release` command and the `sam_release` library routines, however, a user can set other release attributes and can specify the files to be marked for partial release. The command and library options that determine partial release attributes are shown in the following table. For more information, see “[release\(1\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* and “[sam\\_release\(3\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

TABLE 14-1 User Release Options

Options	Effect
<code>release</code> command and <code>--p</code> option or <code>sam_release</code> library routine and <code>-p</code> option	The <code>--p</code> and <code>-p</code> options mark the named file for partial release. If these options are used, the amount of the file remaining in online disk cache after the file is released depends on the value of the <code>-o partial=<i>n</i></code> option that was set when the file system in which the file resides was mounted. These options cannot be used to specify the number of bytes to remain online.

TABLE 14-1 User Release Options (Continued)

Options	Effect
release command and -s partial_size option or sam_release library routine and -s option	The -s and s options mark the named file for partial release, and they specify the amount of the file to remain in online disk cache. The arguments to the -s or s options specify the amount, in kilobytes, to remain online. A user cannot specify that the amount of a file remaining online be greater than the amount specified for the -o maxpartial=n value when the file system was mounted. If the user's value is greater than the value for the file system, the value for the file system is used, and the user's specification is ignored.



# Configuring the Stager

---

This chapter describes how to configure the stager through the `stager.cmd` file.

## About Staging

Staging is the process of copying file data from nearline or offline storage back to online storage.

The stager starts when the `samd` daemon runs. The stager has the following default behavior:

- The stager attempts to use all the drives in the library.
- The stage buffer size is determined by the media type, and the stage buffer is not locked.
- No log file is written.
- Up to 1000 stage requests can be active at any one time.

You can customize the stager's operations for your site by inserting directives into the `/etc/opt/SUNWsamfs/stager.cmd` file.

When an application requires an offline file, its archive copy is staged to disk cache (if the `-n` option's `-stage never` is not set). To make the file available to an application immediately, the read operation tracks along directly behind the staging operation so that the access can begin before the entire file is staged.

Stage errors include media errors, unavailability of media, unavailability of an automated library, and others. If a stage error is returned, the SAM-QFS software attempts to find the next available copy of the file, if one exists and if there is a device available to read the archive copy's media.

## About the stager . cmd File

In the stager . cmd file, specify directives to override the default behaviors. You can configure the stager to stage files immediately, to never stage files, to staging partially, and to specify other staging actions. For example, specifying the never-stage attribute benefits applications that access small records from large files because the data is accessed directly from the archive media without staging the file online.

This section describes the stager directives. For additional information about stager directives, see “stager.cmd(4)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*. If you are using the SAM-QFS Manager software, you can control staging from the File System Summary or File System Details page. You can browse the file system and see the status of individual files, use filters to view certain files, and select specific files to stage. You can select which copy to stage from or let the system choose the copy.

### EXAMPLE 15-1 stager . cmd File

The following example shows a stager . cmd file after all possible directives have been set.

```
# This is stager.cmd file /etc/opt/SUNWsamfs/stager.cmd
drives=dog 1
bufsize=od 8 lock
logfile=/var/adm/stage.log
maxactive=500
```

## ▼ How to Create a stager . cmd File

- 1 In the `/etc/opt/SUNWsamfs/stager . cmd` file, add the directives to control staging at your site, according to the information in the following sections:
  - “How to Create a stager . cmd File” on page 174
  - “drives Directive: Specifying the Number of Drives for Staging” on page 175
  - “bufsize Directive: Setting the Stage Buffer Size ” on page 175
  - “logfile Directive: Specifying a Log File ” on page 176
  - “maxactive Directive: Specifying the Number of Stage Requests” on page 178
  - “copysel Directive: Specifying the Copy Sequence for Staging” on page 179
- 2 Save and close the stager . cmd file.
- 3 Propagate the file changes and restart the system.

```
# samd config
```

## drives Directive: Specifying the Number of Drives for Staging

By default, the stager uses all available drives when staging files. If the stager keeps all the drives busy, it can interfere with the archiver's activities. The `drives` directive specifies the number of drives available to the stager. This directive has the following format:

```
drives = library count
```

Argument	Definition
<i>library</i>	The family set name of a library as it appears in the <code>mc f</code> file.
<i>count</i>	The maximum number of drives to be used. By default, this is the number of drives configured in the <code>mc f</code> file for this library.

### EXAMPLE 15-2 Drives Directive

The following example specifies that only one drive from the `dog` family set's library is used for staging files:

```
drives = dog 1
```

## bufsize Directive: Setting the Stage Buffer Size

By default, a file being staged is read into memory in a buffer before being restored from the archive media to disk cache. Use the `bufsize` directive to specify a buffer size and, optionally, to lock the buffer. These actions can improve performance. You can experiment with various *buffer-size* values. The directive has the following format:

```
bufsize = media buffer-size [lock]
```

Argument	Definition
<i>media</i>	Specify the archive media type from the list on the <code>mc f(4)</code> man page.
<i>buffer-size</i>	A number from 2 through 8192. The default is 16. This value is multiplied by the <code>dev_blksize</code> value for the media type, and the resulting buffer size is used. The <code>dev_blksize</code> value is specified in the <code>defaults.conf</code> file. The higher the number specified for <i>buffer-size</i> , the more memory is used. For more information, see “ <a href="#">defaults.conf(4)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .

Argument	Definition
lock	The lock argument indicates that the stager should use locked buffers when staging archive copies. If lock is specified, the stager sets file locks on the stage buffer in memory for the duration of the copy operation. This avoids the overhead associated with locking and unlocking the buffer for each I/O request and can thereby result in a reduction in system CPU time. The lock argument should be specified only on large systems with large amounts of memory. Insufficient memory can cause an out-of-memory condition. The lock argument is effective only if direct I/O is enabled for the file being staged. By default, lock is not specified, and the file system sets the locks on all direct I/O buffers, including those for staging. For more information about enabling direct I/O, see “setfa(1)” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> , “sam_setfa(3)” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> , or the -O forcedirectio option on “mount_samfs(1M)” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .

## logfile Directive: Specifying a Log File

You can request that the SAM-QFS software collect file-staging event information and write it to a log file. By default, no log file is written. The `logfile` directive specifies a log file to which the stager can write logging information. The stager writes one or more lines to the log file for each file staged. This line includes information such as the name of the file, the date and time of the stage, and the volume serial number (VSN). The directive has the following format:

```
logfile=filename [event]
```

Argument	Definition
<i>filename</i>	Specify a full path name.
<i>event</i>	Specify one or more staging events. If you specify more than one event, use spaces to separate each them. Possible event specifications are as follows.  all - Logs all staging events.  start - Logs when staging begins for a file.  finish - Logs when staging ends for a file. Enabled by default.  cancel - Logs when the operator cancels a stage. Enabled by default.  error - Logs staging errors. Enabled by default.

### EXAMPLE 15-3 Specifying a Stager Log File

The following directive creates the `/var/adm/stage.log` file:

```
logfile=/var/adm/stage.log
```

## EXAMPLE 15-4 Stager Log File

```

S 2003/12/16 14:06:27 dk disk01 e.76d 2557.1759 1743132 /sam1/testdir0/filebu 1 root other root 0 -
F 2003/12/16 14:06:27 dk disk01 e.76d 2557.1759 1743132 /sam1/testdir0/filebu 1 root other root 0 -
S 2003/12/16 14:06:27 dk disk02 4.a68 1218.1387 519464 /sam1/testdir1/fileaq 1 root other root 0 -
S 2003/12/16 14:06:43 dk disk01 13.ba5 3179.41 750880 /sam1/testdir0/filecl 1 root other root 0 -
F 2003/12/16 14:06:43 dk disk01 13.ba5 3179.41 750880 /sam1/testdir0/filecl 1 root other root 0 -
S 2003/12/16 14:06:59 dk disk01 17.167b 1155.1677 1354160 /sam1/testdir0/filedb 1 root other root 0 /
-
F 2003/12/16 14:06:59 dk disk01 17.167b 1155.1677 1354160 /sam1/testdir0/filedb 1 root other root 0 /
-
S 2003/12/16 14:06:59 dk disk02 f.f82 3501.115 1458848 /sam1/testdir1/filecb 1 root other root 0 -
S 2003/12/16 14:07:15 dk disk01 1f.473 1368.1419 636473 /sam1/testdir0/fileed 1 root other root 0 -
S 2003/12/16 14:07:15 dk disk02 16.f15 3362.45 1065457 /sam1/testdir1/filecz 1 root other root 0 -
S 2003/12/16 14:07:31 dk disk01 23.201d 3005.1381 556807 /sam1/testdir0/fileeq 1 root other root 0 -
S 2003/12/16 14:07:47 dk disk01 26.c4d 2831.1113 1428718 /sam1/testdir0/fileez 1 root other root 0 -
S 2003/12/16 14:07:47 dk disk02 1b.835 3736.59 1787855 /sam1/testdir1/filedp 1 root other root 0 -

```

The following table describes the content of the stager log file fields.

Field	Example Value	Content Description
1	S	Stage activity - S for start, C for canceled, E for error, F for finished.
2	2003/12/16	Date of the stage action, in <i>yyyymm/dd</i> format.
3	14:06:27	Time of the stage action, in <i>hhmm:ss</i> format.
4	dk	Archive media type. For information about media types, see “mcf(4)” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
5	disk01	VSN.
6	e.76d	Using hexadecimal format, the physical position of the start of the archive file on media (tarfile) and the file offset on the archive file.
7	2557.1759	Inode number and generation number. The generation number is used in addition to the inode number for uniqueness because inode numbers are reused.
8	1743132	Length of the file.
9	/sam1/testdir0/filebu	Name of the file.
10	1	Archive copy number.
11	root	User ID of the file.
12	other	Group ID of the file.
13	root	Group ID of the requestor.

Field	Example Value	Content Description
14	0	Equipment ordinal of the drive from which the file was staged.
15	-	A V in this field indicates that data verification is being used for the file.

## maxactive Directive: Specifying the Number of Stage Requests

The `maxactive` directive enables you to specify the number of stage requests that can be active at any one time. The directive has the following format:

```
maxactive=number
```

By default, *number* is 4000. The minimum number allowed is 1. The maximum allowed is 500,000.

The following example specifies that no more than 500 stage requests can be in the queue simultaneously:

```
maxactive=500
```

## Archive Set Assignment Directive: Specifying Stage Attributes for All Files in an Archive Set

Most directives in the `archiver.cmd` file affect only archiving, but you can use the archive set assignment directive to specify stage attributes that apply to all files in an archive set.

Chapter 12, “Archive Set Directives (`archiver.cmd`),” describes the archive set assignment directive and its arguments completely. The following table shows the staging directives that can appear in an archive set assignment directive in the `archiver.cmd` file.

Directive	Effect
<code>-stage a</code>	Specifies that the files in the archive set should be associatively staged.
<code>-stage d</code>	Reset to default.
<code>-stage n</code>	Specifies that the files in the archive set should never be staged.

## copysel Directive: Specifying the Copy Sequence for Staging

The configuration option, `copysel`, in the `stager.cmd` file allows changing the stager copy selection sequence per file system.

By default, the copy selection for staging is from copy number 1 to copy number 4. Copy number 1 is selected first, then copy 2, then copy 3, and finally copy number 4. You can change the stager copy selection sequence for each file system by entering values 1 through 4 for  $n$  for the configuration option, `copysel = n1:n2:n3:n4`. Four copies,  $n1$  through  $n4$ , must be defined even if four copies are not available.

The following is an example of a log file that shows the result of the `stager.cmd` operation.

### EXAMPLE 15-5 Log File of the `stager.cmd` Operation

```
logfile = /var/opt/SUNWsamfs/log/stager
drives = hp30 1
copysel = 4:3:2:1
fs = samfs1
copysel = 3:1:4:2
streams
dk -maxsize 2G -maxcount 10000
endstreams
```

The log file in the above example is interpreted as follows:

- The log file is in the `/var/opt/SUNWsamfs/log/stager` directory.
- The media drive, HP30, is allowed to use only one drive for staging files.
- By default, the stager selects copies in the order 4, 3, 2, and 1, for staging the files.
- For the file system `samfs1`, the stager selects copies in the order 3, 1, 4, and 2 for staging the files.
- The maximum size of the stream for the `dk` media type is 2 Gbytes, and the maximum file count for each stream is 10,000.

For more information, see “`stager.cmd(4)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Prioritizing Preview Requests

Both the archiver and stager processes request that media is loaded and unloaded. If the number of requests exceeds the number of drives available for media loads, the excess requests are sent to the preview queue.

The number of entries that can be in the preview queue is determined by the `previews=` directive in the `defaults.conf` file. For information about changing the value of this directive, see “`defaults.conf(4)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

By default, preview requests are satisfied in first-in-first-out (FIFO) order.

The overall priority of preview requests is determined by the combination of static and dynamic factors. Higher numbers correspond to higher priority. A static priority factor is set when the request is generated. Its effect does not change the overall priority after the request is generated and is waiting to be satisfied. A dynamic priority factor can increase or decrease the overall priority of a request while the request is waiting to be satisfied.

You can override the FIFO default by entering directives in the `/etc/opt/SUNWsamfs/preview.cmd` command file.

## About the `preview.cmd` File

The `sam-amld` daemon reads the `preview.cmd` file at startup. This file orders the requests in the preview queue according to whether the request is for staging or archiving. You can increase the priority for specific VSNs and you can control the priority of preview requests for specific file systems.

The following rules apply to the `preview.cmd` file:

- Place one directive per line.
- If you change this file while the `sam-amld` daemon is running, restart the daemon to have your changes take effect.
- Begin comment lines with a hash character (`#`).  
For more information about this file, see “[preview.cmd\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

The following types of directives are used in the `preview.cmd` file:

- Global directives, which apply to all file systems
- File-system directives, specific to individual file systems

Global directives are placed at the top of the file and their settings apply to all file systems.

File system directives begin with the `fs =` directive, which names the file system to which all subsequent directives apply. More than one block of file directives can appear in a file. File system directives apply until the next `fs =` line is encountered or until the end of file is encountered.

When multiple directives affect a file system, the directives that are specific to that file system override the global directives.

The following sections describe how to edit the `preview.cmd` file to control the preview queue:

- “[How to Set the Global VSN and Age Priority Directives](#)” on page 181
- “[How to Set Global or File System-Specific Water Mark Directives](#)” on page 181

- [“Setting Up a Preview Request Priority Scheme” on page 183](#)

## ▼ How to Set the Global VSN and Age Priority Directives



**Caution** – The VSN and age priority directives are global directives, so they are placed before any file-system-specific directives in the `preview.cmd` file.

### 1 Update the `vsn_priority` directive.

`vsn_priority = value`

This directive is a static priority factor that indicates the value by which the total priority increases when there is a high-priority volume. The default value for is `1000.0`. To use this priority factor, a volume must have its priority flag set before it is scheduled as a preview request. Use the `chmed(1M)` command to set the priority flag with the `-p` option (for example, `chmed +p lt.AAA123`).

### 2 Update the `age_priority` directive.

`age_priority = factor`

This directive is a static priority factor, although its effect is dynamic. This factor is multiplied by the number of seconds for which a request is a preview request. The result is then added to the overall priority of the request. The longer a request waits, the higher the priority becomes. Setting this factor ensures that older requests are not indefinitely superseded by newer requests with other higher-priority factors.

Setting this factor to more than `1.0` increases the importance of the time factor in calculation of the total priority and setting it to less than `1.0` decreases the importance of the time factor. Setting the factor to `0.0` eliminates the time factor from the overall priority calculation.

A volume whose priority flag is not set increases in priority based on the time it remains in the queue. Its priority can become higher than a VSN that comes into the queue later with the priority flag already set.

## ▼ How to Set Global or File System-Specific Water Mark Directives

- The water mark preview request directives can be used as either global or file-system-specific directives. These directives determine the water mark priority of the preview requests, as shown in the following equation.

$$lwm\_priority + lhwm\_priority + hlwm\_priority + hwm\_priority = \text{water mark priority}$$

Together, the four water mark settings create a dynamic priority factor that includes a percentage value indicating how full the file system is and the levels at which the HWM and LWM are set. The value assigned to a preview request is determined by whether a factor is global, specific to a file system, or not set.

The water mark priorities are used to calculate only requests for archiving. They are not used to calculate media requests for staging. When the water mark priority factor is a positive number, the result on the overall calculated priorities is to raise archiving requests over staging requests. In contrast, when the water mark priority factor is a negative number, the overall priority for archiving requests is reduced, which tends to favor staging requests over archival requests. A water mark priority factor of 0.0 (or no specified directive) indicates that no special action occurs.

For more information, see Example 1 - Scheme for Enforcing Stage Requests.

The water mark directives have the following format:

*wmtype\_priority = value*

Water Mark Directive	Argument
<i>lwm_priority = value</i>	Specify the amount by which you want the water mark priority factor to change for archiving requests when the file system is below the LWM level. The default is 0.0.
<i>lhwm_priority = value</i>	Specify the amount by which you want the water mark priority factor to change for archiving requests when the file system crosses from below to above the LWM level but remains below the HWM level. This shift indicates that the file system is filling up. The default is 0.0.
<i>hlwm_priority = value</i>	Specify the amount by which you want the water mark priority factor to change for archiving requests when the file system has crossed from above the HWM level to below it, but remains above the LWM level. This shift indicates that the releaser was not able to free enough disk space to leave the file system below the LWM level. The default is 0.0.
<i>hwm_priority = value</i>	Specify the amount by which you want the water mark priority factor to change for archiving requests when the file system is above the HWM level. The default is 0.0.

When a file system crosses from one condition to another, the priority of each volume associated with that file system is recalculated based on the appropriate water mark priority setting, with or without the `chmed` command's `-p` option.

The following example frees enough disk space so that the file system goes below the LWM level.

```
lhwm_priority = -200.0
hlwm_priority = 100.0
```

## Setting Up a Preview Request Priority Scheme

The total priority for a preview request is the sum of all priority factors:

$$\text{total priority} = \text{vsn\_priority} + \text{wm\_priority} + (\text{age\_priority} * \text{time\_in\_sec\_as\_preview\_request})$$

Change the default FIFO scheme only for reasons such as the following:

- Ensure that staging requests are processed before archive requests.
- Ensure that archive requests gain top priority when a file system is about to fill up.
- Push requests that use a specific group of media to the top of the preview request list.

The following example shows a `preview.cmd` file that addresses these three conditions.

### EXAMPLE 15-6 Sample `preview.cmd` File

```
# condition 1
lwm_priority = -200.0
lhwm_priority = -200.0
hlwm_priority = -200.0
# condition 2
hwm_priority = 500.0
# condition 3
age_priority = 1.0
```

For environments in which user access to data is of paramount importance, the VSN drives are limited, or file archiving is performed as a background function, use the `preview.cmd` file to influence how the storage system resources handle staging requests. You can customize the settings in the `preview.cmd` file to support any of the preceding scenarios and influence the configured SAM-QFS environment.

Because data is not affected by the settings in this file, you are encouraged to experiment and adjust the directive settings to achieve the proper balance between archiving and staging requests when weighed against the priorities of each preview request.

### EXAMPLE 15-7 Scheme for Enforcing Stage Requests

The following example calculations show how you can use a negative value for `wm_priority` to ensure that stage requests have priority over archive requests. This example assumes the following:

- Several requests are sitting in the queue for 100 seconds.
- The default value `vsn_priority` is 1000.

The following table shows how the total request priorities are calculated as follows:

Priority	Calculation
Archive VSN with priority, LWM	$1000 + (-200) + (1 \times 100) = 900$

**EXAMPLE 15-7** Scheme for Enforcing Stage Requests (Continued)

Priority	Calculation
Stage VSN with priority, LWM	$1000 + 0 + (1 \times 100) = 1100$
Stage VSN without priority, LWM	$0 + 0 + (1 \times 100) = 100$

**EXAMPLE 15-8** Scheme for Enforcing Archive Requests

When the environment is balanced between the importance of staging a file for the user and the importance of getting new files archived to media, the biggest concern is exceeding the HWM level. In this situation, if not enough files have met their archive requirements to lower the percentage of the file system that is full, meeting the pending archive requests is the best way to keep the file system from reaching its limit.

In this situation, the `preview.cmd` file can be as simple as the following example:

```
hwm_priority = 500.0
```

**EXAMPLE 15-9** Scheme for Ranking Requests by Media Type

Assume that a site has an environment in which users are working on groups of files that use specific volumes and are segregated from other users. In this environment, certain projects might have higher priorities at certain times; therefore, greater priority is required from the available system storage resources. The following example gives users and their media the appropriate priority:

```
vsn_priority = 5000.0
```

Then, for every volume in the priority user's group, include the following information:

```
# chmed +p lt. VSN
```

Now every request that requires the specified VSN is placed above other pending mount requests in the preview queue.

**EXAMPLE 15-10** Scheme for Complex Priorities

Assume two file systems that have the following requirements:

- No request must wait too long in the queue (`age_priority`).
- When one of the file systems is below the LWM level, staging requests take precedence.
- When one of the file systems is above the LWM level but below the HWM level, do not prioritize requests.

The following example shows the affected directives.

**EXAMPLE 15-10** Scheme for Complex Priorities (Continued)

```
lwm_priority = -200.0
lhwm_priority = 0.0
hlwm_priority = 0.0
```

When one of the file systems goes over the HWM level, archive requests take priority.

Assume both file systems are over the HWM level but the second file system (`samfs2`) must be prevented from reaching its limit. The following example shows a `preview.cmd` file that prioritizes requests according to these requirements.

```
age_priority = 100.0
vsn_priority = 20000.0
lhwm_priority = -200.0
hlwm_priority = -200.0
fs = samfs1
hwm_priority = 1000.0
fs = samfs2
hwm_priority = 5000.0
```



# Configuring the Recycler

---

This chapter describes the recycling process and directives.

## About Recycling

Recycling is the process of reclaiming space on archive volumes. The recycler works with the archiver to reclaim the space occupied by unused archive copies. As users modify files, the archive copies associated with the old versions can be purged from the system. The recycler identifies the volumes with the largest proportions of expired archive copies and directs the movement of unexpired copies to different volumes. If only expired copies exist on a given volume, a site-defined action is taken. For example, a volume can be relabeled for immediate reuse or exported to offsite storage, keeping a separate historical record of file changes. Users are unaware of the recycling process.

At any time, the space on an archive volume consists of the following:

- Current data, consisting of archive images that are active
- Expired data, consisting of archive images that are no longer active
- Free space, consisting of space that is not being used by active or expired archive images

The recycler keeps the amount of space consumed by expired data to the minimum defined by site-specified parameters.

The capacity of a volume is the total amount of space for data on a volume. For example, a 10-gigabyte volume with 3 gigabytes written to it has a capacity of 10 gigabytes and 7 gigabytes of free space.

New or newly labeled archive media starts with all of its capacity as free space. As data is archived to the media, the amount of free space decreases and the amount of current data increases.

As files in the file system are changed or removed, their archive images expire and the classification of their data changes from the current data to the expired data. The physical space used by the archive images does not change. However, no file in the file system points to that space. When space is recycled, these images are removed and the space they occupied become free, available for other purposes. The goal of the recycler is to transform space used by expired data into free space without losing any current data.

## Recycle Process

The recycler and the archiver work together as follows:

1. The recycler marks all the current archive images that are present on a volume with the `rearchive` attribute.
2. If you are archiving to removable media, the recycler marks the archive volume with the `recycle` attribute to prevent the archiver from writing any more archive images to the volume.
3. The archiver moves all the marked images to another volume. This operation is called *re-archiving*. After the archiver moves the current archive images from the old volume to the new volume, the old volume contains only free space and expired space. If you are archiving to removable media cartridges, you can relabel and reuse the cartridge. If you are archiving to disk, the recycler removes the file that contains the expired archive images.

The recycler is designed to run periodically, although you can run it at any time. It performs as much work as it can each time it is invoked. The recycler has to finish marking copies for re-archiving before the archiver can re-archive the files. Sometimes expired archive images, with the `rearchive` attribute set, remain on media. This situation can happen under the following conditions:

- The archiver does not run after the recycler marks expired archive images.
- Media is not available for the archiver to use when moving the unexpired archive images.
- Miscellaneous archiver anomalies occur.

Between executions, the recycler keeps state information in the library catalogs and the inodes. During the recycling process, you can use the `sls(1)` command and its `-D` option to display information about a file. The output from the `sls(1)` command shows whether a file is scheduled for re-archiving.

## Planning for Recycling

Recycling is accomplished using two methods, depending on the type of media.

TABLE 16-1 Media Types and Recycling Methods

Archive Media	Recycling Method
Removable media cartridges	By automated library
Removable media cartridges	By archive set
Disks	By archive set

For information about configuring by these methods, see [Chapter 16, “Configuring the Recycler.”](#)

## Recycling Methods

You initiate recycling by entering the `sam-recycler` command either manually or through a cron job. The following table shows the recycling methods.

TABLE 16-2 Recycling Methods and Media Types

Recycling Method	Archive Media	Where To Configure
By automated library	Removable media cartridges	<code>recycler.cmd</code> , <code>recycler.sh</code>
By archive set	Removable media cartridges	<code>recycler.cmd</code> (optional, for VSNs not covered in archive set), <code>recycler.sh</code> , <code>archiver.cmd</code>
By archive set	Disks	<code>archiver.cmd</code>

Before configuring the recycler, note the following:

- Directives in the `archiver.cmd` file control recycling by archive set. Directives in the `recycler.cmd` file control recycling by library. In addition, the `recycler.cmd` file controls general recycler behavior. For information about recycler directives, see [“Creating a recycler.cmd File” on page 191](#).
- Do not recycle volumes that contain removable media files. You create removable media files by using the `request(1)` command. A volume with removable media files can never be drained.
- Do not run the recycler while performing maintenance on a file system. The recycler uses the `.inodes` file and the `mc f` file to determine whether files are current or expired and to identify the devices associated with a file system. Absence of proper information in the `.inodes` and `mc f` files can cause current archived data to appear as expired and be recycled.
- All file systems must be mounted when the recycler is run. If you are recycling from an online disk, the file system that contains the disk volumes must be mounted and the host system must be accessible.



**Caution** – Take extreme care when configuring the recycler if you are using disk archiving in an environment with multiple SAM-QFS servers. The `diskvol1.conf` file for each SAM-QFS server must point to a unique set of disk archiving target directories. If any of these directories are shared by different SAM-QFS servers, running the recycler from one SAM-QFS server will destroy the disk archive data that is managed by the other SAM-QFS server.

## Controlling Recycling

You can enable and disable the recycle process using parameters in the command files.

When you are configuring the recycler and testing the results, edit the appropriate file to add its parameter.

Recycle Method	Parameter
Archive set	- recycle_ignore parameter in the <code>archiver.cmd</code> file.
Automated library	ignore parameter in the <code>recycler.cmd</code> files.

When you are ready to use the recycler, edit the files to remove the parameter.

As an alternative, you can use the SAM-QFS Manager. For more information, see the SAM-QFS Manager online help.

## Starting the Recycler

You start the recycler by issuing the `sam-recycler` command.

```
# sam-recycler
```

The recycler reads the `recycler.cmd` file.

Examine the standard output log, SAM-QFS log, and `/var/adm/messages` for any error messages from the recycler. Correct your files if errors appear.

When the recycler is initiated, the default recycler settings specified in library Directive: Specifying Recycling for an Entire Automated Library<sup>1</sup> take effect. For more information about the recycler, see “[sam-recycler\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

If the system is performing in a routine manner, you can use a `crontab` entry to run the recycler periodically. The frequency you choose depends on your site's conditions. For instructions about creating a `crontab` entry, see the `cron(1M)` man page.

The following example entry in root's crontab file specifies that the cron daemon run the recycler every five minutes after every odd-numbered hour:

```
5 1,3,5,7,9,11,13,15,17,19,21,23 * * * /opt/SUNWsamfs/sbin/sam-recycler
```

## Configuring Recycling on Removable Media Cartridges

If you are recycling archive copies on cartridges in a library, create a `recycler.cmd` file.

If you are recycling by archive set, configure each library in the `recycler.cmd` file. This ensures that VSNs that do not fall into an archive set and can be recycled if needed.

Create a `recycler.sh` file to complete the operation.

### Creating a `recycler.cmd` File

The `recycler.cmd` file contains general recycling directives. It can also contain directives for each library in the SAM-QFS environment. A typical `recycler.cmd` file contains the following directive lines:

- A `logfile=` directive to specify a recycler log file.
- One or more directives for each library that contains volumes to be recycled. This line must contain the family set name (from the `mcf` file) for the library being recycled. The family set name identifies the library to the recycler.
- During testing, include the `ignore` keyword. You remove the `ignore` keyword later.

As the root user, add one or more directive to the `/etc/opt/SUNWsamfs/recycler.cmd` file.

As an alternative, you can create a `recycler.cmd` file using SAM-QFS Manager software. For more information, see the SAM-QFS Manager online help.

#### EXAMPLE 16-1 Sample `recycler.cmd` File

```
logfile = /usr/tmp/recycler.log
stk30 -hwm 51 -mingain 60 -ignore -mail root
```

The `recycler.cmd` file accepts the directives described in the following sections:

- [“logfile Directive: Specifying a Log File” on page 192](#)
- [“no\\_recycle Directive: Preventing Recycling” on page 193](#)
- [“library Directive: Specifying Recycling for an Automated Library” on page 193](#)

## logfile Directive: Specifying a Log File

The logfile directive creates a recycler log file. This directive has the following format:

```
logfile = filename
```

For *filename*, specify the path to the log file.

The following is an example of a logfile= directive line:

```
logfile=/var/adm/recycler.log
```

### EXAMPLE 16-2 Sample Recycler Log File for Removable Media Cartridges

The following example shows a sample recycler log file for recycling removable media cartridges.

```
===== Recycler begins at Wed Dec 12 14:05:21 2001 =====
Initial 2 catalogs:
0 Family: m160 Path: /var/opt/SUNWsamfs/catalog/m160
Vendor: ADIC Product: Scalar 100
SLOT ty capacity space vsn
0 at 25.0G 25.0G CLN005
1 at 48.5G 6.1G 000003
2 at 48.5G 32.1G 000004
3 at 48.5G 35.1G 000005
4 at 48.5G 44.6G 000044
5 at 48.5G 45.1G 000002
6 at 48.5G 45.9G 000033
7 at 48.5G 48.5G 000001
Total Capacity: 364.8G bytes, Total Space Available: 282.3G bytes
Volume utilization 22%, high 95% VSN_min 50%
Recycling is ignored on this robot.
1 Family: hy Path: /var/opt/SUNWsamfs/catalog/historian
Vendor: Sun SAM-FS Product: Historian
SLOT ty capacity space vsn
(no VSNs in this media changer)
Total Capacity: 0 bytes, Total Space Available: 0 bytes
Volume utilization 0%, high 95% VSN_min 50%
Recycling is ignored on this robot.
8 VSNs:
---Archives--- -----Percent----- m160
---Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 87 13 m160:at:000003
no-data VSN 0 0 0 33 67 m160:at:000004
no-data VSN 0 0 0 27 73 m160:at:000005
no-data VSN 0 0 0 8 92 m160:at:000044
no-data VSN 0 0 0 7 93 m160:at:000002
no-data VSN 0 0 0 5 95 m160:at:000033
empty VSN 0 0 0 100 m160:at:CLN005
empty VSN 0 0 0 100 m160:at:000001
Recycler finished.
===== Recycler ends at Wed Dec 12 14:05:32 2001 =====
```

## no\_recycle Directive: Preventing Recycling

The `no_recycle` directive disables recycling of volumes. This directive has the following format:

```
no_recycle media-type VSN-regexP [VSN-regexP . . .]
```

Argument	Definition
<i>media-type</i>	A media type from the <code>mcf(4)</code> man page. You can disable recycling of volumes stored on particular type of media
<i>VSN-regexP</i>	One or more space-separated regular expressions to describe the volumes. You can disable recycling for specific cartridges. For information, see the <code>regexP(5)</code> man page.

The following example excludes any tape volumes whose VSN identifiers begin with DLT:

```
no_recycle lt DLT.*
```

## library Directive: Specifying Recycling for an Automated Library

The `library` directive enables you to specify various recycling parameters for the VSNs associated with a specific library. This directive has the following format:

```
library parameter [parameter . . .]
```

For *library*, specify the library's name as specified in the family set field of the `mcf` file.

For *parameter*, specify one or more space-separated *parameter* keywords from the following table.

Parameter Value	Definition
<code>-dataquantity size</code>	Maximum amount of data that the recycler can schedule for rearchiving in its efforts to clear volumes of useful data. Default is 1 gigabyte.
<code>-hwm percent</code>	Library high-water mark. Default is 95.
<code>-ignore</code>	Directive that prevents volumes in this library from being recycled. This directive is useful during testing of the <code>recycler.cmd</code> file.
<code>-mail email-address</code>	Email addresses to which recycling email messages are to be sent. By default, no email is sent.

Parameter Value	Definition
-mingain <i>value</i>	Minimum VSN gain. Default depends on media: <ul style="list-style-type: none"> <li>■ For volumes with less than 200 Gbyte capacity, the default mingain is 60%.</li> <li>■ For volumes with 200 Gbyte or larger capacity, the default mingain is 90%.</li> </ul>
-vsncount <i>count</i>	Maximum number of recycled volumes to be counted. Default is 1.

**EXAMPLE 16-3** library Directive

The following example specifies the following for library gr47:

- The library qualifies for recycling when the volumes in the library are 85 percent full.
- The minimum percent gain is 40 percent.
- Only one volume is to be recycled. This is also a default setting.
- Recycling messages are emailed to root.
- No more than 1 gigabyte is to be re-archived. This value is the default, so it is not specified in the `recycler.cmd` file.

```
gr47 -hwm 85 -ignore -mail root -mingain 40
```

The following sections describe the parameters.

**-hwm Parameter**

By specifying a high-water mark, you set the percentage of media usage below which recycling cannot occur. This percentage is the ratio of the used space in the library to its total capacity. For example, a library that holds ten 20-gigabyte tapes, three of them 100 percent full and the remaining seven each 30 percent full, has the following media utilization percentage:

$$((3 * 1.00 + 7 * 0.30) * 20G) / (10 * 20G) * 100\% = 51\%$$

This calculation does not distinguish between current data and expired data. It addresses only the amount of media used.

In the example, when the utilization percentage is 51 percent or less, the recycler does not automatically select any of the automated library's VSNs for recycling.

You can force a VSN to be recycled by using the following command to set the recycling flag:

```
# chmed +c lt. VSN
```

When the `+c` flag is set, the archiver does not write any more archive images to the volume. The `+c` flag can be viewed through the `samu` utility. For more information, see “[chmed\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* and “[samu\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*. For information about using the `samu(1M)` operator utility, see the *Sun QFS File System 5.3 Configuration and Administration Guide*.

## -mingain Parameter

The minimum VSN gain percentage sets a lower limit on the amount of space to be gained by recycling a cartridge. For example, if a cartridge in an automated library is 95 percent current data and 5 percent is efficient, setting the minimum gain to 6 percent or more inhibits the recycler from automatically selecting this VSN.

## -ignore Parameter

The `-ignore` parameter disables the recycler for a particular library. Use it while you are configuring and testing the recycler.

## -mail Parameter

The `-mail` parameter specifies that the recycler sends email messages when recycling occurs on a library. The email message has the following subject line:

Robot *robot-name* recycle

### EXAMPLE 16-4 Sample Recycling Messages

```
I will recycle VSN vsn.
Cannot find any candidate VSN in this media changer.
Previously selected VSN _vsn_ is not yet finished recycling.
Previously selected VSN _vsn_ is now finished recycling. It will now be post-recycled.
```

## Creating a `recycler.sh` File

If you are archiving on removable media cartridges, create a `recycler.sh` file.

If you are archiving only to disk, do not perform this step.

Determine your site's policy for recycled cartridges. Some sites relabel and reuse the cartridges and some sites remove the cartridges from the automated library to use later for accessing historical files.

The recycler executes the `recycler.sh` script when all the current images from a VSN have been re-archived to another VSN.

The recycler calls the `/opt/SUNWsamfs/scripts/recycler.sh` script with the following arguments:

```
Media type: $1 VSN: $2 Slot: $3 Eq: $4
```

For examples of the script, see “[recycler.sh\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* or view the `/opt/SUNWsamfs/examples/recycler.sh` script, which shows how to relabel a recycled VSN and send mail to the superuser.

## Configuring Recycling for Disk Archive Volumes

If you are archiving to disk, you must edit the `archiver.cmd` file to recycle.

If you are recycling by archive set, you must add archive set recycling directives between the `params` and `endparams` directives.

If you are recycling by library, this step is optional.

### Editing the `archiver.cmd` File

To edit the `archiver.cmd` file, follow the steps described in “[About the `archiver.cmd` File](#)” on page 101.

As an alternative, you can edit the `archiver.cmd` file by using the File System Manager. For more information, see the File System Manager online help.

The following table shows the archive set recycling directives that you can use.

TABLE 16-3 Archive Set Recycling Directives

Directive	Function
<code>-recycle_dataquantity size</code>	Limits the amount of data the recycler schedules for re-archiving to clear a disk volume of useful data. By default, a limit is ignored for disk archive recycling.
<code>-recycle_ignore</code>	Prevents the archive set from being recycled.
<code>-recycle_mailaddr mail-address</code>	Sends recycler messages to the specified email address.
<code>-recycle_mingain percent</code>	Limits recycling of volumes in the archive set by setting the <code>mingain</code> mark for a disk volume. The <code>mingain</code> is expressed as a percentage of the expired data associated with the volume. When the expired data of the volume exceeds the <code>mingain</code> percentage, the recycler begins to recycle the volume. The default is 50%.
<code>-recycle_minobs percent</code>	Limits the recycler's selection of tar files in volume by setting a threshold for the recycler's rearchiving process of disk archive volumes. When the percentage of expired files within an archived tar file on the disk reaches this threshold, the recycler begins moving the current files from the archive into a new tar file. Once all the current files have been moved, the original tar file is marked as a candidate to be removed from the disk archive. The default is 50%.
<code>-rearch_stage_copy copy-number</code>	Sets staging for re-archiving to take place from selected (faster) copies.

For more information about archiver directives, see [Chapter 10, “Configuring the Archiver,”](#) or the `archiver.cmd(4)` man page.

## Recycler Logging for Disk Archives

**EXAMPLE 16-5** Recycler Log File for Disk Archive Files

```

---Archives---   -----Percent-----
---Status-----   Count   Bytes   Use Obsolete Free   Library:Type:VSN
new candidate           0      0      0   41   59  none:dk:disk01
677 files recycled from VSN disk01 (mars:/sam4/copy1)
0 directories recycled from VSN disk01 (mars:/sam4/copy1)

```

## Recycling for Archive Copy Retention

As an alternative to the normal recycling process, you can use the `sam-nrecycler` tool to work with the File System Manager's backup and recovery point features. This tool removes expired archive copies and frees archive volumes to aid in the ability to use SAM-QFS dump files for archive retention. To take advantage of this functionality, you must use this recycler in place of the existing `sam-recycler` command.

The `sam-nrecycler` tool scans file system metadata and SAM-QFS dump files to determine which volumes contain archive images. You can invoke the tool through the `crontab` file at an off-peak time, or at any time using the `sam-nrecycler` command. The `nrecycler` identifies all archive images on a removable media volume or in a disk archive tar file by scanning all file system `.inodes` files and specified SAM-QFS dump files. The `nrecycler` can then determine whether volumes contain any archive images. The space on these volumes can be reclaimed. If a removable media volume does not contain any archive images, relabel the cartridge. If a disk archive tar file does not contain any archive images, remove the tar file from the disk archive directory.

When `sam-nrecycler` detects that a removable media volume contains only free or expired space and is safe to relabel, it invokes the `sam-nrecycler.sh` script. The script can relabel the cartridge using either the original VSN or a new VSN. It can then export the cartridge from the library, or it can perform another user-defined action.

When `sam-nrecycler` detects that a disk archive volume contains only free or expired space, it unlinks the unused disk archive tar file.

You control the actions of the `sam-nrecycler` tool by including directives in the `/etc/opt/SUNWsamfs/nrecycler.cmd` file. You must also specify the path to the directories that contain the SAM-QFS dump files. The list of directories must be complete and all SAM-QFS dump files must be contained in the directory list.

You can also include a `logfile=` directive line in the `nrecycler.cmd` file to specify an `nrecycler` log file. The system writes recycling messages and recycling reports to this file.

For more information about `sam-nrecycler(1M)`, see “`sam-nrecycler(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Advanced SAM-QFS Topics

---

This chapter discusses advanced topics for Sun Storage Archive Manager (SAM-QFS) system administration and usage.

### Using Device Logging

The device-logging facility provides device-specific error information that you can use to analyze certain types of device problems. It can help to determine a failing sequence of events for an automated library, tape drive, or optical drive. The device-logging facility does not collect soft media errors (such as recoverable read errors).

Device-logging messages are written to individual log files. A log file is created for each automated library, for each tape and optical drive device, and for the historian. The log files are located in `/var/opt/SUNWsamfs/devlog`. The name of each log file corresponds to the name of the equipment ordinal.

For example, assume that you have a QFS file system and a single Hewlett-Packard optical library with two optical drives.

The following example shows the `mcf` file.

```
/dev/samst/clt5u0 40 hp hp40 - /etc/opt/SUNWsamfs/hp40_cat
/dev/samst/clt4u0 41 mo hp40 -
/dev/samst/clt6u0 42 mo hp40 -
```

The following example shows the `/var/opt/SUNWsamfs/devlog` file. Device 43 is the historian.

```
# pwd
/var/opt/SUNWsamfs/devlog
# ls
40 41 42 43
#
```

## When to Use the Device Log

The device log can easily generate many log messages, especially when all logging options for all devices are turned on and great deal of device activity occurs. Initially, the device log settings are set to the following default values:

```
err retry syserr date
```

If you suspect a problem exists with one of the devices, consider enabling additional logging events for that device. Also, enable device logging if you are advised to do so by your service provider. In these situations, set the event to `detail`. In extreme cases, your service provider might advise you to set the event to `all` for a device. However, in general, running the system with excessive logging is not practical.

The device log information is collected automatically when the `samexplorer(1M)` command is issued. This process enables the file system service to review any device error information as part of problem analysis activity.

### ▼ How to Enable the Device Log by Using the `samset` Command

- Issue the `samset` command.

```
# samset devlog eq event
```

For `eq`, specify the equipment ordinal of the device for which you want to log messages.

For `event`, specify one or more of the events listed in the `samset(1M)` man page. If you specify more than one event, separate them with space characters.

### ▼ How to Enable the Device Log by Editing the `defaults.conf` File

- 1 Become superuser.
- 2 In the `/etc/opt/SUNWsamfs/defaults.conf` file, add the `devlog` directive.

```
devlog eq event
```

For `eq`, specify the equipment ordinal of the device for which you want to log messages.

For `event`, specify one or more of the events listed in the `samset(1M)` man page. If you specify more than one event, separate them with space characters.

When a QFS file system starts, it sets the event type for each available device to `default`. You can also use the `samset` command to determine the present settings for each device log.

- 3 Save and close the `defaults.conf` file.
- 4 Use the `samd config` command to propagate the `defaults.conf` file changes.  
# `samd config`

## Using Removable Media Files

You can use the `request` command to manually create, write, and read files that do not use the disk cache for buffering the data. Files created in this manner are called *removable media files*.

---

**Note** – The `request` command bypasses the typical functions of the archiver.

---

Removable media files look like typical QFS files in that they have permissions, a user name, a group name, and size characteristics. However, their data does not reside in the disk cache. Therefore, you can create removable media files that are larger than the disk cache and write them to removable media cartridges.

The system creates an inode entry in the `.inodes` file for the file that you specify with the `request` command. The QFS file systems read that information from the inode entry. Multiple removable media files can reside on the same volume.

A removable media file that spans multiple volumes is called a volume overflow file. The *volume overflow* feature enables a single large file to span multiple volumes on multiple cartridges. The volume overflow feature is useful if you have very large files that exceed the capacity of their chosen media.

You must read and write removable media files sequentially. The QFS file system automatically mounts the requested volume if the volume resides in an automated library defined in the `mc f` file.

The presence of a removable media file on a volume prevents that volume from being recycled. The recycler expects that only archived files reside on the particular volume that is assigned for archiving. In addition, removable media files are never archived. Removable media files are not supported over NFS.

## Creating a Removable Media or Volume Overflow File

Use the `tplabel` or `odlabel` command to label a tape or magneto-optical cartridge, respectively. See “[Labeling and Unlabeling Cartridges](#)” on page 64 for details.

To create the file, issue the `request` command. At a minimum, use the following options:

```
request -m media-type -v vsn [vsn/vsn ...] [-l /  
vsn-file] input-file
```

Argument	Definition
<i>media-type</i>	The media type of the removable media cartridge. For information about valid <i>media-type</i> specifications, see “ <a href="#">mcf(4)</a> ” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> .
<i>vsn</i>	The volume serial name (VSN) of the removable media cartridge. If you specify more than one VSN, you are creating a volume overflow file. You can specify up to 256 VSNs for volume overflow files. Use forward slash characters (/) to separate the <i>vsn</i> arguments. The VSNs specified should not be among the volumes that are used in a SAM-QFS environment for automated archiving. Archiving appends the next file to be archived to the end of the current data and moves the EOF label beyond the data.
<i>vsn-file</i>	An input file that contains a list of VSNs. When you have many VSNs, use an input file containing the list of VSNs
<i>input-file</i>	The file to be written to the removable media cartridge. This file must reside in a QFS file system.

**EXAMPLE 17-1** Create a Removable Media File

The following command creates a removable media file:

```
# request -m lt -v aaa rem1
```

For detailed examples of how to create removable media files, see “[request\(1\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

**EXAMPLE 17-2** Create a Volume Overflow File

The following command creates a volume overflow file on three volumes:

```
# request -m lt -v TAPE01/TAPE02/TAPE03 large.file
```

## Using Segmented Files

The SAM-QFS environment supports segmented files. Segmenting files improves tape storage retrieval speed, access, and manageability for very large files. A segmented file can be larger than the physical disk cache. In this case, only part of a segmented file resides on the disk cache at any one time.

The `segment` command enables you to specify the segment size. You cannot set a segment size that is smaller than the current file size.

Segmented files support tape striping. After a file is segmented, it can be striped simultaneously over multiple tape devices, which significantly reduces the time needed to store the file segments. Data access is accelerated by allowing users to retrieve only the desired file segments rather than the entire file.

Segmentation can enhance archiving efficiency because only changed portions of a file are re-archived. Segments of a file can be archived in parallel, and segmented files can be staged in parallel. This increases performance during archiving and retrieving.

Segmentation can be enabled on a file, directory, or entire file system. Segmented files support all other SAM-QFS capabilities.

---

**Note** – The `mmap` function cannot take place on a segmented file. Therefore, a segmented file cannot be an executable binary.

---

The following sections describe how segmented files differ from nonsegmented files. For more information about segmented files, see “[segment\(1\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual* or the “[sam\\_segment\(3\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Archiving Segmented File

For a segmented file, the archivable unit is the segment itself, not the file. All archiving properties and priorities apply to the individual segment and not to the file.

You can stripe a segment by specifying both the `- drives` and `- drivemin` parameters for the archive set in the `archiver . cmd` file. For example, assume that a 100-megabyte segmented file in the file system has segment size of 10 megabytes. If the `archiver . cmd` file defines an archive set with a `- drives 2` directive, this file is archived to two drives in parallel. Segments 1, 3, 5, 7, and 9 are archived using the first drive, and segments 2, 4, 6, 8, and 10 are archived using the second drive.

Only segments that have been modified are archived. Up to four archive copies can be made for each segment. SAM-QFS also supports volume overflow for segments.

---

**Note** – The index of a segmented file contains no user data. It is considered metadata and is assigned to the file system archive set.

---

## Using System Error Facility Reporting

The system error facility (SEF) reporting system captures log sense data from tape devices in an automated library, writes it to a log file, and translates it into human-readable form. This utility consists of the following:

- A log file containing data from tape device log sense pages.
- A command, `sef report`, for writing the log file to `stdout` in a human-readable format. This log file can be used as input to a user-supplied analysis script.

The log sense pages are different from vendor to vendor. For the meanings of the parameter codes, control bits, and parameter values, see the vendor documentation for each specific device.

SEF is not supported for stand-alone tape drives. SEF reporting is most useful for older SCSI-2 devices that do not support the `tapeAlert` functionality. For more information, see “`tapeAlert(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## ▼ How to Enable SEF Reporting

### 1 Become superuser.

### 2 Issue the `mkdir` command to create the SEF directory.

For example:

```
# mkdir /var/opt/SUNWsamfs/sef
```

### 3 Use the `touch` command to create the log file.

You can enable SEF reporting any time after installation by creating the `sefdata` log file. Initially, the SEF log file must be empty. The following command shows the SEF log file being created in the default location.

```
# touch /var/opt/SUNWsamfs/sef/sefdata
```

### 4 Use the `samd stop` and `samd start` commands to initialize SEF reporting.

```
# samd stop
# samd config
# samd start
```

#### a. Wait until the tape drives are idle. Then unload the tape drives.

For *eq*, specify the Equipment Number of the drive.

#### b. Use the `samd unload` command to unload all removable media:

```
# samd unload eq
```

### 5 Repeat steps 1 through 4 for each tape drive.

SEF data is appended to the log file as it is generated.

---

**Note** – SEF reporting is enabled as long as the `sefdata` log file exists. To disable SEF reporting, rename or remove this file.

---

You can configure SEF reporting to log and read log sense data from an alternate location. For more information about reading log sense data from an alternate location, see “`sefreport(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## ▼ How to Generate SEF Report Output

The SEF report output consists of header lines and log sense data.

- 1 **Verify that `/opt/SUNWsamfs/sbin` is in your command path.**
- 2 **Use the `sefreport` command to generate SEF output.**

The following are the most commonly used options with the `sefreport` command:

- The `-d` option. The `-d` option generates additional device information. It writes an additional header line that contains the equipment ordinal and path name to the device for each record. This process makes searching for and locating SEF records that pertain to a specific device easier.
- The `-v` option or the `-t` option. Do not specify the `-t` and `-v` options on the same command line. They are mutually exclusive.
  - The `-v` option generates information in verbose mode. It appends information regarding the equipment ordinal, page code, and VSN to each line of a record. This method enables selecting only those lines that pertain to a specific device or a specific volume.
  - The `-t` option generates log sense output with text descriptions. For each line of log sense data output, the report includes an additional string containing the equipment ordinal, page code, VSN, and parameter code description.

For example, the following SEF command reads the SEF log file from the default location, writes the device number and path name for each device, and generates output:

```
# sefreport -d /var/opt/SUNWsamfs/sef/sefdata > sef.output
```

### Example 17-3 Content of `sef.output` File

```
Record no. 1
Mon Mar 26 11:17:48 2001 STK 9840 1.25 VSN 002981
Eq no. 32 Dev name /dev/rmt/1cbn
PAGE CODE 2
param code control param value
00h 74h 0x0
01h 74h 0x0
02h 74h 0x0
03h 74h 0x0
04h 74h 0x0
05h 74h 0x40050
06h 74h 0x0
PAGE CODE 3
param code control param value
00h 74h 0x0
01h 74h 0x0
02h 74h 0x0
03h 74h 0x0
04h 74h 0x0
05h 74h 0x140
06h 74h 0x0
```

PAGE CODE 6  
param code control param value  
00h 74h 0x0  
Record no. 2  
Mon Mar 26 11:30:06 2001 STK 9840 1.25 VSN 002999  
Eq no. 31 Dev name /dev/rmt/0cbn

PAGE CODE 2  
param code control param value  
00h 74h 0x0  
01h 74h 0x0  
02h 74h 0x0  
03h 74h 0x0  
04h 74h 0x0  
05h 74h 0x1400a0  
06h 74h 0x0

PAGE CODE 3  
param code control param value  
00h 74h 0x0  
01h 74h 0x0  
02h 74h 0x0

03h 74h 0x0  
04h 74h 0x0  
05h 74h 0x190  
06h 74h 0x0

PAGE CODE 6  
param code control param value  
00h 74h 0x0  
Record no. 3  
Mon Mar 26 11:30:23 2001 STK 9840 1.25 VSN 002981  
Eq no. 32 Dev name /dev/rmt/1cbn

PAGE CODE 2  
param code control param value  
00h 74h 0x0  
01h 74h 0x0  
02h 74h 0x0  
03h 74h 0x0  
04h 74h 0x0  
05h 74h 0x18400f0  
06h 74h 0x0

PAGE CODE 3  
param code control param value

00h 74h 0x0  
01h 74h 0x0  
02h 74h 0x0  
03h 74h 0x0  
04h 74h 0x0  
05h 74h 0x1e0  
06h 74h 0x0

PAGE CODE 6  
param code control param value  
00h 74h 0x0

.  
.  
.

For more information about the SEF log file, including its content and format, see “[sefdata\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*. For more information about optional SEF report formats, see “[sefreport\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Managing the SEF Log File

You manage the SEF log file just as you manage any other SAM-QFS log file. You can run a cron job periodically to save the current log file to another location, to delete old SEF files, to create new (empty) SEF files, or to perform other file management tasks.

You can also use the `log_rotate.sh` utility to rotate this log file.

For more information about tools for managing the SEF log file, see the `cron(1M)` or `log_rotate.sh(1M)` man pages.

## SEF sysevent Functionality

In addition to using the SEF log file, you can use the Oracle Solaris sysevent feature to obtain tape drive SCSI log sense error counter pages 2 and 3 for media analysis. By default, the SEF sysevent feature is enabled and set to poll once before unload. The SEF sysevent behavior is controlled by `defaults.conf` and `samset`.

In the `defaults.conf` file, the `sef` parameter can be used to enable SEF sysevent feature by equipment ordinal, or to specify the log sense polling frequency. For more information, see “[defaults.conf\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## ▼ How to Create the SEF sysevent Handler

### 1 Create a `/var/tmp/xx` file.

For example:

```
# !/bin/ksh
echo "$@" >> /var/tmp/xx.dat
exit 0
```

### 2 Make the `/var/tmp/xx` file executable.

```
# chmod a+rx /var/tmp/xx
```

Add the SEF sysevent handler to the `syseventd` file by adding the following information.

```
# syseventadm add -vSUNW -pSUNWsamfs -cDevice -sSEF
/var/tmp/xx "$VENDOR" "$PRODUCT" "$USN" "$REV"
$TOD $EQ_ORD "$NAME" $INQ_TYPE "$MEDIA_TYPE" "$VSN"
```

```
$LABEL_TIME $LP2_PC0 $LP2_PC1 $LP2_PC2 $LP2_PC3 $LP2_PC4
$LP2_PC5 $LP2_PC6 $LP3_PC0 $LP3_PC1 $LP3_PC2 $LP3_PC3
$LP3_PC4 $LP3_PC5 $LP3_PC6 $WHERE $sequence
# syseventadm restart
```

This command creates the `/etc/sysevent/config/SUNW,SUNWsamfs,Device,sysevent.conf` file containing the SEF sysevent handler `/var/tmp/xx` and loads the event handler into the `syseventd` daemon.

**3 To load the SEF sysevent handler, issue the following command:**

```
# kill -HUP syseventd
```

For more information, see “[sefsysevent\(4\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Using the Sun SAM-Remote Software

---

The Sun SAM-Remote client and the Sun SAM-Remote server form an implementation that enables libraries and other removable media devices to be shared between SAM-QFS host systems. Use the SAM-Remote software to configure multiple storage clients that archive and stage files from a centralized tape library or magneto-optical library. For example, if you have host systems on a network that spans a large geographical area, files created in one city can be archived to cartridges in a library located miles away.

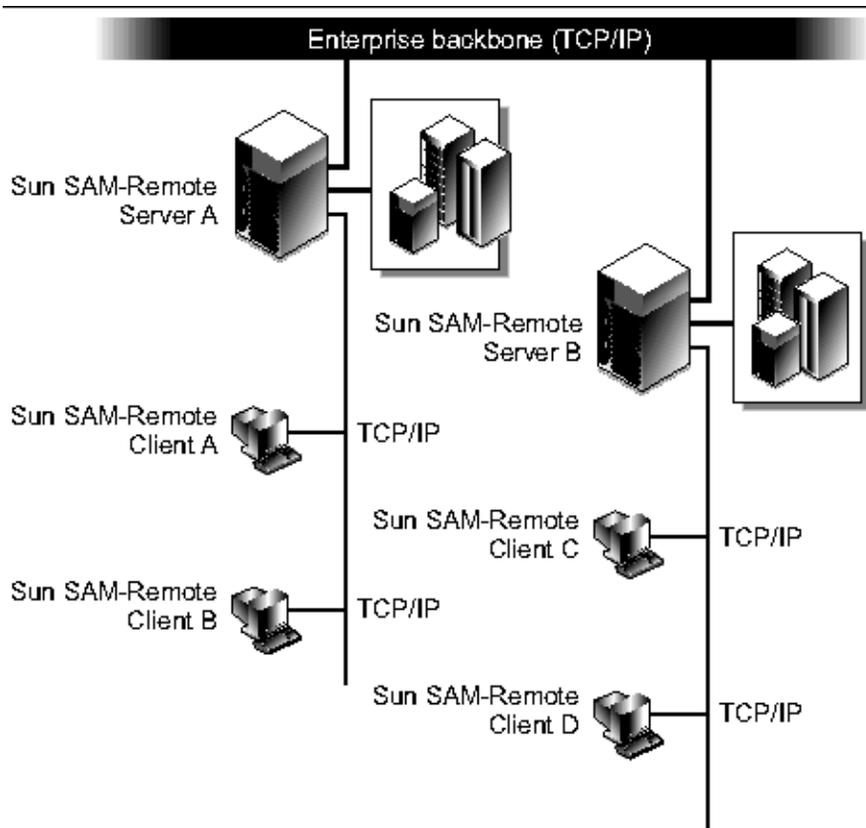
### Sun SAM-Remote Software Overview

Sun SAM-Remote software provides the following advantages:

- Enables you to configure remote sharing of an expensive removable media resource, such as a library, between two or more Sun SAM-Remote clients.
- Enables clients to migrate data to a server.
- Enables multiple SAM-QFS servers to be hosts to one another. In a Sun SAM-Remote environment, the server is the host system that is configured with an equipment type of `ss` in the `mc f` file.

You can configure the Sun SAM-Remote server and clients to provide multiple archive copies between two or more Oracle Solaris host systems. For example, you can configure two Oracle Solaris systems running SAM-QFS software as both Sun SAM-Remote servers and Sun SAM-Remote clients to each other. Benefits of this configuration include the ability to create local copies for each server with an additional archive copy of data on the other server. File systems can be shared between servers using standard NFS. In the event of a loss of access to the local library, Sun SAM-Remote software would automatically retrieve file data from the archive copy. Users of both servers would have uninterrupted access to their data even if their primary storage library were unavailable.

FIGURE 18-1 Two Sun SAM-Remote Servers, Each With Two Clients



## System Requirements

Before attempting to configure a Sun SAM-Remote environment, make sure that your environment includes the following software and hardware:

- SPARC or x64 systems with licensed, installed, and operable SAM-QFS archive management software packages.
- Host systems with identical SAM-QFS software revision levels and identical patch collections installed. If some host systems have to be upgraded, see [Chapter 10, “Upgrading Sun QFS and SAM-QFS,” in \*Sun QFS and Sun Storage Archive Manager 5.3 Installation Guide\*](#).
- One or more host systems to act as the Sun SAM-Remote server with at least one SAM-QFS file system.
- A network connection running a TCP/IP connection between the clients and the server.

## Software Limitations

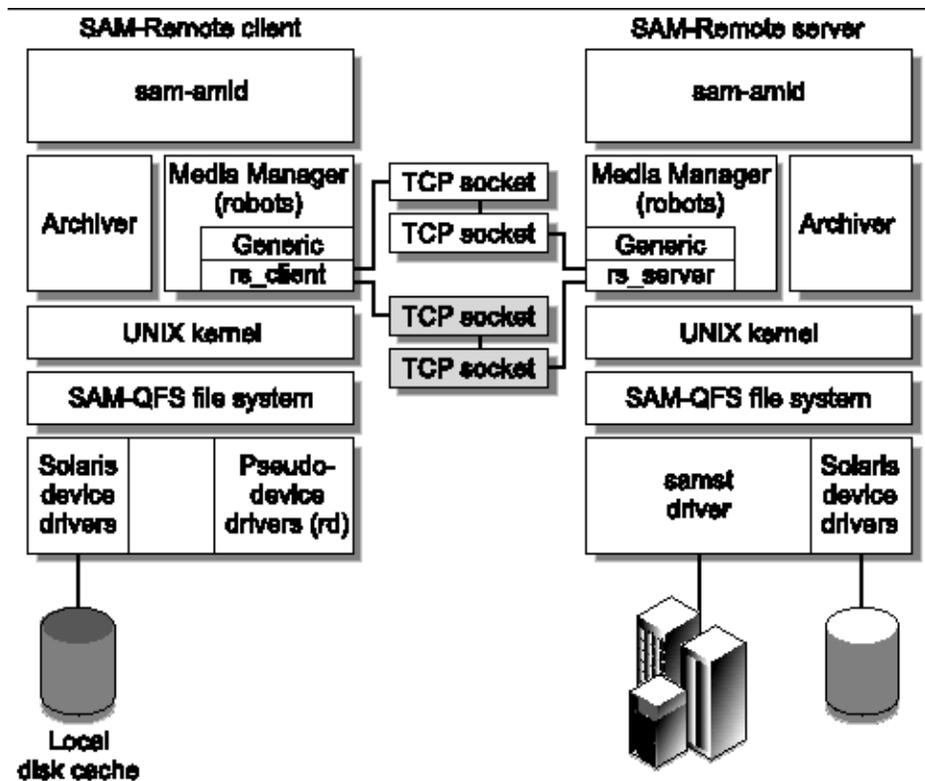
The SAM-QFS software treats cartridges in a remote library no differently than it treats cartridges in a local library. The following information, however, indicates the limits of Sun SAM-Remote software:

- Before recycling media using Sun SAM-Remote, be sure to thoroughly test your environment. For more information, see [“Recycling With the Sun SAM-Remote Software” on page 224](#).
- Only one daemon on a Sun SAM-Remote client can communicate to the Sun SAM-Remote server.
- SAM-QFS software, and therefore Sun SAM-Remote, cannot operate on Sun QFS clients in a shared Sun QFS file system. When running on a server that is a metadata server for some Sun QFS file systems and a client for other Sun QFS file systems, SAM-QFS software and Sun SAM-Remote operate only on the file systems for which that server is a metadata server.

## Client and Server Interactions

Sun SAM-Remote clients interact with the Sun SAM-Remote server using a TCP/IP connection. The network between Sun SAM-Remote clients can be any network type supported by the Oracle Solaris OS, such as Ethernet, Fast Ethernet, or Fibre Channel.

FIGURE 18-2 Sun SAM-Remote Server and Client Interactions



## Sun SAM-Remote Server Overview

The Sun SAM-Remote server consists of a full-capability SAM-QFS storage management host and a Sun SAM-Remote server daemon that defines libraries to be shared among the clients. At least one SAM-QFS file system must be configured on the Sun SAM-Remote server.

You define a host system as a Sun SAM-Remote server by adding a line in the server system's `/etc/opt/SUNWsamfs/mcf` file with an equipment type of `ss`. You must provide a *unique* family set name for each server. Up to ten clients can be configured per daemon. To configure more than ten clients, add an additional remote server entry in the `mcf` file for each ten clients that you want to configure. For more information about the server daemon, see “[sam-remote\(7\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Sun SAM-Remote Client Overview

The Sun SAM-Remote client is a SAM-QFS host system that establishes a Sun SAM-Remote client daemon containing a number of pseudo-devices.

You define a host system as a Sun SAM-Remote client by adding a line in the client system's `/etc/opt/SUNWsamfs/mcf` file with an equipment type of `sc`. For more information about the client daemon, see “[sam-remote\(7\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

A pseudo-device defines a network connection to an actual removable media device on the Sun SAM-Remote server. Pseudo-devices have an equipment type of `rd`, which is an abbreviation for *remote device*. You define the pseudo-devices in the Sun SAM-Remote client's `/etc/opt/SUNWsamfs/mcf` file. The Sun SAM-Remote daemon and pseudo-devices are associated with one particular server.

The Sun SAM-Remote daemon supports an unlimited number of pseudo-devices for each client. The actual number of pseudo-devices to be used by the client is configurable. When determining how many pseudo-devices should be configured per client, think of these devices as the number of simultaneous data transfers that can occur between the client and the server. As more pseudo-devices are defined, the possibility of increasing the total network traffic load increases. As the system administrator, determine the actual number of pseudo-devices needed for the system.

## Interaction Between the Sun SAM-Remote Server and the Sun SAM-Remote Client

The Sun SAM-Remote server daemon, `sam-serverd`, listens for the clients on port 1000. You can configure a different port in the Oracle Solaris `/etc/services` directory with a service name of `rmt sam`. When a Sun SAM-Remote client connects to the Sun SAM-Remote server, the `sam-serverd` daemon establishes a connection on another port and communicates this port number to that client, using the defined port. The socket size is passed to the client. The socket size is configurable and is described in more detail in “[Configuring the Sun SAM-Remote Software](#)” on page 214.

## Library Catalogs

The Sun SAM-Remote library catalog is a subset of the catalog located on the Sun SAM-Remote server. The client catalog is updated in real time. The slots allotted to a Sun SAM-Remote client catalog are controlled only by the Sun SAM-Remote server.

After initialization, the system builds a client catalog and passes it to the Sun SAM-Remote client based on information from the Sun SAM-Remote server catalog file. After the connection between the host and client is established, media available to the client is flagged as available. If the connection between the client and server is lost, the media on the client side is flagged as unavailable. You can view the media availability through the `samu v display`. The information that appears in the `samu v display` on the client is a subset of what appears in the `v display` on the server. A good practice is to access the media catalog through the `samu v display` on Sun SAM-Remote server. For more information about the Sun SAM-Remote server client file, see

“[Configuring the Sun SAM-Remote Software](#)” on page 214. For information about the `samu` operator utility, see [Chapter 13, “Using the samu Operator Utility,” in \*Sun QFS File System 5.3 Configuration and Administration Guide\*](#).

Changes to the catalog are passed between hosts as necessary. Any changes in the server catalog that involve a media type associated with a client are passed to the client, and the client catalog is updated.

## Archiving

Sun SAM-Remote archive processing is the same as SAM-QFS archive processing. The Sun SAM-Remote client makes a mount request to be added to the server's mount request table. The client then waits for the server to respond with a message indicating that the media is mounted. Archiving begins when the media is available.

## Data Integrity Validation in Sun SAM-Remote Software

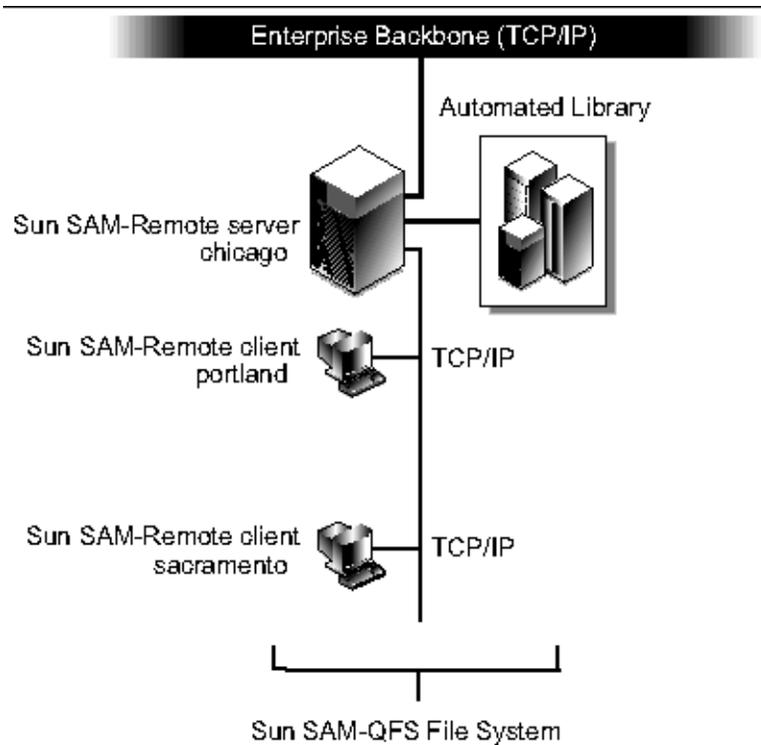
In order to be able to use the DIV feature, the Sun SAM-Remote server as well as the client must have `div` enabled in the `defaults.conf` file. For more information about the DIV feature, see [Chapter 13, “Data Integrity Validation in SAM-QFS.”](#)

# Configuring the Sun SAM-Remote Software

This section explains how to perform an initial configuration of the Sun SAM-Remote server and client software.

In the following sample configurations, the SAM-QFS file systems on `portland` and `sacramento` are clients of the Sun SAM-Remote server on `chicago`.

FIGURE 18-3 Example Sun SAM-Remote Configuration



The following procedures explain how to configure the Sun SAM-Remote software on a Sun SAM-Remote server and on one or more Sun SAM-Remote clients. The procedures use configuration shown in [Figure 18-3](#).

- Perform the procedures in the order described.
- You must have superuser access to the server system on which the Sun Sun SAM-Remote software is to be installed.
- You must have superuser access to the client system or systems on which the Sun SAM-Remote software is to be installed.
- The same release and revision level of SAM-QFS software must be installed on all client and server hosts in the Sun SAM-Remote environment.
- The same patch collection must be installed on all client and server hosts in the SAM-Remote environment.

## ▼ Step 1: Log In to the Potential Server and Client Hosts

- 1 Log in to the Sun SAM-Remote server as the superuser.
- 2 Log in to the Sun SAM-Remote clients as the superuser.

## ▼ Step 2: Verify Client and Server Software

Follow these steps on each system to be configured as part of a Sun SAM-Remote environment.

- 1 Issue the `pkginfo -l` command and examine the output.

```
# pkginfo -l SUNWsamfsr
PKGINST: SUNWsamfsr
  PKGINST: SUNWsamfsr
  NAME: Sun SAM and Sun SAM-QFS software Solaris 11 (root)
CATEGORY: system
  ARCH: i386
VERSION: 5.3.0,REV=5.11.2011.10.27
BASEDIR: /
  VENDOR: Sun Microsystems, Inc.
  DESC: Storage and Archive Manager File System
  PSTAMP: build34-mn20111027150903
INSTDATE: Oct 28 2011 02:57
HOTLINE: Please contact your local service provider
STATUS: completely installed
FILES: 641 installed pathnames
       7 shared pathnames
       23 linked files
       79 directories
       208 executables
       1 setuid/setgid executables
       96696 blocks used (approx)
```

```
chicago# pkginfo -l SUNWsamfsu
PKGINST: SUNWsamfsu
  NAME: Sun SAM and Sun SAM-QFS software Solaris 11 (usr)
CATEGORY: system
  ARCH: i386
VERSION: 5.3.0,REV=5.11.2011.10.27
BASEDIR: /
  VENDOR: Sun Microsystems, Inc.
  DESC: Storage and Archive Manager File System
  PSTAMP: build34-mn20111027150917
INSTDATE: Oct 28 2011 02:58
HOTLINE: Please contact your local service provider
STATUS: completely installed
FILES: 57 installed pathnames
       9 shared pathnames
       20 directories
       19 executables
       12584 blocks used (approx)
```

```
chicago#
```

The output shows that the server (chicago) is running software version 5.3. Any systems included in an environment with this server must also run version 5.3.

## 2 Issue the `showrev -p` command and examine the output.

```
chicago# showrev -p | grep SUNWsamfs
```

```
chicago#
```

The output shows that the server has no SAM-QFS patches installed. Any systems included in an environment with this server must also run version and patch level.

If you need to perform any software upgrades, see [Chapter 10, “Upgrading Sun QFS and SAM-QFS,”](#) in *Sun QFS and Sun Storage Archive Manager 5.3 Installation Guide*.

## ▼ Step 3: Edit the mcf Files on Each Client

The `mcf` file defines a file system. This procedure adds the definition of the host as a Sun SAM-Remote client.

### 1 From the Sun SAM-Remote server, stop the SAM-QFS functions.

### 2 Issue the `samcmd` command to idle each removable media. For more information about the `samcmd` command, see “`samcmd(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

```
# samcmd idle eq1
# samcmd idle eq2.
.
# samcmd idle eqn
```

`eq` specifies the equipment ordinal of the removable media drive as defined in the `mcf` file.

As an alternative, you can also use the `samu` operator utility to idle the drives. For information, see [Chapter 13, “Using the `samu` Operator Utility,”](#) in *Sun QFS File System 5.3 Configuration and Administration Guide*.

---

**Note** – All drives must be idle before you issue the next command, `samd stop`, so that the archiver, stager, and other processes can complete current tasks, and cartridges can be unloaded and put into their storage slots.

---

### 3 Issue the `samd` command with its `stop` option to stop the `sam-amld` daemon and its child processes.

```
# /opt/SUNWsamfs/sbin/samd stop
```

### 4 On each client, open the `/etc/opt/SUNWsamfs/mcf` file.

**5 Define each system as a client of the Sun SAN-Remote server.**

The example code shows the client `portland`'s `mcf` file after it is edited to make `portland` a client of the Sun SAM-Remote server `chicago`.

**6 Open the `mcf` file of another client.**

In the example, the client `sacramento`'s `mcf` file is edited.

**7 Copy the last set of lines from the first client's `mcf` file to next client's `mcf` file.**

In the example, copy the last set of lines from `portland`'s `mcf` file to `sacramento`'s `mcf` file.

**8 Save and close the `mcf` files.****Example 18-1** `mcf` Files on the Clients

```
# mcf file on portland
#
# Define a Sun QFS file system
#
# Equipment                               Eq Eq Family Dev Additional
# Identifier                               Ord Ty Set   St Parameters
# =====                               === == ===== == =====
samfs1                                     10  ms samfs1 on
/dev/dsk/c7t60003BA13F71500048EDCE5F00048F2Dd0s6 11  md samfs1 on
/dev/dsk/c7t60003BA13F71500048EDCE23000B24C2d0s6 12  md samfs1 on

# Define Sun SAM-Remote Client portland to Sun SAM-Remote server chicago
#
/etc/opt/SUNWsamfs/rmt200 200 sc chicagoss on /var/opt/SUNWsamfs/catalog/tcat
/dev/samrd/rd0           201 rd chicagoss on
/dev/samrd/rd1           202 rd chicagoss on
```

The `mcf` entry on the client consists of a single-line entry for the Sun SAM-Remote client and a pseudo-device entry, indicated by the `rd` equipment type, for each device you want to configure. A pseudo-device defines a network connection to an actual device on the Sun SAM-Remote server. Each entry uses the fields listed in the following table.

Field	Description
Equipment Identifier	The full path name of the client configuration file. You create the client configuration file in Step 4.
Eq Ord	The number that identifies this client system. Use this number to specify the <i>eq</i> parameter.
Eq Ty	A two-letter mnemonic that identifies the type of the client system.

Field	Description
Family set	The family set name of the daemon to be used on this server. A Sun SAM-Remote server can have one server daemon per client. <b>Note</b> – The family set name on the Sun SAM-remote client must match the family set name on the Sun SAM-remote server.
Dev St	Device state: on or off
Additional parameters	An optional field. In this example, it is the path to the catalog file.

## ▼ Step 4: Create a Sun SAM-Remote Client Configuration File

A Sun SAM-Remote client's configuration file contains one entry: the name of the Sun SAM-Remote server.

- 1 **On each client, create the file to be used as the Sun SAM-Remote client configuration file in the location that you specified in the mcf file.**

For example:

```
portland# vi /etc/opt/SUNWsamfs/rmt200
```

- 2 **Type the name of the Sun SAM-Remote server.**

The following example shows the client configuration file for the client portland, specifying that its Sun SAM-Remote server is the chicago system.

The following example shows a client configuration file.

```
portland# cat /etc/opt/SUNWsamfs/rmt200
chicago
```

## ▼ Step 5: Edit the Server's mcf File

- 1 **On the Sun SAM-Remote server, edit the /etc/opt/SUNWsamfs/mcf file.**
- 2 **Add the entries that define the SAM-QFS file systems and this system as the Sun SAM-Remote server.**

You must have at least one SAM-QFS file system. The following example shows the mcf file on chicago.

The following example shows mcf files on the server

```
# mcf file on Sun SAM-Remote server chicago:
# Define a SAM-QFS file system
#
```

```

# Equipment                               Eq Eq Family Dev Additional
# Identifier                               Ord Ty Set   St   Parameters
# =====                               === == ===== == =====
samfs1                                     10 ms samfs1 on
/dev/dsk/c6t600A0B80004850A600000F8048EF90AdD0s6 11 md samfs1 on
/dev/dsk/c6t600A0B800048505600000E9D48EF91EE0s6 12 md samfs1 on

# Define a tape library that client portland can use:
/dev/samst/c4t500104F0009C2F6Fu0 100 sn rb100 on /var/opt/SUNWsamfs/catalog/rb100.cat
/dev/rmt/0cbn                       101 li rb100 on
/dev/rmt/1cbn                       102 li rb100 on

# Define Sun SAM-Remote server chicago
#
/etc/opt/SUNWsamfs/rmt200           50 ss chicagoss on

```

---

**Note** – The family set name on the Sam-Remote server must match the family set name on the Sam-Remote client.

---

## ▼ Step 6: Create the Server's Configuration File

The Sun SAM-Remote server configuration file defines the disk buffer characteristics and media to be used for each client. Ten clients can be configured per server daemon. To support more clients, configure another Sun SAM-Remote server daemon.

- 1 On the server, create the Sun SAM-Remote server configuration file.
- 2 Add definitions of each client, using the following format:

```

client-name
[ parameter1 ]
  media
    eq media-type regex
    [eq media-type regex ]
    [. . .]
  endmedia

```

The following table describes the fields that are available for the clients.

Field	Definition and Requirements
<i>client-name</i>	Identifies the system to be served by this invocation of the Sun SAM-Remote daemon. Use the network name, its IP address, or a fully qualified domain name.  The first character in <i>client-name</i> must be the first character in the line.

Field	Definition and Requirements
<i>parameter</i> (optional)	<p>Defines an attribute of the client, using a <i>keyword = value</i> pair. For example, you can use the <i>parameter</i> to specify the network block size to be used by this client's socket in kilobytes. The format for this parameter: <code>net_blk_size=size</code> where <i>size</i> is an integer from <math>4 \leq \text{size} \leq 64</math>. The default is 4, which specifies 4096 bytes.</p> <p>The line containing the <i>parameter</i> must start with space or tab characters.</p>
media and endmedia	<p>Keywords that contain the media definitions. The definitions within these two keywords define the media archive volumes that a client can use.</p> <p><b>Note</b> – These keywords are required and must be <i>indented</i> with space or tab characters.</p>
<i>eq media-type</i> ( <i>regex</i> )	<p>Defines a media archive volume that this client can use. Enclose the <i>regex</i> data with parentheses. Because network-attached libraries have mixed media, specify each media type on a separate line.</p> <p><b>Note</b> – Use the space or tab characters to <i>indent</i> the media definitions.</p> <ul style="list-style-type: none"> <li>▪ <i>eq</i>: Equipment Number as shown in the <code>mcf</code> file.</li> <li>▪ <i>media-type</i>: Two-character specific media type, such as <code>li</code>. For information about valid media types, see “<code>mcf(4)</code>” in <i>Sun QFS and Sun Storage Archive Manager Reference Manual</i> but do not use the generic media type.</li> <li>▪ (<i>regex</i>): The volume serial names (VSN) of the cartridges to which the files are archived, expressed as an extended regular expression. For information about extended regular expressions, see the <code>egrep(1)</code> man page. For information about regular expressions, see the <code>regcomp(3C)</code> man page.</li> </ul>

For example, the following is a valid media type definition:

```
media
  100 li (VSN1)
  100 li (VSN2)
endmedia
```

**Note** – Do not allow the same physical media cartridges to be used by more than one client. Also, if the Sun SAM-Remote server has its own file system outside of the Sun SAM-Remote environment, a cartridge should not be used by both the client and the server.

The following example shows the server configuration file, `/etc/opt/SUNWsamfs/rmt200` for the Sun SAM-Remote server `chicago`. This file defines clients `portland` and `sacramento`.

```
# Sun SAM-Remote server config file /etc/opt/SUNWsamfs/rmt200
#
portland
  media
    100 li (100031|100032|100034|100035|100037|100038)
    100 li (200001|200002|200003|200004|200005|200006)
  endmedia
```

```
#
#
sacramento
  media
    100 li (300141|300142|300143|300145|300146|300147)
    100 li (400001|400002|400003|400005|400006|400007)
  endmedia
```

## ▼ Step 7: Enable Archiving

- 1 Verify the `archiver.cmd` file on each client. Depending on your configuration, you might need to perform the following tasks:

- a. Make sure that the VSNs that are defined in the server configuration file are assigned to the correct archive sets.
- b. Remove the following directives if they apply to archive sets to be archived in the library connected to the Sun SAM-Remote server:
  - `-tapenonstop`
  - `-offline_copy direct`

- 2 Start the SAM-QFS processes on the server and on the clients. Issue the following command on the clients and the server:

```
server# samd start
```

- 3 Check the status of the Sun SAM-Remote connections.

- a. Issue the `samu` command on the server and the clients.

For more information, see “`samu(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual* or Chapter 13, “Using the `samu` Operator Utility,” in *Sun QFS File System 5.3 Configuration and Administration Guide*.

- b. On each client, view the the `samu` utility's `s` device status display.

The following example shows the status `s` display on the Sun SAM-Remote client `portland`. The device type `sc` identifies the Sun SAM-Remote client. The message after that line indicates that a connection with the server `chicago` has been established.

```
Device status samcmd      5.3.0 20:44:09 Jul 30 2009
samcmd on portland

ty   eq state  device_name                fs status
sc   200 on    /etc/opt/SUNWsamfs/rmt200  200  -----r
      Remote server 10.1.229.92 connected
rd   201 on    /dev/samrd/rd0             200  -----
rd   202 on    /dev/samrd/rd1             200  -----
```

```
hy    203 on    historian                203 -----
```

**c. On the server, view the samu utility's s device status display.**

The following example shows the samu status s display on the Sun SAM-Remote server chicago. The device type ss identifies the Sun SAM-Remote server.

```
Device status samcmd      5.3.0 20:40:05 Jul 30 2009
samcmd on chicago

ty    eq state  device_name                fs status
sk    1 on     /etc/opt/SUNWsamfs/SL500_SAM  1 m-----r
      running
li    2 on     /dev/rmt/0cbn                1 -----p
      empty
li    3 on     /dev/rmt/1cbn                1 -----p
      empty
ss    50 on    /etc/opt/SUNWsamfs/rmt200    50 -----o-r
hy    51 on    historian                    51 -----
```

**d. On the server, view the the samu utility's R SAM-Remote display.**

The following example shows the samu Sun SAM-Remote R display from the Sun SAM-Remote server chicago.

```
Remote server eq: 50    addr: 00003858 samu      5.3.0 20:41:38 Jul 30 2009
```

message:

```
Client IPv4: jimmy 192.10.10.3                port - 5000
          client index - 0 port - 0 flags - 0004
```

```
Client IPv4: portland 10.1.229.97            port - 5000
          client index - 1 port - 32848 flags - 0005 connected
```

If the Sun SAM-Remote configuration includes several clients, press the CONTROL-f key sequence to scroll through each client. Each client is identified by name and by its cLiEnt index field, an integer 0 - 9, which indicates its order in the possible 10 clients defined for this server daemon. The network block size, maximum file size, and minimum file size are listed in bytes. Flags indicate the state of the connection.

Flag	Number	Definition
0x00000000	0004	No connection.
0xc0000000	0005	A connection has been established.

**4 On the server, use the samu utility's v automated library catalog display to ensure that the Sun SAM-Remote catalog is available on the clients. From samu command prompt, issue the following command:**

```
:v eq
```

For *eq*, specify the equipment ordinal of the Sun SAM-Remote client daemon as defined in the *mc f* file.

The example shows the volumes that *portland* can access.

```
Robot VSN catalog by slot : eq 200 samu 5.3.0 Wed May 02 15:24:13
count 32
slot access time count use flags ty vsn
1 2003/01/02 10:40 0 0% -il-o-b-R-U- at 000032
2 2003/01/02 11:41 0 0% -il-o-b-R--- at 000034
3 2003/01/02 12:42 170 91% -il-o-b----- at 000035
4 2003/01/02 13:43 20 7% -il-o-b----- at 000037
5 2003/01/02 14:44 0 0% -il-o-b----- at 000038
6 2003/01/02 13:41 0 0% -il-o-b----- at 000031
```

- 5 From each client, issue the archiver command and its -A option to verify that archiving is taking place from the client to the server.**

This command writes a listing from the archiver, including the VSNs from the server. For information about this command, see “[archiver\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Recycling With the Sun SAM-Remote Software

This section contains information about recycling with Sun SAM-Remote. You should recycle in a Sun SAM-Remote environment only under the very specific circumstances described here.

Because the recycling process involves freeing space on cartridges for more data, the recycler could destroy important data on archive cartridges if the recycling process is not configured properly.

---

**Note** – These restrictions are not enforced by the SAM-QFS software.

---

To avoid data loss, be sure to adhere to the following restrictions:

- Before using the recycler in a Sun SAM-Remote environment you must have a complete understanding of each step of the recycler. Executing commands in the wrong order, or on the wrong system, can result in an irreversible loss of data. Make sure you have analyzed a command's actions before executing any command, such as *tplabel*, that can delete data on the Sun SAM-Remote client or the Sun SAM-Remote server.
- Recycling activities on the Sun SAM-Remote server and the Sun SAM-Remote client must not overlap. The result could be accidental relabeling of cartridges and irreversible loss of data.
- You must not recycle cartridges that contain removable media files.

- In a Sun SAM-Remote client and server environment, the client and server are unaware of each other's file systems, data files, and inode files. Therefore, the server and the client each must have exclusive use of a certain set of cartridges. Neither must ever use the other's cartridges.

You can prevent accidental recycling of VSNs used by Sun SAM-Remote clients by creating a `no_recycle` list in the Sun SAM-Remote server's `/etc/opt/SUNWsamfs/recycler.cmd` file. However, be careful of using the `chmed` command's `+c` option on volumes in a `no_recycle` list. When you use this command to set the recycling flag (`+c`) on a volume, that action overrides the `no_recycle` list in the `/etc/opt/SUNWsamfs/recycler.cmd` file.

- You must not attempt to recycle volumes on the Sun SAM-Remote server and Sun SAM-Remote client on the same day.

Recycling in a Sun SAM-Remote environment is allowed to occur only if the following conditions are present:

- Each VSN in the system is used by one client system or by the server. There cannot be files from multiple systems on any VSN.
- No Sun SAM-Remote client has catalog entries for any VSNs other than those VSNs containing that client's archive images. The *regex* values in the server configuration file's media definition lines (the *eq media-type regex* lines) must agree with the volumes specified in the client catalog. In addition, the *regex* specifications in the client catalogs cannot specify the same volumes.
- The archiving is performed on an archive set basis. When you are using Sun SAM-Remote, recycling must be performed by archive set, not by library.

The following sections describe two methods for enabling recycling using a Sun SAM-Remote client and server.

## Recycling in a Sun SAM-Remote Environment (Method 1)

The procedures in this section describe one method for enabling recycling in a Sun SAM-Remote environment. The procedures assume a setup in which the server is named `sky` and the client is named `zeke`.

---

**Note** – To use the recycler in a Sun SAM-Remote environment, you must follow this procedure completely and you must test your configuration to verify that recycling is taking place correctly.

---

### Configuration Files for the Server

The server must have Sun SAM-Remote configuration information in its `mc f` file and in its server configuration file. The following examples show these files.

**EXAMPLE 18-2** mcf File on Server jimmy

```

# This is the mcf file for the server (jimmy).
# The server parameters file (rmt2000) points
# back to the correct automated library's equipment number
# (1000) for the SL8500 tape library.
#

# Equipment                               Eq Eq Family Dev Additional
# Identifier                               Ord Ty Set   St Parameters
# =====                               === == ===== == =====
samfs1
/dev/dsk/c6t600A0B80004850A600000F8048EF90ADd0s0 100 ma samfs1 on
/dev/dsk/c6t600A0B800048505600000E9D48EF91EEd0s6 101 mm samfs1 on
/dev/dsk/c7t60003BA13F71500048EDCE5F00048F2Dd0s6 102 mr samfs1 on
/dev/dsk/c7t60003BA13F71500048EDCE23000B24C2d0s6 103 mr samfs1 on
/dev/dsk/c7t60003BA13F71500048EDCE50000ADAECd0s6 104 mr samfs1 on

samfs2
/dev/dsk/c7t60003BA13F71500048EDCE720001B17Fd0s6 200 ms samfs2 on
/dev/dsk/c7t60003BA13F71500048EDCE7200014BEAd0s6 201 md samfs2 on
/dev/dsk/c7t60003BA13F71500048EDCE7200014BEAd0s6 202 md samfs2 on

# SL8500
/etc/opt/SUNWsamfs/T10K      1000  sk    T10K   on      /etc/opt/SUNWsamfs/T10K_cat
/dev/rmt/4cbn                1001  ti    T10K   on
/dev/rmt/5cbn                1002  ti    T10K   on
/dev/rmt/0cbn                1003  ti    T10K   on
/dev/rmt/1cbn                1004  ti    T10K   on
/dev/rmt/6cbn                1005  ti    T10K   on
/dev/rmt/7cbn                1006  ti    T10K   on
/dev/rmt/2cbn                1007  ti    T10K   on
/dev/rmt/11cbn              1008  ti    T10K   on
/dev/rmt/10cbn              1009  ti    T10K   on
/dev/rmt/12cbn              1010  ti    T10K   on

# Define Sun SAM-Remote server jimmy
/etc/opt/SUNWsamfs/rmt2000 2000 ss jimmy on

```

**EXAMPLE 18-3** Server Configuration File on Server jimmy

```

# Server configuration file /etc/opt/SUNWsamfs/rmt2000 on jimmy.
# The eq of the automated library MUST match the eq of the
# automated library that you want to use in the mcf file.
tera
  media
    1000 ti 00002[0-9]
  endmedia

```

## Configuration Files for Clients

The client must have Sun SAM-Remote configuration information in its mcf file and in its client configuration file. The following examples show these files.

**EXAMPLE 18-4** mcf File on Client tera

```

# mcf file for client (tera)
#

```

**EXAMPLE 18-4** mcf File on Client tera (Continued)

```

# Equipment                               Eq Eq Family Dev Additional
# Identifier                               Ord Ty Set   St   Parameters
# =====                               === == ===== == =====
samfs1                                     100 ms samfs1 on
/dev/dsk/c7t60003BA13F71500048EDCE5F00048F2Dd0s6 101 md samfs1 on
/dev/dsk/c7t60003BA13F71500048EDCE23000B24C2d0s6 102 md samfs1 on
/dev/dsk/c7t60003BA13F71500048EDCE50000ADAECd0s6 103 md samfs1 on

# Define a L500 with 2 drives
/dev/samst/c4t500104F0009C2F6Fu0 300 rb L500 on
/dev/rmt/0cbn                       301 li L500 on
/dev/rmt/1cbn                       302 li L500 on

# Define tera as a Sun SAM-Remote client using jimmy as the server
/etc/opt/SUNWsamfs/rmt2000          2000 ss jimmy on
/dev/samrd/rd0                     2001 rd jimmy on
/dev/samrd/rd1                     2002 rd jimmy on
/dev/samrd/rd2                     2003 rd jimmy on
/dev/samrd/rd3                     2004 rd jimmy on
/dev/samrd/rd4                     2005 rd jimmy on
/dev/samrd/rd5                     2006 rd jimmy on
/dev/samrd/rd6                     2007 rd jimmy on
/dev/samrd/rd7                     2008 rd jimmy on

```

**EXAMPLE 18-5** Client Configuration File on Client tera

```

# cat /etc/opt/SUNWsamfs/rmt2000
# File /etc/opt/SUNWsamfs/rmt2000 on Sun SAM-Remote client tera: jimmy

```

## ▼ How to Configure Recycling (Method 1)

The procedure for configuring the recycling process includes a test for archiving and recycling. Because of the testing period, this procedure can take a day or two to complete, depending on how frequently files are archived and recycled.

---

**Note** – Do not use the `chmed` command on the server to set the recycling flag (+c) for a client VSN. That action overrides the `no_recycle` list in the `/etc/opt/SUNWsamfs/recycler.cmd` file on the server.

---

Before starting the procedure, read [“About Recycling” on page 187](#). Using the recycler in a Sun SAM-Remote environment requires a complete understanding of the steps in the recycling process. If you have not already familiarized yourself with the recycling process, do so now.

**1 Make sure that the Sun SAM-Remote client and server are configured properly and that archiving is occurring.**

For more information, see [“Configuring the Sun SAM-Remote Software” on page 214](#), which contains detailed information about configuring the Sun SAM-Remote client and server. That procedure includes steps for ensuring that archiving is taking place.

**2 Edit the archiver . cmd file on the client system and add recycling directives.**

In this example, the recycling is performed by archive set, not by library. The directives specifying that recycling be done by archive set must appear in the archiver . cmd file.

The following example shows the archiver . cmd file on client zeke. This file has been edited to communicate with the recycler.

```
# This is file /etc/opt/SUNWsamfs/archiver.cmd
# on Sun SAM-Remote client zeke.
#
archivemeta = off
archmax = li 12G

fs = samfs1
logfile = /var/adm/samfs1.archiver.log
no_archive tmp
all .
    1 -norelease 10m
    2 -norelease 10m

params
allsets -sort path -offline_copy stageahead -reserve set
allsets -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 2 -dataquantity 100G
allsets -ignore
allsets.1 -startage 8h -startsize 10G -drives 2
allsets.2 -startage 24h -startsize 20G -drives 2 -archmax 24G
endparams

vsns
all.1 li ^10.*
all.2 li ^20.*
endvsns
```

The directives shown in the example do the following:

- The `-recycle_hwm` directive sets the library's high-water mark for the archive set. When the utilization of the VSNs exceeds this percentage, recycling of the archive set begins.
- The `-recycle_ignore` directive is inserted only *temporarily*. This directive prevents recycling from occurring until you have configured and tested your environment. You can remove this directive in a later step.
- The `-recycle_mingain` directive is set high to ensure efficiency by limiting the amount of work needed to regain space.
- The `-recycle_vsncount2` directive specifies that the recycler drain two VSN at a time. Do not let recycling overwhelm the system.

**3 Edit the `recycler.cmd` file on the client and specify a log file to receive recycling log output.**

The following example shows the `recycler.cmd` file on client `zeke`, which has been edited to specify a `recycler` log file.

```
#
# This is the /etc/opt/SUNWsamfs/recycler.cmd file
# on client zeke.
#
logfile = /var/adm/recycler.log
```

**4 Verify that the `archiver.cmd` file on the server is written to specify recycling by archive set.**

When using Sun SAM-Remote, you must specify that recycling be performed on an archive set basis, not by library. The directives specifying that recycling be done by archive set must appear in the `archiver.cmd` file.

The following example shows the `archiver.cmd` file on server `sky`. This file specifies archiving by archive set.

```
# This is the archiver.cmd for the server (sky).
#
# Number of drives: 10
# Number of Mounted Filesystems: 1
# Number of Tests per Filesystem: 1
# Number of Archive Copies per Test: 2
#wait
#trace = /var/opt/SUNWsamfs/trace/archiver all
logfile = /var/opt/SUNWsamfs/log/archiver
interval = 1m
no_archive .
archmax = at 5G
drives = adic1 6
fs = samfs1
1 4h
testset testdir0
1 1m
2 1m
allsam1 .
1 1m
2 1m
params
allsam1.1 -drives 4 -drivemin 50m
allsam1.1 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
allsam1.1 -recycle_ignore
allsam1.2 -drives 4 -drivemin 50m
allsam1.2 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
allsam1.2 -recycle_ignore
testset.1 -drives 4 -drivemin 50m
testset.1 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
testset.1 -recycle_ignore
testset.2 -drives 4 -drivemin 50m
testset.2 -recycle_hwm 60 -recycle_mingain 90 -recycle_vsncount 1
testset.2 -recycle_ignore
endparams
vsns
samfs1.1 at 000000
allsam1.1 at 00000[1-5] # vsns 1 through 5.
```

```
allsam1.2 at 00000[6-9] # vsns 6 through 9.  
testset.1 at 00001[0,4] # vsns 10 and 14.  
testset.2 at 00001[5,9] # vsns 15 and 19.  
endvsns
```

## 5 Edit the `recycler.cmd` file on the server to specify recycler log file to receive output from the recycler and `no_recycle` directive for the Sun SAM-Remote client's VSNs.

The Sun SAM-Remote client is configured to write its copy 2 archive copies to cartridges in the Sun SAM-Remote server's library. The `no_recycle` directive is necessary to prevent the VSNs being used by the Sun SAM-Remote client for archiving from being recycled by the Sun SAM-Remote server.

The following example shows the `recycler.cmd` file on server `sky`, which has been edited to specify a recycler log file.

```
#  
# This is the /etc/opt/SUNWsamfs/recycler.cmd file  
# on Sun SAM-Remote server sky.  
#  
logfile = /var/opt/SUNWsamfs/recycler/recycler.log  
adicl -ignore  
no_recycle at 00002[0-9] # Prevents VSNs assigned to zeke from  
# being recycled.
```

## 6 Use the `sam-recycler` command to test the recycler on the Sun SAM-Remote client.

Run the recycler on the Sun SAM-Remote client system to see whether the recycler properly acknowledges the devices and VSNs specified in the configuration files.

For example, you can use the following command to perform the initial test of the recycler:

```
zeke# sam-recycler -dvx
```

This testing is important because if the recycler detects that the system on which it is running has no archive images on a particular VSN listed in any of that system's catalogs (including the historian catalog), the `recycler.sh` script can call for the cartridge to be labeled. The Sun SAM-Remote client and the SAM-QFS servers communicate between them to inform each side of the presence of archive copies. All such information is provided locally from local SAM-QFS file systems.



**Caution** – Labeling a cartridge destroys all data on the cartridge.

---

The recycler runs and logs its activity to the recycler log file. The recycler log file is defined in the `recycler.cmd` file. For more information about the `sam-recycler` command, see “[sam-recycler\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## 7 Examine the recycler log file to find the message.

Recycling is ignored on this archive set.

See [Example 18–6](#).

- 8 Issue the `sam-recycler` command from the Sun SAM-Remote server to verify that the recycler is not recycling any VSNs reserved for the Sun SAM-Remote client.**

For example:

```
sky# sam-recycler -dvx
```

The preceding command runs the recycler and writes its activity to the recycler log file. For more information about the `sam-recycler` command, see “[sam-recycler\(1M\)](#)” in *Sun QFS and Sun Storage Archive Manager Reference Manual*. Also, see [Example 18–7](#).

- 9 Analyze the server and client `recycler.log` files to choose VSNs that are candidates for recycling.**

Near the end of the `recycler.log` file is a Status column.

- In the client log files, VSNs with the following types of status entries are candidates for recycling:
  - `no-data VSN`. To recycle a no-data VSN, see “[How to Recycle no-data VSNs](#)” on [page 240](#).
  - `partially full`. To recycle a partially full VSN, see [To Recycle partially full VSNs](#).
- In the server log file, the best candidates for recycling are those with a 0 value in the Count, Bytes, and Use columns.

## Examples for How to Configure Recycling (Method 1)

### EXAMPLE 18–6 Sample Log File

The following example shows a sample log file.

```
# recycler.log from client zeke.
===== Recycler begins at Mon Jun 4 09:49:41 2001 =====
Initial 7 catalogs:
0 Family: stk_l20 Path: /var/opt/SUNWsamfs/catalog/L20_cat
Vendor: STK Product: L20
SLOT ty capacity space vsn
0 lt 33.0G 33.0G 000173
1 lt 32.8G 44.1M CEL170
2 lt 33.0G 33.0G CEL139
4 lt 32.8G 16.8G CFC504
5 lt 33.0G 33.0G CFC503
6 lt 32.9G 0 CSM689
7 lt 32.9G 19.6G CSM690
8 lt 33.0G 33.0G CSM691
9 lt 33.0G 33.0G CSM692
10 lt 10.0G 10.0G CLN018
11 lt 33.0G 33.0G 000766
Total Capacity: 339.2G bytes, Total Space Available: 244.3G bytes
Volume utilization 27%, high 95% VSN_min 50%
Recycling is ignored on this robot.
```

## EXAMPLE 18-6 Sample Log File (Continued)

```
1 Family: skyr Path: /var/opt/SUNWsamfs/catalog/sky_cat
Vendor: (NULL) Product: (NULL)
SLOT ty capacity space vsn
0 at 48.5G 23.3G 000020
1 at 23.8G 23.8G 000021
2 at 48.5G 48.5G 000022
3 at 48.5G 48.5G 000023
4 at 48.5G 48.5G 000024
5 at 48.5G 2.6G 000025
6 at 48.5G 361.4k 000026
7 at 48.5G 48.5G 000027
8 at 48.5G 48.5G 000028
9 at 48.5G 0 000029
Total Capacity: 460.8G bytes, Total Space Available: 292.5G bytes
Volume utilization 36%, high 95% VSN_min 50%
Recycling is ignored on this robot.
```

```
2 Family: hy Path: /var/opt/SUNWsamfs/catalog/historian
Vendor: Sun SAM-FS Product: Historian
SLOT ty capacity space vsn
(no VSNs in this media changer)
Total Capacity: 0 bytes, Total Space Available: 0 bytes
Volume utilization 0%, high 95% VSN_min 50%
Recycling is ignored on this robot.
```

```
3 Family: defaultset.1 Path: /etc/opt/SUNWsamfs/archiver.cmd
Vendor: Sun SAM-FS Product: Archive set
SLOT ty capacity space vsn
0 lt 33.0G 33.0G 000766
1 lt 33.0G 33.0G 000173
2 lt 32.9G 0 CSM689
3 lt 32.9G 19.6G CSM690
4 lt 33.0G 33.0G CSM691
5 lt 33.0G 33.0G CSM692
Total Capacity: 197.6G bytes, Total Space Available: 151.5G bytes
Volume utilization 23%, high 60% VSN_min 90%
Recycling is ignored on this archive set.
```

```
4 Family: defaultset.2 Path: /etc/opt/SUNWsamfs/archiver.cmd
Vendor: Sun SAM-FS Product: Archive set
SLOT ty capacity space vsn
0 lt 32.9G 0 CSM689
1 at 48.5G 23.3G 000020
2 at 23.8G 23.8G 000021
3 at 48.5G 2.6G 000025
4 at 48.5G 361.4k 000026
5 at 48.5G 48.5G 000027
6 at 48.5G 48.5G 000028
7 at 48.5G 0 000029
Total Capacity: 348.0G bytes, Total Space Available: 146.8G bytes
Volume utilization 57%, high 60% VSN_min 90%
Recycling is ignored on this archive set.
```

```
5 Family: archiveset.1 Path: /etc/opt/SUNWsamfs/archiver.cmd
Vendor: Sun SAM-FS Product: Archive set
```

## EXAMPLE 18-6 Sample Log File (Continued)

```

SLOT ty capacity space vsn
0 lt 32.8G 44.1M CEL170
1 lt 32.8G 16.8G CFC504
2 lt 33.0G 33.0G CFC503
Total Capacity: 98.6G bytes, Total Space Available: 49.8G bytes
Volume utilization 49%, high 60% VSN_min 90%
Recycling is ignored on this archive set.

6 Family: archiveset.2 Path: /etc/opt/SUNWsamfs/archiver.cmd
Vendor: Sun SAM-FS Product: Archive set
SLOT ty capacity space vsn
0 at 48.5G 23.3G 000020
1 at 23.8G 23.8G 000021
2 at 48.5G 48.5G 000022
3 at 48.5G 48.5G 000023
4 at 48.5G 48.5G 000024
Total Capacity: 218.0G bytes, Total Space Available: 192.8G bytes
Volume utilization 11%, high 60% VSN_min 90%
Recycling is ignored on this archive set.

21 VSNs:

---Archives--- -----Percent----- defaultset.1
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
in multiple sets 0 0 0 100 0 stk_l20:lt:CSM689
partially full 111 2.8G 8 31 61 stk_l20:lt:CSM690
empty VSN 0 0 0 0 100 stk_l20:lt:000173
empty VSN 0 0 0 0 100 stk_l20:lt:CSM691
empty VSN 0 0 0 0 100 stk_l20:lt:CSM692
empty VSN 0 0 0 0 100 stk_l20:lt:000766

---Archives--- -----Percent----- defaultset.2
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 100 0 skyrs:at:000029
no-data VSN 0 0 0 99 1 skyrs:at:000026
partially full 111 2.8G 6 88 6 skyrs:at:000025
empty VSN 0 0 0 0 100 skyrs:at:000028
empty VSN 0 0 0 0 100 skyrs:at:000027

---Archives--- -----Percent----- archiveset.1
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 99 1 stk_l20:lt:CEL170
partially full 677 2.3G 8 40 52 stk_l20:lt:CFC504
empty VSN 0 0 0 0 100 stk_l20:lt:CFC503

---Archives--- -----Percent----- archiveset.2
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
in multiple sets 0 0 0 51 49 skyrs:at:000020
empty VSN 0 0 0 0 100 skyrs:at:000022
empty VSN 0 0 0 0 100 skyrs:at:000023
empty VSN 0 0 0 0 100 skyrs:at:000024
in multiple sets 0 0 0 0 100 skyrs:at:000021

---Archives--- -----Percent----- stk_l20
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
empty VSN 0 0 0 0 100 stk_l20:lt:CLN018

```

**EXAMPLE 18-6** Sample Log File (Continued)

```
partially full 13 80.3k 0 0 100 stk_l20:lt:CEL139
Recycler finished.
===== Recycler ends at Mon Jun 4 09:49:53 2001 =====
```

**EXAMPLE 18-7** Sample Recycler Log File

The following example shows a sample recycler log file.

```
# recycler.log file from server sky.
===== Recycler begins at Mon Jun 4 09:50:44 2001 =====
Initial 6 catalogs:
0 Family: adic1 Path: /var/opt/SUNWsamfs/catalog/adic1
Vendor: ADIC Product: Scalar 1000
SLOT ty capacity space vsn
0 at 1.3G 1.2G 000001
1 at 1.3G 1.3G 000002
2 at 1.3G 1.3G 000004
3 at 48.5G 0 000010
4 at 48.5G 0 000011
5 at 48.5G 43.5G 000018
6 at 48.5G 0 000019
7 at 48.5G 23.3G 000020
8 at 23.8G 23.8G 000021
9 at 48.5G 48.5G 000022
10 at 48.5G 48.5G 000023
11 at 48.5G 48.5G 000024
12 at 48.5G 2.6G 000025
13 at 48.5G 361.4k 000026
14 at 48.5G 48.5G 000027
15 at 48.5G 48.5G 000028
16 at 48.5G 0 000029
17 at 1.3G 1.3G 000005
18 at 48.5G 48.5G 000016
19 at 23.8G 23.8G CLN001
20 at 23.8G 23.8G CLN002
21 at 23.8G 23.8G CLN004
22 at 23.8G 23.8G CLN003
23 at 48.5G 421.6M 000015
24 at 1.3G 1.3G 000000
25 at 48.5G 0 000013
26 at 1.3G 1.3G 000003
27 at 48.5G 43.6G 000007
28 at 48.5G 41.8G 000008
29 at 48.5G 46.9G 000006
30 at 48.5G 48.3G 000009
31 at 48.5G 0 000014
32 at 48.5G 0 000012
33 at 48.5G 40.1G 000017
Total Capacity: 1.2T bytes, Total Space Available: 708.7G bytes
Volume utilization 43%, high 95% VSN_min 50%
Recycling is ignored on this robot.

1 Family: hy Path: /var/opt/SUNWsamfs/catalog/historian
Vendor: Sun SAM-FS Product: Historian
SLOT ty capacity space vsn
```

## EXAMPLE 18-7 Sample Recycler Log File (Continued)

```
(no VSNs in this media changer)
Total Capacity: 0 bytes, Total Space Available: 0 bytes
Volume utilization 0%, high 95% VSN_min 50%
Recycling is ignored on this robot.

2 Family: testset.1 Path: /etc/opt/SUNWsamfs/archiver.cmd
Vendor: Sun SAM-FS Product: Archive set
SLOT ty capacity space vsn
0 at 48.5G 0 000010
1 at 48.5G 0 000014
Total Capacity: 97.1G bytes, Total Space Available: 0 bytes
Volume utilization 100%, high 60% VSN_min 90%: *** Needs recycling ***
Recycling is ignored on this archive set.

3 Family: testset.2 Path: /etc/opt/SUNWsamfs/archiver.cmd
Vendor: Sun SAM-FS Product: Archive set
SLOT ty capacity space vsn
0 at 48.5G 0 000019
1 at 48.5G 421.6M 000015
Total Capacity: 97.1G bytes, Total Space Available: 421.6M bytes
Volume utilization 99%, high 60% VSN_min 90%: *** Needs recycling ***
Recycling is ignored on this archive set.

4 Family: allsam1.1 Path: /etc/opt/SUNWsamfs/archiver.cmd
Vendor: Sun SAM-FS Product: Archive set
SLOT ty capacity space vsn
0 at 1.3G 1.2G 000001
1 at 1.3G 1.3G 000002
2 at 1.3G 1.3G 000004
3 at 1.3G 1.3G 000005
4 at 1.3G 1.3G 000003
Total Capacity: 6.5G bytes, Total Space Available: 6.3G bytes
Volume utilization 3%, high 60% VSN_min 90%
Recycling is ignored on this archive set.

5 Family: allsam1.2 Path: /etc/opt/SUNWsamfs/archiver.cmd
Vendor: Sun SAM-FS Product: Archive set
SLOT ty capacity space vsn
0 at 48.5G 43.6G 000007
1 at 48.5G 41.8G 000008
2 at 48.5G 46.9G 000006
3 at 48.5G 48.3G 000009
Total Capacity: 194.2G bytes, Total Space Available: 180.6G bytes
Volume utilization 6%, high 60% VSN_min 90%
Recycling is ignored on this archive set.

Need to select candidate for media changer testset.1 to free up 39.8G bytes.
Quantity of data to move limited to (no limit) bytes and 1 VSNs.
Checking 000010. Need to free 39.8G, quantity limit: (no limit), VSN count: 1.
VSN is in correct media changer... good.
VSN is not already recycling... good.
VSN has no request files... good.
VSN has no 'archive -n' files...good.
VSN was not specified as "no_recycle" in recycler.cmd file... good.
VSN does not exceed VSN count limit... good.
VSN does not exceed data quantity limit... good.
```

## EXAMPLE 18-7 Sample Recycler Log File (Continued)

```
VSN meets minimum gain requirement.
Recycling is ignored on this media changer - VSN not marked for recycling.

Checking 000014. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN is in correct media changer... good.
VSN is not already recycling... good.
VSN has no request files... good.
VSN has no 'archive -n' files...good.
VSN was not specified as "no_recycle" in recycler.cmd file... good.
VSN exceeds VSN count limit - skipped.
Checking 000019. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000015. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking 000001. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000003. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000004. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000005. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000002. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000008. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000007. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking 000006. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000009. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000011. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000029. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000013. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000012. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking 000026. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000025. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000020. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000017. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000018. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN003. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
```

## EXAMPLE 18-7 Sample Recycler Log File (Continued)

Checking 000021. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking 000022. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking 000027. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking 000028. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking 000023. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking 000024. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking 000016. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.

Checking CLN001. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking CLN002. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking CLN004. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking 000000. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 No candidate was found in this media changer.

Need to select candidate for media changer testset.2 to free up 38.8G bytes.  
 Quantity of data to move limited to (no limit) bytes and 1 VSNs.

Checking 000010. Need to free 38.8G, quantity limit: (no limit), VSN count: 1.  
 VSN not in correct media changer.  
 Checking 000014. Need to free 38.8G, quantity limit: (no limit), VSN count: 1.  
 VSN not in correct media changer.

Checking 000019. Need to free 38.8G, quantity limit: (no limit), VSN count: 1.  
 VSN is in correct media changer... good.  
 VSN is not already recycling... good.  
 VSN has no request files... good.  
 VSN has no 'archive -n' files...good.  
 VSN was not specified as "no\_recycle" in recycler.cmd file... good.  
 VSN does not exceed VSN count limit... good.  
 VSN does not exceed data quantity limit... good.  
 VSN meets minimum gain requirement.  
 Recycling is ignored on this media changer - VSN not marked for recycling.

Checking 000015. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN is in correct media changer... good.  
 VSN is not already recycling... good.  
 VSN has no request files... good.  
 VSN has no 'archive -n' files...good.  
 VSN was not specified as "no\_recycle" in recycler.cmd file... good.  
 VSN exceeds VSN count limit - skipped.

Checking 000001. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking 000003. Need to free 0E, quantity limit: (no limit), VSN count: 0.  
 VSN not in correct media changer.  
 Checking 000004. Need to free 0E, quantity limit: (no limit), VSN count: 0.

## EXAMPLE 18-7 Sample Recycler Log File (Continued)

```
VSN not in correct media changer.
Checking 000005. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000002. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking 000008. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000007. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000006. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000009. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000011. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000029. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking 000013. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000012. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000026. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000025. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000020. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000017. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking 000018. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN003. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000021. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000022. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000027. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000028. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000023. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.

Checking 000024. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking 000016. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN001. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN002. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
Checking CLN004. Need to free 0E, quantity limit: (no limit), VSN count: 0.
```

## EXAMPLE 18-7 Sample Recycler Log File (Continued)

```

VSN not in correct media changer.
Checking 000000. Need to free 0E, quantity limit: (no limit), VSN count: 0.
VSN not in correct media changer.
No candidate was found in this media changer.
34 VSNs:

---Archives--- -----Percent----- testset.1
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 100 0 adic1:at:000010
no-data VSN 0 0 0 100 0 adic1:at:000014

---Archives--- -----Percent----- testset.2
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 100 0 adic1:at:000019
partially full 677 2.3G 5 93 2 adic1:at:000015

---Archives--- -----Percent----- allsam1.1
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
partially full 97 173.8M 1 9 90 adic1:at:000001
no-data VSN 0 0 0 2 98 adic1:at:000003
no-data VSN 0 0 0 2 98 adic1:at:000004
empty VSN 0 0 0 100 adic1:at:000005
empty VSN 0 0 0 100 adic1:at:000002

---Archives--- -----Percent----- allsam1.2
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 13 87 adic1:at:000008
partially full 98 1.6G 3 7 90 adic1:at:000007
no-data VSN 0 0 0 3 97 adic1:at:000006
empty VSN 0 0 0 100 adic1:at:000009

---Archives--- -----Percent----- adic1
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 100 0 adic1:at:000011
no_recycle VSN 0 0 0 100 0 adic1:at:000029
no-data VSN 0 0 0 100 0 adic1:at:000013
no-data VSN 0 0 0 100 0 adic1:at:000012
no_recycle VSN 0 0 0 99 1 adic1:at:000026
no_recycle VSN 0 0 0 94 6 adic1:at:000025
no_recycle VSN 0 0 0 51 49 adic1:at:000020
no-data VSN 0 0 0 17 83 adic1:at:000017
no-data VSN 0 0 0 10 90 adic1:at:000018
empty VSN 0 0 0 100 adic1:at:CLN003

no_recycle VSN 0 0 0 100 adic1:at:000021
no_recycle VSN 0 0 0 100 adic1:at:000022
no_recycle VSN 0 0 0 100 adic1:at:000027
no_recycle VSN 0 0 0 100 adic1:at:000028
no_recycle VSN 0 0 0 100 adic1:at:000023
no_recycle VSN 0 0 0 100 adic1:at:000024
empty VSN 0 0 0 100 adic1:at:000016
empty VSN 0 0 0 100 adic1:at:CLN001
empty VSN 0 0 0 100 adic1:at:CLN002
empty VSN 0 0 0 100 adic1:at:CLN004
partially full 12 88.3k 0 0 100 adic1:at:000000

```

**EXAMPLE 18-7** Sample Recycler Log File (Continued)

```

Recycler finished.
===== Recycler ends at Mon Jun 4 09:51:05 2001 =====

```

## ▼ How to Recycle no-data VSNs

The no-data VSNs are the easiest VSNs to recycle. For these VSNs, the Count, Bytes, and Use field values are all 0 (zero).

### 1 Examine the `recycle` . log file from the client for any no-data VSNs.

In this example, VSNs 000029 and 000026 from the client `zeke` can be considered for recycling because they are no-data VSNs, as shown in the code below.

```

# From the client zeke recycle.log file:
---Archives--- -----Percent----- defaultset.2
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 100 0 skyrs:at:000029
no-data VSN 0 0 0 99 1 skyrs:at:000026
partially full 111 2.8G 6 88 6 skyrs:at:000025
empty VSN 0 0 0 100 skyrs:at:000028
empty VSN 0 0 0 100 skyrs:at:000027

```

### 2 If no-data VSNs are found, examine the `recycle` . log file from the server and determine whether they are represented identically in the server's recycle log file.

Verify that there is no active data from the server archived on those VSNs.

The code below shows the data for the `no_recycle` VSNs in the server's `recycle` . log file. For VSNs 000029 and 000026, the data in the server's `recycle` . log file is identical to that in the client's `recycle` . log file.

```

# From the Server log file:
---Archives--- -----Percent----- adic1
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 100 0 adic1:at:000011
no_recycle VSN 0 0 0 100 0 adic1:at:000029
no-data VSN 0 0 0 100 0 adic1:at:000013
no-data VSN 0 0 0 100 0 adic1:at:000012
no_recycle VSN 0 0 0 99 1 adic1:at:000026
no_recycle VSN 0 0 0 94 6 adic1:at:000025
no_recycle VSN 0 0 0 51 49 adic1:at:000020
no-data VSN 0 0 0 17 83 adic1:at:000017
no-data VSN 0 0 0 10 90 adic1:at:000018
empty VSN 0 0 0 100 adic1:at:CLN003
.
.
.

```

### 3 If no active data from the server is archived on the selected VSNs, use the `tpLabel` or `odLabel` command to relabel the VSNs.




---

**Caution** – This destroys all data on the VSN and reclaims space.

---

For example, for tape VSN 000029, use the following command:

```
sky# tpLabel -vsn 000029 -old 000029 at.000029
```

When this VSN 000029 is relabeled, you regain 100 percent of the space on that VSN.

If the media is a magneto-optical disc, use the `odLabel` command. For more information, see “`odlabel(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## ▼ How to Recycle partially full VSNs

The VSNs for which a partially full status is reported can also be recycled.

### 1 Examine the `recycler.log` file from the client for any partially full VSNs.

In the example, you can consider VSN 000025 from the client, `zeke`, for recycling because its status is partially full, as shown below.

```
# From the client zeke recycler.log file:
---Archives--- -----Percent----- defaultset.2
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 100 0 skyrs:at:000029
no-data VSN 0 0 0 99 1 skyrs:at:000026
partially full 111 2.8G 6 88 6 skyrs:at:000025
empty VSN 0 0 0 0 100 skyrs:at:000028
empty VSN 0 0 0 0 100 skyrs:at:000027
```

VSN 000025 shows that 6 percent of its space is in use. These active archive images must be rearchived before this VSN can be recycled. The following steps in this process show how to ensure that these active archive images are rearchived to another VSN.

### 2 Examine the `recycler.log` file from the server side to ensure that no active data from the server is archived on that VSN.

The server's `recycler.log` file indicates that VSN 000025 is 6 percent free, which is the same percentage that was reported in the client's `recycler.log` file. The server is not aware of the client's archive images, so it reports that all of the remaining 94 percent is consumed by obsolete archive images.

```
# From the Server log file:
---Archives--- -----Percent----- adic1
-----Status----- Count Bytes Use Obsolete Free Library:Type:VSN
no-data VSN 0 0 0 100 0 adic1:at:000011
no_recycle VSN 0 0 0 100 0 adic1:at:000029
no-data VSN 0 0 0 100 0 adic1:at:000013
no-data VSN 0 0 0 100 0 adic1:at:000012
no_recycle VSN 0 0 0 99 1 adic1:at:000026
no_recycle VSN 0 0 0 94 6 adic1:at:000025
```

```
no_recycle VSN 0 0 0 51 49 adic1:at:000020
no-data VSN 0 0 0 17 83 adic1:at:000017
.
.
.
```

**3 On the server, rearchive the active files on the VSN.**

```
sky# chmed +c at.000025
```

For more information about the `chmed` command, see “`chmed(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

**4 On the client, run the recycler again.**

```
zeke# sam-recycler -dvx
```

This command marks each active file to be rearchived to another VSN.

**5 Start the archiver.**

You can either let the archiver run normally or type `:arrun` from the `samu(1M)` utility on the client. For more information about the `:arrun` command, see “`samu(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

**6 When archiving is complete, issue the `sam-recycler` command to run the recycler on the client again.**

```
zeke# sam-recycler -dvx
```

This ensures that all active files have been rearchived.

**7 If the Count, Bytes, and Use field values are all 0 (zero), use the `tplabel` or `odlabel` command to relabel the VSN from the server.**

For example, for tape VSN 000025, use the following command:

```
sky# tplabel -vsn 000025 -old 000025 at.000025
```

This command relabels the VSN and destroys all data on it. After this VSN is relabeled, you regain 88 percent of the space on this VSN.

If the media had been a magneto-optical disc, you would have used the `odlabel` command. For more information about the `odlabel` command, see “`odlabel(1M)`” in *Sun QFS and Sun Storage Archive Manager Reference Manual*.

## Recycling in a Sun SAM-Remote Environment (Method 2)

This section presents another way you can recycle volumes using Sun SAM-remote software.



---

**Caution** – Use the recycler in a Sun SAM-Remote environment only after following the steps in this procedure completely and only after testing your configuration to verify that recycling is taking place correctly.

---

## ▼ How To Configure Recycling (Method 2)

- 1 **On the Sun SAM-Remote client, issue the `sam-recycler` command to determine which volumes are the best candidates for recycling. For example:**

```
client# sam-recycler -dvx
```

- 2 **Analyze the recycler log file for recycling candidates.**

Toward the end of the `recycler.log` file is a Status column. VSNs with the following types of status entries in the client log file are candidates for recycling:

- **no-data VSN.** To recycle a no-data VSN, see [To Recycle no-data VSNs](#).
- **partially full.** To recycle a partially full VSN, see [To Recycle partially full VSNs](#).  
In the server log file, the best candidates for recycling are those with a 0 value in the Count, Bytes, and Use columns.

- 3 **On the Sun SAM-Remote server, issue the `chmed` command to set the recycle flag on the selected VSNs.**

For example:

```
server# chmed +c at.00025
```

- 4 **Wait until the VSNs being recycled are drained completely of archive images.**

The archiver on the client side does this.

- 5 **On the Sun SAM-Remote server, issue the `tpLabel` or `odLabel` command, depending on the archive media, to relabel the volumes.**

- 6 **On the Sun SAM-Remote server, clear any flags such as `R` or `c` that prevent the volumes from being used for archiving on the Sun SAM-Remote client.**

