



Sun StorEdge™ SAM-FS File System Configuration and Administration Guide

Version 4, Update 4

Sun Microsystems, Inc.
www.sun.com

Part No. 819-3635-10
Sept. 2005, Revision A

Submit comments about this document at: <http://www.sun.com/hwdocs/feedback>

Copyright 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and in other countries.

This document and the product to which it pertains are distributed under licenses restricting their use, copying, distribution, and decompilation. No part of the product or of this document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and in other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, AnswerBook2, docs.sun.com, Solaris, Java, Solstice DiskSuite, SunPlex, and Sun StorEdge are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and in other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and in other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

Mozilla is a trademark or registered trademark of Netscape Communications Corporation in the United States and other countries.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

U.S. Government Rights—Commercial use. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, Californie 95054, Etats-Unis. Tous droits réservés.

Sun Microsystems, Inc. a les droits de propriété intellectuelle relatants à la technologie qui est décrit dans ce document. En particulier, et sans la limitation, ces droits de propriété intellectuelle peuvent inclure un ou plus des brevets américains énumérés à <http://www.sun.com/patents> et un ou les brevets plus supplémentaires ou les applications de brevet en attente dans les Etats-Unis et dans les autres pays.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a.

Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, AnswerBook2, docs.sun.com, Solaris, Java, Solstice DiskSuite, SunPlex, et Sun StorEdge sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

Mozilla est une marque de Netscape Communications Corporation aux Etats-Unis et à d'autres pays.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciées de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

LA DOCUMENTATION EST FOURNIE "EN L'ÉTAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFAÇON.



Adobe PostScript

Contents

Preface xix

How This Book Is Organized xix

Using UNIX Commands xx

Shell Prompts xx

Typographic Conventions xxi

Related Documentation xxii

Accessing Sun Documentation Online xxii

▼ To Access Documentation From `docs.sun.com` xxii

▼ To Access Documentation From Sun's Network Storage Documentation
Web Site xxiii

Third-Party Web Sites xxiii

Contacting Sun Technical Support xxiii

Licensing xxiii

Installation Assistance xxiv

Sun Welcomes Your Comments xxiv

1. File System Overview 1

File System Features 1

Volume Management 2

Support for Paged and Direct I/O 2

High Capacity	3
Fast File System Recovery	3
vnode Interface	3
Sun StorEdge SAM-FS Archive Management	4
Additional File System Features	4
Design Basics	5
Inode Files and File Characteristics	5
Specifying Disk Allocation Units	6
DAU Settings and File System Geometry	6
Sun StorEdge SAM-FS File Systems	7
Dual Allocation Scheme	7
Sun StorEdge QFS Data Alignment	7
Stripe Widths on Data Disks	8
File Allocation Methods	9
Metadata Allocation	9
Round-Robin Allocation	9
Striped Allocation	10
2. System Configuration Tasks	13
Using the File System Manager Software	13
▼ To Invoke File System Manager for the First Time	14
Creating Additional Administrator and User Accounts	15
▼ To Create Additional Administrator Accounts	15
▼ To Create Additional Guest Accounts	16
▼ To Add an Additional Server for File System Manager Access	16
Using the File System Manager Portal Agent	17
▼ To Enable the File System Manager Portal Agent	17
About Port Numbers for the File System Manager Portal Agent	18

About Configuration and Log Files for the File System Manager Portal Agent	18
Function of the <code>mcf</code> File	19
The Equipment Identifier Field	20
The Equipment Ordinal Field	20
The Equipment Type Field	20
The Family Set Field	21
The Device State Field	21
The Additional Parameters Field	22
Interactions Among File Settings, Options, and Directives	22
Initializing a File System	22
3. Performing Operational Tasks	25
Viewing Files and File Attributes	25
File Attributes and File States	25
System-Specified File States	27
Displaying File Information	28
Retention Line Explanation	31
Checksum Line Explanation	31
Propagating Configuration File Changes to the System	31
▼ To Change <code>mcf(4)</code> or <code>defaults.conf(4)</code> File System Information in a Sun StorEdge SAM-FS Environment	32
▼ To Change <code>mcf(4)</code> or <code>defaults.conf(4)</code> Removable Media Drive Information	33
▼ To Change <code>archiver.cmd(4)</code> or <code>stager.cmd(4)</code> Information	34
Setting Up Mount Parameters	35
The <code>mount(1M)</code> Command	36
The <code>/etc/vfstab</code> File	36
The <code>samfs.cmd</code> File	37
Unmounting a File System	38

- ▼ To Unmount a Sun StorEdge SAM-FS File System 39

Adding Disk Cache to a File System 39

- ▼ To Add Disk Cache to a File System 40

Re-creating a File System 41

- ▼ To Back Up and Re-create a File System 41

4. Administering File System Quotas 45

Overview 45

Types of Quotas, Quota Files, and Quota Records 46

Soft Limits and Hard Limits 47

Disk Blocks and File Limits 47

Enabling Quotas 48

Guidelines for Setting Up Quotas 48

- ▼ To Configure a New File System to Use Quotas 48

- ▼ To Configure an Existing File System to Use Quotas 50

- ▼ To Assign Admin Set IDs to Directories and Files 52

Setting Infinite Quotas 53

- ▼ To Set an Infinite Quota 53

Enabling Default Quota Values 54

- ▼ To Enable Default Quota Values for Users, Groups, or Admin Sets 54

Enabling Limits 55

- ▼ To Enable Limits for Users, Groups, or Admin Sets 55

- ▼ To Enable or Change Limits for Users, Groups, or Admin Sets Using an Existing Quota File 55

Checking Quotas 57

- ▼ To Check for Exceeded Quotas 57

Changing and Removing Quotas 59

- ▼ To Change the Grace Period 59

Changing the Grace Period Expiration 61

▼ To Inhibit Additional File System Resource Allocations	63
▼ To Remove a File System's Quotas	65
▼ To Correct Quotas	66
5. Advanced Topics	69
Using Daemons, Processes, and Tracing	69
Daemons and Processes	69
Trace Files	71
Trace File Content	71
Trace File Rotation	72
Determining Which Processes Are Being Traced	72
Using the <code>setfa(1)</code> Command to Set File Attributes	73
Selecting File Attributes for Files and Directories	73
Preallocating File Space	74
Selecting a File Allocation Method and Stripe Width	75
Configuring WORM-FS File Systems	75
Enabling the WORM-FS Feature	76
Setting a File's Retention Period Using <code>touch</code>	79
Using <code>sls</code> to View WORM-FS Files	81
Using <code>touch</code> to Extend a File's Retention Period	82
Using <code>sfind</code> to Find WORM-FS Files	82
Accommodating Large Files	83
Understanding I/O Types	84
Paged I/O	84
Direct I/O	84
I/O Switching	85
Increasing File Transfer Performance for Large Files	85
▼ To Increase File Transfer Performance	85
Setting the Write Throttle	88

Setting the Flush-Behind Rate	89
Tuning the Number of Inodes and the Inode Hash Table	90
The <code>ninodes</code> Parameter	90
The <code>nhino</code> Parameter	91
When to Set the <code>ninodes</code> and <code>nhino</code> Parameters	91
A. Troubleshooting Sun StorEdge QFS	93
Checking File System Integrity and Repairing File Systems	93
▼ To Check a File System	94
▼ To Repair a File System	95
B. Using the <code>samu(1M)</code> Operator Utility	97
Overview	97
▼ To Invoke <code>samu(1M)</code>	98
▼ To Display a <code>samu(1M)</code> Screen	98
▼ To Stop <code>samu(1M)</code>	99
Interacting With <code>samu(1M)</code>	99
Entering a Device	99
Getting Online Help	100
▼ To Access Online Help From a Display Screen	100
Operator Displays	100
(a) - Archiver Status Display	101
Navigation	102
Sample Display	102
Field Descriptions	103
(c) - Device Configuration Display	104
Navigation	104
Sample Display	104
Field Descriptions	105

(C) - Memory Display	105
Sample Display	106
(d) - Daemon Trace Controls Display	106
Sample Display	107
(D) - Disk Volume Dictionary	108
Sample Display	108
Flags	109
(f) - File Systems Display	109
Sample Display	109
Field Descriptions	110
(F) - Optical Disk Label Display	111
(h) - Help Display	111
Navigation	112
Sample Display	112
(I) - Inode Display	113
Navigation	113
Sample Display	114
(J) - Preview Shared Memory Display	114
Navigation	115
Sample Display	115
(K) - Kernel Statistics Display	115
Navigation	116
Sample Display	116
(l) - Usage Display	116
Sample Display	117
(L) - Shared Memory Tables	117
Sample Display	118
(m) - Mass Storage Status Display	118

Sample Display	119
Field Descriptions	120
(M) - Shared Memory Display	120
Navigation	121
Sample Display	121
(n) - Staging Status Display	122
Sample Display	122
(N) - File System Parameters Display	123
Navigation	123
Sample Display	123
(o) - Optical Disk Status Display	124
Navigation	124
Sample Display	125
Field Descriptions	125
(p) - Removable Media Load Requests Display	126
Navigation	127
Sample Display	127
Field Descriptions	127
Flags	128
(P) - Active Services Display	128
Navigation	129
Sample Display	129
(r) - Removable Media Status Display	129
Sample Display	130
Field Descriptions	130
(R) - Sun SAM-Remote Information Display	131
(s) - Device Status Display	131
Navigation	132

Sample Display	132
Field Descriptions	133
(S) - Sector Data Display	133
Navigation	133
(t) - Tape Drive Status Display	134
Navigation	134
Sample Display	134
Field Descriptions	135
(T) - SCSI Sense Data Display	135
Navigation	136
(u) - Staging Queue Display	136
Navigation	136
Sample Display	137
Field Descriptions	137
(U) - Device Table Display	138
Navigation	138
Sample Display	139
(v) - Automated Library Catalog Display	139
Navigation	140
Sample Display	141
Field Descriptions	142
Flags	142
(w) - Pending Stage Queue	143
Navigation	144
Sample Display	144
Field Descriptions	145
Operator Display Status Codes	145
Removable Media Device Display Status Codes	145

File System Display Status Codes	146
Operator Display Device States	147
▼ To Change a Drive State from down to on	148
▼ To Change a Drive State from on to down	149
Operator Commands	150
Device Commands	150
Sun StorEdge SAM-FS Commands: Archiver Control	151
The <code>:hwm_archive eq</code> and <code>:nohwm_archive eq</code> Commands	153
The <code>:thresh eq high low</code> Command	153
Sun StorEdge SAM-FS Commands: Releaser Control	154
The <code>:maxpartial eq value</code> Command	154
The <code>:partial eq value</code> Command	154
Sun StorEdge SAM-FS Commands: Stager Control	155
The <code>:partial_stage eq value</code> Command	155
The <code>:stage_flush_behind eq value</code> Command	155
The <code>:stage_n_window eq value</code> Command	155
The <code>:stage_retries eq value</code> Command	156
The <code>:stclear mt.vsn</code> Command	156
The <code>:stidle</code> Command	156
The <code>:strun</code> Command	156
File System Commands: I/O Management	156
The <code>:flush_behind eq value</code> Command	156
The <code>:force_nfs_async eq</code> and <code>:noforce_nfs_async eq</code> Commands	157
The <code>:readahead eq contig</code> Command	157
The <code>:sw_raid eq</code> and <code>:nosw_raid eq</code> Commands	158
The <code>:writebehind eq contig</code> Command	158
The <code>:wr_throttle eq value</code> Command	158
File System Commands: Direct I/O Management	158

The <code>:dio_rd_form_min eq value</code> and <code>:dio_wr_form_min eq value</code> Commands	159
The <code>:dio_rd_ill_min eq value</code> and <code>:dio_wr_ill_min eq value</code> Commands	159
The <code>:dio_rd_consec eq value</code> and <code>:dio_wr_consec eq value</code> Commands	159
The <code>:dio_szero eq</code> and <code>:nodio_szero eq</code> Commands	160
The <code>:forcedirectio eq</code> and <code>:noforcedirectio eq</code> Commands	160
File System Commands: Miscellaneous	160
The <code>:abr eq</code> and <code>:noabr eq</code> Commands	160
The <code>:dmr eq</code> and <code>:nodmr eq</code> Commands	161
The <code>:invalid eq interval</code> Command	161
The <code>:mm_stripe eq value</code> Command	161
The <code>:refresh_at_eof eq</code> and <code>:norefresh_at_eof eq</code> Commands	162
The <code>:suid eq</code> and <code>:nosuid eq</code> Commands	162
The <code>:stripe eq value</code> Command	162
The <code>:sync_meta eq value</code> Command	163
The <code>:trace eq</code> and <code>:notrace eq</code> Commands	163
Automated Library Commands	163
The <code>:audit [-e] eq [:slot [:side]]</code> Commands	163
The <code>:export eq:slot</code> and <code>:export mt.vsn</code> Commands	164
The <code>:import eq</code> Command	164
The <code>:load eq:slot [:side]</code> and <code>:load mt.vsn</code> Commands	164
The <code>:priority pid newpri</code> Command	165
Miscellaneous Commands	165
The <code>:clear vsn [index]</code> Command	165
The <code>:devlog eq [option]</code> Command	165
The <code>:diskvols volume [+flag -flag]</code> Command	166

The <code>:dtrace</code> Commands	166
The <code>:fs <i>fsname</i></code> Command	167
The <code>:mount <i>mntpt</i></code> Command	167
The <code>:open <i>eq</i></code> Command	167
The <code>:read <i>addr</i></code> Command	167
The <code>:refresh <i>i</i></code> Command	168
The <code>:snap [<i>filename</i>]</code> Command	168
The <code>:! <i>shell_command</i></code> Command	168

Glossary 169

Index 179

Tables

TABLE P-1	Shell Prompts	xx
TABLE P-2	Typographic Conventions	xxi
TABLE P-3	Related Sun StorEdge Documentation	xxii
TABLE 1-1	Product Overview	1
TABLE 1-2	Content of <code>.inode</code> Files	5
TABLE 1-3	<code>ms</code> File System Default Stripe Widths	8
TABLE 1-4	Metadata Allocation	9
TABLE 2-1	Sun StorEdge SAM-FS Equipment Type Field	21
TABLE 3-1	User-Specified File Attributes	26
TABLE 3-2	System-Specified File States	27
TABLE 3-3	<code>sls(1)</code> Output Explanation	28
TABLE 4-1	Quota File Names	46
TABLE 5-1	Daemons and Processes	70
TABLE 5-2	File Allocations and Stripe Widths	75
TABLE A-1	Error Indicators	93
TABLE 5-3	Control Keys for the <code>a</code> Display	102
TABLE 5-4	Control Keys for the <code>:a filesystem</code> Display	102
TABLE 5-5	<code>samu(1M)</code> <code>a</code> Display Field Descriptions	103
TABLE 5-6	Control Keys for the <code>c</code> Display	104
TABLE 5-7	<code>samu(1M)</code> <code>c</code> Display Field Descriptions	105

TABLE 5-8	Flags Field for the <code>samu(1M) D</code> Display	109
TABLE 5-9	<code>samu(1M) f</code> Display Field Descriptions	110
TABLE 5-10	Control Keys for the <code>h</code> Display	112
TABLE 5-11	Control Keys for the <code>I</code> Display	113
TABLE 5-12	Control Keys for the <code>J</code> Display	115
TABLE 5-13	Control Keys for the <code>K</code> Display	116
TABLE 5-14	<code>samu(1M) m</code> Display Field Descriptions	120
TABLE 5-15	Control Keys for the <code>M</code> Display	121
TABLE 5-16	Control Keys for the <code>N</code> Display	123
TABLE 5-17	Control Keys for the <code>o</code> Display	124
TABLE 5-18	<code>samu(1M) o</code> Display Field Descriptions	125
TABLE 5-19	Control Keys for the <code>p</code> Display	127
TABLE 5-20	<code>samu(1M) p</code> Display Field Descriptions	127
TABLE 5-21	Flags Field for the <code>samu(1M) p</code> Display	128
TABLE 5-22	Control Keys for the <code>P</code> Display	129
TABLE 5-23	<code>samu(1M) r</code> Display Field Descriptions	130
TABLE 5-24	Control Keys for the <code>s</code> Display	132
TABLE 5-25	<code>samu(1M) s</code> Display Field Descriptions	133
TABLE 5-26	Control Keys for the <code>S</code> Display	133
TABLE 5-27	Control Keys for the <code>t</code> Display	134
TABLE 5-28	<code>samu(1M) t</code> Display Field Descriptions	135
TABLE 5-29	Control Keys for the <code>T</code> Display	136
TABLE 5-30	Control Keys for the <code>u</code> Display	136
TABLE 5-31	<code>samu(1M) u</code> Display Field Descriptions	137
TABLE 5-32	Control Keys for the <code>U</code> Display	138
TABLE 5-33	Control Keys for the <code>v</code> Display	140
TABLE 5-34	<code>samu(1M) v</code> Display Field Descriptions	142
TABLE 5-35	Flags Field for <code>samu(1M) v</code> Display	142
TABLE 5-36	Control Keys for the <code>w</code> Display	144
TABLE 5-37	<code>samu(1M) w</code> Display Field Descriptions	145

TABLE 5-38	Removable Media Device Display Status Codes	145
TABLE 5-39	File System Display Status Codes	147
TABLE 5-40	Operator Display Device States	147
TABLE 5-41	Device Command Actions	150
TABLE 5-42	Archiver Command Actions	151
TABLE 5-43	Archiver Command Arguments	152
TABLE 5-44	Tracing Command Arguments	166

Preface

This manual, the *Sun StorEdge™ SAM-FS File System Configuration and Administration Guide*, describes the file system software included in Sun StorEdge SAM-FS Version 4, Update 4 (4U4).

The Sun StorEdge SAM-FS product includes both a file system and a storage and archive manager.

The Sun StorEdge SAM-FS software package runs in the following operating system (OS) environments:

- Solaris™ 9 04/03
- Solaris 10

This manual is written for system administrators responsible for installing, configuring, and maintaining Sun StorEdge SAM-FS file systems. You, the system administrator, are assumed to be knowledgeable about Solaris OS procedures, including installing, configuring, creating accounts, performing system backups, and performing other basic Solaris OS system administration tasks.

How This Book Is Organized

This manual contains the following chapters:

- Chapter 1 provides overview information.
- Chapter 2 provides file system configuration information.
- Chapter 3 explains how to perform various tasks, such as initializing a file system, adding a server, adding disk cache, and other system administration activities.
- Chapter 4 explains how to use file system quotas.

- Chapter 5 explains miscellaneous advanced topics such using a multireader file system and performance features.
- Appendix A contains information about troubleshooting the Sun StorEdge SAM-FS software.
- Appendix B explains how to use the `samu(1M)` operator utility.

Using UNIX Commands

This document does not contain information on basic UNIX[®] commands and procedures such as shutting down the system, booting the system, and configuring devices. Refer to one or more of the following for this information:

- Software documentation that you received with your system
- Solaris OS documentation, which is at the following URL:

<http://docs.sun.com>

Shell Prompts

TABLE P-1 shows the shell prompts used in this manual.

TABLE P-1 Shell Prompts

Shell	Prompt
C shell	<i>machine-name%</i>
C shell superuser	<i>machine-name#</i>
Bourne shell and Korn shell	\$
Bourne shell and Korn shell superuser	#

Typographic Conventions

TABLE P-2 lists the typographic conventions used in this manual.

TABLE P-2 Typographic Conventions

Typeface or Symbol	Meaning	Examples
AaBbCc123	The names of commands, files, and directories; on-screen computer output.	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. % You have mail.
AaBbCc123	What you type, when contrasted with on-screen computer output.	% su Password:
<i>AaBbCc123</i>	Book titles; new words or terms; words to be emphasized; and command-line variables to be replaced with a real name or value.	Read Chapter 6 in the <i>User's Guide</i> . These are called <i>class</i> options. You <i>must</i> be <i>root</i> to do this. To delete a file, type <code>rm filename</code> .
[]	In syntax, brackets indicate that an argument is optional.	<code>scmadm [-d sec] [-r n[:n][,n]...] [-z]</code>
{ arg arg }	In syntax, braces and pipes indicate that one of the arguments must be specified.	<code>sndradm -b { phost shost }</code>
\	At the end of a command line, the backslash (\) indicates that the command continues on the next line.	<code>atm90 /dev/md/rdisk/d5 \ /dev/md/rdisk/d1 atm89</code>

Related Documentation

This manual is part of a set of documents that describes the operations of the Sun StorEdge QFS and Sun StorEdge SAM-FS software products. TABLE P-3 shows the complete release 4U4 documentation set for these products.

TABLE P-3 Related Sun StorEdge Documentation

Title	Part Number
<i>Sun StorEdge SAM-FS Installation and Upgrade Guide</i>	819-2754-10
<i>Sun StorEdge SAM-FS Storage and Archive Management Guide</i>	819-2755-10
<i>Sun StorEdge SAM-FS Troubleshooting Guide</i>	819-2756-10
<i>Sun StorEdge QFS Configuration and Administration Guide</i>	819-2758-10
<i>Sun StorEdge QFS Installation and Upgrade Guide</i>	819-2757-10
<i>Sun StorEdge QFS and Sun StorEdge SAM-FS 4.4 Release Notes</i>	819-2759-10

Accessing Sun Documentation Online

The Sun StorEdge QFS software distribution includes PDF files that you can view from Sun's Network Storage documentation web site or from `docs.sun.com`.

▼ To Access Documentation From `docs.sun.com`

This web site contains documentation for Solaris and many other Sun software products.

1. **Go to the following URL:**

`http://docs.sun.com`

The `docs.sun.com` page appears.

2. **Find the documentation for your product by searching for Sun StorEdge QFS in the search box.**

▼ To Access Documentation From Sun's Network Storage Documentation Web Site

This web site contains documentation for Network Storage products.

1. Go to the following URL:

`http://www.sun.com/products-n-solutions/hardware/docs/Software/Storage_Software`

The Storage Software page appears.

2. Click the Sun StorEdge QFS Software link.

Third-Party Web Sites

Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Contacting Sun Technical Support

If you have technical questions about this product that are not answered in this document, go to the following web site:

`http://www.sun.com/service/contacting`

Licensing

For information on obtaining licenses for Sun StorEdge QFS and Sun StorEdge SAM-FS software, contact your Sun sales representative or your authorized service provider (ASP).

Installation Assistance

For installation and configuration services please contact Sun's Enterprise Services at 1-800-USA4SUN or contact your local Enterprise Services sales representative.

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can submit your comments by going to the following web site:

<http://www.sun.com/hwdocs/feedback>

Please include the title and part number of your document with your feedback (*Sun StorEdge SAM-FS File System Configuration and Administration Guide*, part number 819-3635-10).

File System Overview

This chapter covers only the Sun StorEdge SAM-FS file system features. For information on the Sun StorEdge QFS file system, see the *Sun StorEdge QFS Configuration and Administration Guide*.

This chapter contains the following sections:

- “File System Features” on page 1
- “Design Basics” on page 5
- “File Allocation Methods” on page 9

File System Features

The Sun StorEdge SAM-FS file system is a configurable file system that presents a standard UNIX file system interface to users. TABLE 1-1 shows the entire family of Sun StorEdge SAM-FS and Sun StorEdge QFS software.

TABLE 1-1 Product Overview

Product	Components
Sun StorEdge QFS file system	A standalone file system.
Sun StorEdge QFS shared file system	A distributed file system that can be mounted on multiple host systems.

TABLE 1-1 Product Overview (*Continued*)

Product	Components
SAM-FS file system	The file system that is included with the Sun StorEdge SAM-FS software. This file system does not include some of the features found in the Sun StorEdge QFS file system.
SAM-QFS	When the Sun StorEdge QFS and the Sun StorEdge SAM-FS software are used together, you can take advantage of the advanced file system features in the Sun StorEdge QFS product as well as the storage management features of the Sun StorEdge SAM-FS product. This combination is called SAM-QFS.

The Sun StorEdge SAM-FS file system does not require changes to user programs or to the UNIX kernel. Some of the features of the Sun StorEdge SAM-FS file system are described in the following sections.

Volume Management

Sun StorEdge SAM-FS file systems support both striped and round-robin disk access. The master configuration file (`mcf`) and the mount parameters specify the volume management features and enable the file system to recognize the relationships between the devices it controls. This is in contrast to most UNIX file systems, which can address only one device or one portion of a device. Sun StorEdge SAM-FS file systems do not require additional volume management applications. However, if you want to use mirroring for devices in a Sun StorEdge SAM-FS environment, you must obtain an additional package, such as a logical volume manager.

The Sun StorEdge SAM-FS integrated volume management features use the standard Solaris OS device driver interface to pass I/O requests to and from the underlying devices. The Sun StorEdge SAM-FS software groups storage devices into family sets upon which each file system resides.

Support for Paged and Direct I/O

The Sun StorEdge SAM-FS file system supports two different types of I/O: paged (also called cached or buffered I/O) and direct. These I/O types perform as follows:

- When paged I/O is used, user data is cached in virtual memory pages and the kernel writes the data to disk. The standard Solaris OS interfaces manage paged I/O. This is the default type of I/O.

- When direct I/O is used, user data is written directly from user memory to disk. You can specify direct I/O by using the Solaris OS `directio(3C)` function call or the `setfa(1)` command with its `-D` option. By using direct I/O, you can realize substantial performance improvements for large block, sequential, aligned I/O.

High Capacity

The Sun StorEdge SAM-FS software supports files of up to 2^{63} bytes in length. Such very large files can be striped across many disks or RAID devices, even within a single file system. This is true because Sun StorEdge SAM-FS file systems use true 64-bit addressing, in contrast to standard UNIX file systems (UFSs), which are not true 64-bit file systems.

The number of file systems that you can configure is virtually unlimited. The volume manager enables each file system to include up to 252 device partitions, typically disk. Each partition can include up to 16 terabytes of data. This configuration offers virtually unlimited storage capacity.

There is no predefined limit on the number of files in a Sun StorEdge SAM-FS file system. Because the inode space (which holds information about the files) is dynamically allocated, the maximum number of files is limited only by the amount of disk storage available. The inodes are cataloged in the `.inodes` file under the mount point. The `.inodes` file requires 512 bytes of storage per file.

Fast File System Recovery

A key function of a file system is its ability to recover quickly after an unscheduled outage. Standard UNIX file systems require a lengthy file system check (`fsck(1M)`) to repair inconsistencies after a system failure.

A Sun StorEdge SAM-FS file system often does not require a file system check after a disruption that prevents the file system from being written to disk (using `sync(1M)`). In addition, Sun StorEdge SAM-FS file systems recover from system failures without using journaling. They accomplish this dynamically by using identification records, serial writes, and error checking for all critical I/O operations. After a system failure, even multiterabyte-sized Sun StorEdge SAM-FS file systems can be remounted immediately.

vnode Interface

The Sun StorEdge SAM-FS file system is implemented through the standard Solaris OS virtual file system (`vfs/vnode`) interface.

By using the `vfs/vnode` interface, the file system works with the standard Solaris OS kernel and requires no modifications to the kernel for file management support. Thus, the file system is protected from operating system changes and typically does not require extensive regression testing when the operating system is updated.

The kernel intercepts all requests for files, including those that reside in Sun StorEdge SAM-FS file systems. If a file is identified as a Sun StorEdge SAM-FS file, the kernel passes the request to the appropriate file system for handling. Sun StorEdge SAM-FS file systems are identified as type `samfs` in the `/etc/vfstab` file and through the `mount(1M)` command.

Sun StorEdge SAM-FS Archive Management

The Sun StorEdge SAM-FS software combines file system features with a storage and archive management utility. Users can read and write files directly from magnetic disk, or they can access archive copies of files as though they were all on primary disk storage.

When possible, Sun StorEdge SAM-FS software uses the standard Solaris OS disk and tape device drivers. For devices not directly supported under the Solaris OS, such as certain automated library and optical disk devices, Sun Microsystems provides special device drivers in the Sun StorEdge SAM-FS software package.

See the *Sun StorEdge SAM-FS Storage and Archive Management Guide* manual for more information about the storage and archive management features of Sun StorEdge SAM-FS.

Additional File System Features

The following additional features are also supported by the Sun StorEdge SAM-FS file system:

- **Preallocation of file space** – You can use the `setfa(1)` command to preallocate contiguous disk space for fast sequential reads and writes.
- **Application programming interface (API) routines** – API routines enable a program to perform various specialized functions, such as preallocating contiguous disk space or accessing a specific striped group. For more information about these routines, see the `intro_libsam(3)` man page.
- **Adjustable disk allocation units (DAUs)** – The DAU is the basic unit of online storage. The Sun StorEdge SAM-FS file system software includes an adjustable DAU, which is useful for tuning file systems with the physical disk storage device and for eliminating the system overhead caused by read-modify-write operations. You can adjust the DAU size in multiples of 4 kilobytes. For more information, see “Specifying Disk Allocation Units” on page 6.

Design Basics

Sun StorEdge SAM-FS file systems are multithreaded, advanced storage management systems. To take maximum advantage of the software's capabilities, create multiple file systems whenever possible.

Sun StorEdge SAM-FS file systems use a linear search method for directory lookups, searching from the beginning of the directory to the end. As the number of files in a directory increases, the search time through the directory also increases. Search times can become excessive when you have directories with thousands of files. These long search times are also evident when you restore a file system. To increase performance and speed up file system dumps and restores, keep the number of files in a directory under 10,000.

The directory name lookup cache (DNLC) feature improves file system performance. This cache stores the directory lookup information for files whose paths are short (30 characters or less), removing the need for directory lookups to be performed on the fly. The DNLC feature is available in all Solaris OS 9 and later releases.

The following sections cover some additional features that affect file system design:

- "Inode Files and File Characteristics" on page 5
- "Specifying Disk Allocation Units" on page 6

Inode Files and File Characteristics

The types of files to be stored in a file system affect file system design. An inode is a 512-byte block of information that describes the characteristics of a file or directory. This information is allocated dynamically within the file system.

Inodes are stored in the `.inodes` file located under the file system mount point.

Like a standard Solaris OS inode, a Sun StorEdge SAM-FS file system inode contains the file's POSIX standard inode times: file access, file modification, and inode changed times. A Sun StorEdge SAM-FS file system inode includes other times as well, as shown in TABLE 1-2.

TABLE 1-2 Content of `.inode` Files

Time	Incident
access	Time the file was last accessed. POSIX standard.
modification	Time the file was last modified. POSIX standard.

TABLE 1-2 Content of .inode Files (*Continued*)

Time	Incident
changed	Time the inode information was last changed. POSIX standard.
attributes	Time the attributes specific to the Sun StorEdge SAM-FS files system were last changed. Sun Microsystems extension.
creation	Time the file was created. Sun Microsystems extension.
residence	Time the file changed from offline to online or vice versa. Sun Microsystems extension.

Note – If the WORM-FS (write once read many) package is installed, the inode also includes a retention-end date. See “Configuring WORM-FS File Systems” on page 75 for more information.

For more information on viewing inode file information, see “Viewing Files and File Attributes” on page 25.

Specifying Disk Allocation Units

Disk space is allocated in basic units of online disk storage called disk allocation units (DAUs). Whereas sectors, tracks, and cylinders describe the physical disk geometry, the DAU describes the file system geometry. Choosing the appropriate DAU size and stripe size can improve performance and optimize magnetic disk usage. The DAU setting is the minimum amount of contiguous space that is used when a file is allocated.

The following subsections describe how to configure DAU settings and stripe widths.

DAU Settings and File System Geometry

Sun StorEdge SAM-FS file systems use an adjustable DAU. You can use this configurable DAU to tune the file system to the physical disk storage device. This feature minimizes the system overhead caused by read-modify-write operations and is therefore particularly useful for applications that manipulate very large files. For information about how to control the read-modify-write operation, see “Increasing File Transfer Performance for Large Files” on page 85.

Each file system can have its own unique DAU setting, even if it is one of several mounted file systems active on a server. The DAU setting is determined through the `sammkfs(1M)` command when the file system is created. It cannot be changed dynamically.

The following sections introduce the master configuration (`mcf`) file. You create this ASCII file at system configuration time. It defines the devices and file systems used in your Sun StorEdge SAM-FS environment. For details about the `mcf(4)` file, see “System Configuration Tasks” on page 13.

Sun StorEdge SAM-FS File Systems

In a Sun StorEdge SAM-FS file system the file system is defined in your `mcf` file by an Equipment Type value of `ms`. In the `ms` file system, the only device type allowed is type `md`, and both metadata and file data are written to the `md` devices. By default, the DAU on an `md` device is 16 kilobytes.

Dual Allocation Scheme

The `md` devices use a dual allocation scheme, as follows:

- On `md` data devices, the small allocation is 4 kilobytes, and the large allocation is a DAU. The default DAU is 16 kilobytes in a Sun StorEdge SAM-FS file system. You can override this default when the file system is initialized by using the `-a allocation-unit` option to the `sammkfs(1M)` command. The DAU size can be 16, 32, or 64 kilobytes.

When a file is created on an `md` device, the system allocates the first eight addresses of the file in the small allocation. If more space is needed, the file system uses one or more large allocations (DAUs) to expand the file. As a result, I/O performance improves for large files while minimizing the disk fragmentation that can result from many small files.

Depending on the type of file data stored in the file system, a larger DAU size can improve file system performance significantly. For information about tuning file system performance, see Chapter 5 “Advanced Topics” on page 69.

Sun StorEdge QFS Data Alignment

Data alignment refers to matching the allocation unit of the RAID controller with the allocation unit of the file system. The optimal Sun StorEdge SAM-FS file system alignment formula is as follows:

$$\text{allocation-unit} = \text{RAID-stripe-width} \times \text{number-of-data-disks}$$

For example, suppose a RAID-5 unit has nine disks, with one of the nine being the parity disk, making the number of data disks eight. If the RAID stripe width is 64 kilobytes, then the optimal allocation unit is 64 multiplied by 8, which is 512 kilobytes.

Data files are allocated as striped or round-robin through each data disk (md) within the same file system.

A mismatched alignment hurts performance because it can cause a read-modify-write operation.

Stripe Widths on Data Disks

The stripe width is specified by the `-o stripe=n` option in the `mount(1M)` command. If the stripe width is set to 0, round-robin allocation is used.

Stripe Widths on ms File Systems

On `ms` file systems, the stripe width is set at mount time. TABLE 1-3 shows default stripe widths.

TABLE 1-3 `ms` File System Default Stripe Widths

DAU	Default Stripe Width	Amount of Data Written to Disk
16 kilobytes (default)	8 DAUs	128 kilobytes
32 kilobytes	4 DAUs	128 kilobytes
64 kilobytes	2 DAUs	128 kilobytes

For example, if `sammkfs(1M)` is run with default settings, the default large DAU is 64 kilobytes. If no stripe width is specified when the `mount(1M)` command is issued, the default is used, and the stripe width set at mount time is 2.

Note that if you multiply the number in the first column of TABLE 1-3 by the number in the second column, the resulting number is 128 kilobytes. Sun StorEdge SAM-FS file systems operate most efficiently if the amount of data being written to disk is at least 128 kilobytes.

File Allocation Methods

The Sun StorEdge SAM-FS software enables you to specify both round-robin and striped allocation methods.

The rest of this section describes allocation in more detail.

Metadata Allocation

For `ms` file systems, metadata is allocated across the `md` devices.

Inodes are 512 bytes in length. Directories are initially 4 kilobytes in length. TABLE 1-4 shows how the system allocates metadata.

TABLE 1-4 Metadata Allocation

Metadata Type	Allocation Increments for <code>ms</code> File Systems
Inodes (<code>.inodes</code> file)	16-, 32-, or 64-kilobyte DAU
Indirect blocks	16-, 32-, or 64-kilobyte DAU
Directories	4 kilobytes, up to a 3 kilobytes total, then DAU size

Round-Robin Allocation

The round-robin allocation method writes one data file at a time to each successive device in the family set. Round-robin allocation is useful for multiple data streams, because in this type of environment aggregate performance can exceed striping performance.

Round-robin disk allocation enables a single file to be written to a logical disk. The next file is written to the next logical disk, and so on. When the number of files written equals the number of devices defined in the family set, the file system starts over again with the first device selected. If a file exceeds the size of the physical device, the first portion of the file is written to the first device, and the remainder of the file is written to the next device with available storage. The size of the file being written determines the I/O size.

You can specify round-robin allocation explicitly in the `/etc/vfstab` file by entering `stripe=0`.

The following figures depict round-robin allocations. In these figures, file 1 is written to disk 1, file 2 is written to disk 2, file 3 is written to disk 3, and so on. When file 6 is created, it is written to disk 1, restarting the round-robin allocation scheme.

FIGURE 1-1 depicts round-robin allocation on five devices in an ms file system.

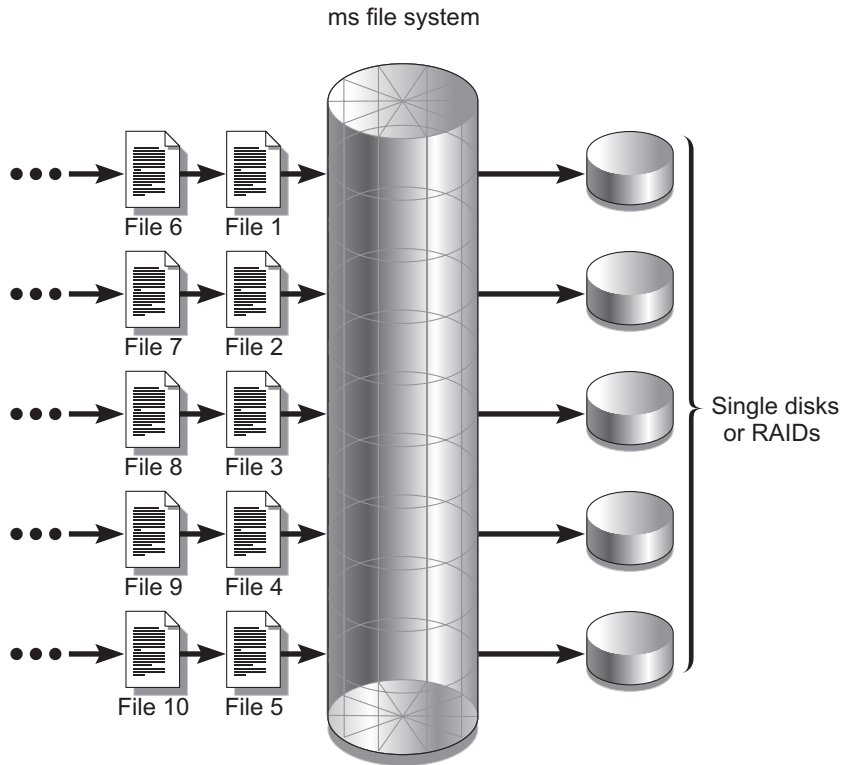


FIGURE 1-1 Round-Robin Allocation in an ms File System Using Five Devices

Striped Allocation

By default, Sun StorEdge SAM-FS file systems use a striped allocation method to spread data over all the devices in the file system family set. Striping is a method of concurrently writing files in an interlaced fashion across multiple devices.

Striping is used when performance for one file requires the additive performance of all the devices. A file system that is using striped devices addresses blocks in an interlaced fashion rather than sequentially. Striping generally increases performance

because it enables multiple I/O streams to simultaneously write a file across multiple disks. The DAU and the stripe width determine the size of the I/O transmission.

In a file system using striping, file 1 is written to disk 1, disk 2, disk 3, disk 4, and disk 5. File 2 is written to disks 1 through 5 as well. The DAU multiplied by the stripe width determines the amount of data written to each disk in a block.

When a Sun StorEdge SAM-FS file system writes a file to an md device, it starts by trying to fit the file into a small DAU, which is 4 kilobytes. If the file does not fit into the first eight small DAUs (32 kilobytes) allocated, the file system writes the remainder of the file into one or more large DAUs.

Multiple active files cause significantly more disk head movement with striped than with round-robin allocation. If I/O is to occur to multiple files simultaneously, use round-robin allocation.

The following figures depict file systems using striped allocations. In these figures, $\text{DAU} \times \text{stripe-width}$ bytes of the file are written to disk 1. $\text{DAU} \times \text{stripe-width}$ bytes of the file are written to disk 2 and so on. The order of the stripe is first-in-first-out for the files. Striping spreads the I/O load over all the disks.

FIGURE 1-2 depicts striping in an ms file system.

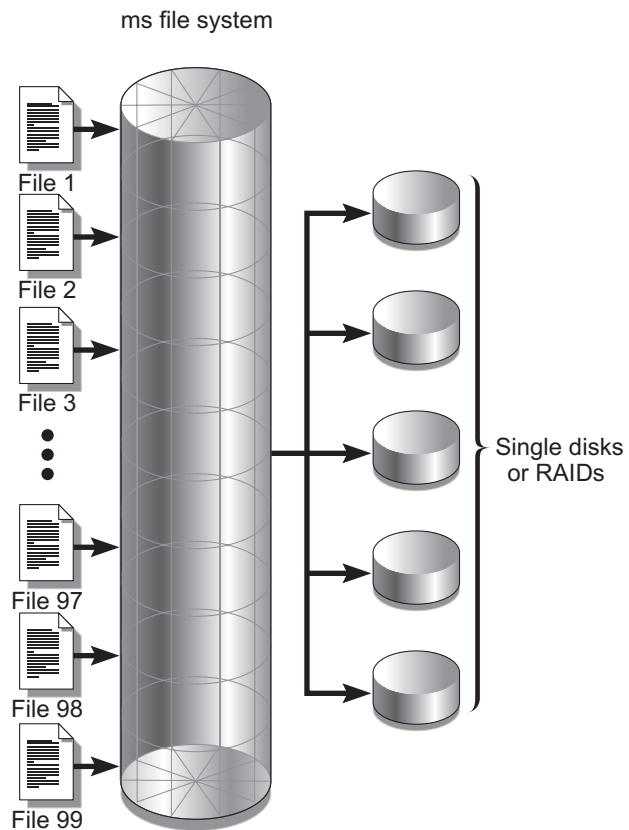


FIGURE 1-2 Striping in an ms File System Using Five Devices

System Configuration Tasks

The installation and configuration process is described completely in the *Sun StorEdge SAM-FS Installation and Upgrade Guide*. This chapter provides additional information about configuring the file system used in the Sun StorEdge SAM-FS environment. This chapter contains the following sections:

- “Using the File System Manager Software” on page 13
- “Function of the `mcf` File” on page 19
- “Initializing a File System” on page 22

Using the File System Manager Software

The File System Manager software is a browser interface tool that enables you to configure, control, protect, and monitor one or more file systems in your network from a central location. To access this central location, you can use the web browser on any host in your network.

The goal of the software is to provide a less complex way than command-line interface (CLI) commands of performing the most common tasks associated with file systems. For instructions on installing the File System Manager software, see the *Sun StorEdge SAM-FS Installation and Upgrade Guide*.

After File System Manager is installed, you can log in to the software using either of two possible user names (`samadmin` and `samuser`) and two different roles (`SAMadmin` or `no role`). The tasks you can perform using File System Manager differ as follows, depending on the user name and the role you assume at login:

- If you log in as `samadmin`, you can choose from one of the following roles:
 - The role of `SAMadmin` grants you full administrator privileges to configure, monitor, control, and reconfigure the devices in your Sun StorEdge SAM-FS environment.

Only the Sun StorEdge SAM-FS administrator should log in using the SAMadmin role. All other users should log in as samuser.

- The role of `no_role` enables you only to monitor the environment. You cannot change or reconfigure it in any way.
- If you log in as `samuser`, you can only monitor the environment. You cannot change or reconfigure it in any way.

With regard to system administration, be aware that the Solaris OS root user on the server that hosts File System Manager is not necessarily the administrator of the File System Manager. Only `samadmin` has administrator privileges for the File System Manager application. The root user is the administrator of the management station.

By default, File System Manager is set up to manage the server on which it is installed. It can also be used to manage other servers running Sun StorEdge SAM-FS software, but those additional servers must first be configured to allow File System Manager access. For instructions on adding additional managed servers, see “To Add an Additional Server for File System Manager Access” on page 16.

▼ To Invoke File System Manager for the First Time

Perform this procedure if you want to invoke File System Manager and use it, rather than CLI commands, to perform file system administration and configuration tasks.

1. **Log in to server where File System Manager is installed, or in to any computer that has network access to it.**
2. **If you upgraded from a previous version of the software, open the web browser and clear the browser cache.**
3. **From the web browser, invoke the File System Manager software.**

The URL is as follows:

```
https://hostname:6789
```

For *hostname*, type the name of the host where the File System Manager software is installed. If you need to specify a domain name in addition to the host name, specify the *hostname* in this format: *hostname.domainname*. Note that this URL begins with `https`, not `http`.

The Sun Java Web Console login page is displayed.

4. **At the User Name prompt, enter `samadmin`.**

5. **At the Password prompt, enter the password you entered during the software setup.**
6. **Click the SAMadmin role.**
Only the Sun StorEdge SAM-FS administrator should log in with the SAMadmin role.
7. **At the Role Password prompt, enter the password you entered in Step 5.**
8. **Click Log In.**
9. **In the Storage section, click File System Manager.**
You are now logged in to File System Manager.

Creating Additional Administrator and User Accounts

You can create additional administrator and guest accounts at any time after the initial File System Manager configuration. These guest accounts are local to the management station.

After being installed, File System Manager creates the following two Solaris operating system (OS) login accounts and the following role:

- Accounts: `samadmin`, `samuser`
- Role: `SAMadmin`

The user account `samadmin` is assigned the `SAMadmin` role. This user has administrator privileges (read and write) for managing File System Manager and Sun StorEdge SAM-FS software.

The user account `samuser` is assigned only Guest privileges. This user has read-only access to Sun StorEdge SAM-FS operations.

If you remove the File System Manager software, the system removes both the `samadmin` and `samuser` Solaris accounts and the `SAMadmin` role. However, the removal scripts do not remove any additional accounts that you create manually. It is your responsibility to use one or both of the following procedures to administer any accounts you add manually.

▼ To Create Additional Administrator Accounts

An administrator account holder has administrator privileges (read and write) for managing File System Manager and the Sun StorEdge SAM-FS software.

1. **Log in to the management station.**
2. **Type** `useradd username`.

3. Type `passwd username`.
4. Type the password as instructed on the screen.
5. Type `usermod -R SAMadmin username`.

Note – Do not use `root` as the *username*.

▼ To Create Additional Guest Accounts

A guest account holder has read-only access to Sun StorEdge SAM-FS operations.

1. Log in to the management station.
2. Type `useradd account_name`.
3. Type `passwd account_name`.
4. Type the password as instructed on the screen.

▼ To Add an Additional Server for File System Manager Access

File System Manager is set up by default to manage the server on which it is installed. It can also be used to manage other servers running Sun StorEdge SAM-FS software, but those additional servers must first be configured to allow File System Manager access.

1. Outside of the browser interface, use `telnet` to connect to the server you want to add. Log in as `root`.
2. Use the `fsmadm(1M) add` command to add the management station (the system on which the File System Manager software is installed) to the list of hosts that can remotely administer this server.

Only hosts that are added to the list through this command can remotely administer the server.

For example:

```
# fsmadm add management_station.sample.com
```

3. To ensure that the management station is successfully added, use the `fsmadm(1M) list` command and verify that your management station is listed in the output.
4. Log in to the File System Manager browser interface as an administrator user.

5. From the Servers page, click Add.

The Add Server window is displayed.

6. In the Server Name or IP Address field, type the name or the IP address of the new server.
7. Click OK.

Using the File System Manager Portal Agent

When the File System Manager software is installed, the File System Manager Portal agent is also installed. This application acts as an information source for the Sun StorEdge Management Portal application. The Sun StorEdge Management Portal provides a customizable, single secure entry point for managing or monitoring the storage environment. It enables IT managers, system administrators, and business unit managers to create views of the storage environment that best meet their needs, and it provides centralized storage management.

By default, the File System Manager Portal agent is disabled. It should be enabled only if you are using the Sun StorEdge Management Portal software. The agent is a simple servlet that responds to requests made by the Sun StorEdge Management Portal software. The agent uses the same underlying software as the File System Manager and provides a thin-scripting, remote API for a small subset of the File System Manager data. The data returned by the agent consists of server name and file system summary information.

When the File System Manager software is uninstalled, the File System Manager Portal agent is also uninstalled. If the agent is running, it is stopped, and the system boot time startup support is removed, as are all log and temporary files.

The following subsections tell you how to start and configure the agent.

▼ To Enable the File System Manager Portal Agent

- Use the following command to start the agent or restart it when it becomes unavailable:

```
# /opt/SUNWfsmgr/bin/fsmgr agent config -a
```

See the `fsmgr(1M)` man page for more options.

About Port Numbers for the File System Manager Portal Agent

The File System Manager Portal Agent uses an instance of the Tomcat web server to provide the remote data access service. This service normally runs on TCP ports 31218 and 31219. You can change the service ports by editing the port numbers defined in the file `/var/opt/SUNWfsmgr/agent/tomcat/conf/server.xml`.

To change the port numbers, first stop the agent using the `fsmgr(1M)` script. Edit the `server.xml` file and modify the port numbers. Then start the agent using the `fsmgr(1M)` script as described above.

If you change the port number in `server.xml`, you must also change it in the Sun StorEdge Management Portal software. By default, the port in that software is set to 31218.

About Configuration and Log Files for the File System Manager Portal Agent

The following files are used for configuring and logging data from the File System Manager Portal Agent:

- `/etc/opt/SUNWfsmgr/agent/conf.sh` – The configuration script that is used when starting the Tomcat process. It defines location of Tomcat, Java, and other critical components.
- `/var/opt/SUNWfsmgr/agent/tomcat/logs` – The directory that contains the following log files:
 - `catalina.out` – The general log file. It contains the log message output from both Tomcat and the agent servlet. Any errors cause log messages to be written to this file.
 - `fsmgr.date-stamp.log` – The application and servlet log file. It contains messages specific to the loading and the running of the agent servlet. It also contains stack trace and fatal error information from the underlying software.

To verify that the agent is running, check the `catalina.out` log file or use the `ps` and `grep` commands to find the agent process:

```
# /usr/ucb/ps -augxww | grep SUNWfsmgr/agent/tomcat
```

Function of the mcf File

The master configuration file (mcf), located in `/etc/opt/SUNWsamfs/mcf`, describes all devices that are under the control of, or used by, the Sun StorEdge SAM-FS software. When you create this ASCII file at system configuration time, you declare attributes for each device, and you group the devices in each file system into family sets.

The `mcf(4)` file contains the information that these file systems need in order to identify and organize RAID and disk devices into file systems. It also contains entries for each automated library or device included in a file system. A sample `mcf(4)` file is located in `/opt/SUNWsamfs/examples/mcf`.

An `mcf(4)` file consists of lines of specification code divided into six columns, or fields, as shown in CODE EXAMPLE 2-1.

CODE EXAMPLE 2-1 Fields in an mcf File

Equipment Identifier	Equipment Ordinal	Equipment Type	Family Set	Device State	Additional Parameters
-------------------------	----------------------	-------------------	---------------	-----------------	--------------------------

Follow these rules when entering data in the `mcf(4)` file:

- Enter either space or tab characters between the fields in the file.
- You can include comment lines in an `mcf(4)` file. Comment lines start with a pound character (#).
- Some fields are optional. Use a dash character (-) to indicate that an optional field contains no meaningful information.

For more information about writing the `mcf` file, see the `mcf(4)` man page. You can also use File System Manager to automatically create an `mcf` file. For information about installing File System Manager, see the *Sun StorEdge SAM-FS Installation and Upgrade Guide*. For information about using File System Manager, see its online help.

The following subsections describe each field in an `mcf(4)` file:

- “The Equipment Identifier Field” on page 20
- “The Equipment Ordinal Field” on page 20
- “The Equipment Type Field” on page 20
- “The Family Set Field” on page 21
- “The Device State Field” on page 21
- “The Additional Parameters Field” on page 22

The Equipment Identifier Field

The Equipment Identifier field is a required field. Use the Equipment Identifier field to specify the following kinds of information:

- The file system name. If this field contains a file system name, it must be identical to the name in the Family Set field, and the subsequent lines in the `mcf(4)` file must define all the disks or devices included in the file system. More than one file system can be declared in an `mcf(4)` file. Typically, the first data line in an `mcf(4)` file declares the first file system, and subsequent lines specify the devices included in the file system. Other file systems declared in the `mcf(4)` file can be preceded by a blank comment line for readability. File system names must start with an alphabetic character and can contain only alphabetic characters, numeric characters, or underscore (`_`) characters.
- A disk partition or slice description. A `/dev/` entry in this field identifies a disk partition or slice.
- An automated library or optical drive description. A `/dev/samst` entry identifies an automated library or optical drive. If you are configuring a network-attached automated library, see the *Sun StorEdge SAM-FS Installation and Upgrade Guide* for more information.
- A tape drive description. This entry can be in one of two forms:
 - A `/dev/rmt` entry.
 - A path to a symbolic link that points to the same file to which the `/dev/rmt` link points. If you specify a tape drive in this manner, be sure to create the link before mounting the file system.

If the Equipment Identifier field contains the name of a file system, it is limited to 31 characters. For all other content, this field is limited to 127 characters.

The Equipment Ordinal Field

For each row in the `mcf(4)` file, the Equipment Ordinal field must contain a numeric identifier for the file system component or device being defined. Specify a unique integer between 1 and 65534, inclusive. This is a required field.

The Equipment Type Field

Enter a 2-, 3-, or 4-character code for the Equipment Type field. This is a required field.

The Equipment Type field in a Sun StorEdge SAM-FS file system can contain any of the values shown in TABLE 2-1.

TABLE 2-1 Sun StorEdge SAM-FS Equipment Type Field

Equipment Type Field Content	Meaning
ms	Defines a Sun StorEdge SAM-FS file system.
md	Defines a striped or round-robin device for storing file data.

Besides the file system equipment types, other codes are used to identify automated libraries and other devices. For more information about specific equipment types, see the `mcf(4)` man page.

The Family Set Field

The Family Set field contains the name for a group of devices. This is a required field.

Family set names must start with an alphabetic character and can contain only alphabetic characters, numeric characters, or underscore (`_`) characters.

The lines that define the disk devices in a file system must all contain the same family set name. The software uses the family set name to group devices together as a file system. It physically records the family set name on all of the devices in the file system when the `sammkfs(1M)` command is issued. You can change this name by using the `-F` and `-R` options together in the `samfsck(1M)` command. For more information about the `sammkfs(1M)` command, see the `sammkfs(1M)` man page. For more information about the `samfsck(1M)` command, see the `samfsck(1M)` man page.

The lines that define the devices in an automated library, as well as the lines that define the devices in the library's associated drives, must contain the same family set name.

For a standalone, manually loaded removable media device, this field can contain a dash (`-`).

The Device State Field

The Device State field specifies the state of the device when the file system is initialized. Valid device states are `on` and `off`. This is an optional field. If you do not want to specify a value, insert a dash character (`-`) to indicate that this field is omitted.

The Additional Parameters Field

For a Sun StorEdge SAM-FS file system, the Additional Parameters field is optional and can be left blank. By default, library catalog files are written to `/var/opt/SUNWsamfs/catalog/family-set-name`. Use this field if you want to specify an alternative path to the library catalog file.

Interactions Among File Settings, Options, and Directives

The `mcf(4)` file defines each file system, but file system behavior depends on interactions among default system settings, settings in the `/etc/vfstab` file, settings in the `samfs.cmd` file, and options in the `mount(1M)` command.

You can specify some mount options, such as the stripe width, in more than one place. When this happens, settings in one place can override the settings in another.

For information about the various ways to specify mount options, see “Setting Up Mount Parameters” on page 35.

Initializing a File System

If you want to create a new file system or replace an old or damaged file system, you must initialize it using the `sammkfs(1M)` command.

The `sammkfs(1M)` command constructs new file systems, and its `-a allocation-unit` option enables you to specify the DAU setting.

Version 4U2 and later releases of the Sun StorEdge SAM-FS software support two different superblock designs. In CODE EXAMPLE 2-2, the `samfsinfo(1M)` command output shows that the `samfs1` file system is using a version 2 superblock.

CODE EXAMPLE 2-2 `samfsinfo(1M)` Command Example

# samfsinfo samfs1				
name:	samfs1	version:		2
time:	Wed Feb 21 13:32:18 1996			
count:	1			
capacity:	001240a0	DAU:		16
space:	000d8ea0			
ord	eq	capacity	space	device
0	10	001240a0	000d8ea0	/dev/dsk/c1t1d0s0

Be aware of the following operational and feature differences that pertain to these superblocks:

- Releases before 4U0 support only the version 1 superblock design.
- The 4U0 and later releases support the version 2 superblock. If you installed the 4U0 software as an upgrade, you must use the 4U0 or later `sammkfs(1M)` command to reinitialize your existing file systems before you attempt to use any of the features that depend on the version 2 superblock. Certain features, such as access control lists (ACLs), are supported only in the version 2 superblock. Reinitializing a file system is described as a step in the software installation upgrade process, but you can do this any time after the software is installed.



Caution – File systems that use a version 2 superblock cannot revert to a release before 4U0. You cannot use 4U4 release software to create a version 1 superblock.

For more information about features that require a version 2 superblock, or about using the `sammkfs(1M)` command to create the version 2 superblock, see the *Sun StorEdge SAM-FS Installation and Upgrade Guide*.

The following example shows the `sammkfs(1M)` command in its simplest form, with the file system name as its only argument:

```
# sammkfs samqfs1
```

The preceding command builds a version 2 superblock for a Sun StorEdge SAM-FS file system.

For more information about the `sammkfs(1M)` command, its options, and the implications of the version 1 and version 2 superblocks, see the `sammkfs(1M)` man page.

Performing Operational Tasks

This chapter presents topics related to file system operations. This chapter contains the following sections:

- “Viewing Files and File Attributes” on page 25
 - “Propagating Configuration File Changes to the System” on page 31
 - “Setting Up Mount Parameters” on page 35
 - “Unmounting a File System” on page 38
 - “Adding Disk Cache to a File System” on page 39
 - “Re-creating a File System” on page 41
-

Viewing Files and File Attributes

The attributes specific to Sun StorEdge SAM-FS file systems include both user settings and general file states. This section describes these characteristics and describes how to view them using the `sls` command.

File Attributes and File States

The user-specified attributes of a file and its system-specified states are stored in the file’s inode. You can use the `sls(1) -D` command to display these inode attributes. For more information about `sls(1)` options, see the `sls(1)` man page.

A user can specify the following commands to set attributes:

- `archive(1)`
- `ssum(1)`
- `release(1)`

- `segment(1)`
- `setfa(1)`
- `stage(1)`

Users can set attributes from within applications by specifying the following application programming interface (API) routines:

- `sam_archive(3)`
- `sam_release(3)`
- `sam_segment(3)`
- `sam_setfa(3)`
- `sam_ssum(3)`
- `sam_stage(3)`

TABLE 3-1 shows the user-specified attributes that are listed in the inode.

TABLE 3-1 User-Specified File Attributes

File Attribute	Description
<code>archive -C</code>	Marks the file for concurrent archiving. This means that the file can be archived even if it is open for a write operation. You can use the <code>archive(1)</code> command to set this attribute.
<code>archive -n</code>	Marks the file to never be archived. The superuser can use the <code>archive(1)</code> command to set this attribute.
<code>release -a</code>	Marks the file to be released as soon as one archive copy is made. You can set this attribute from within the <code>archiver.cmd</code> file or by using the <code>release(1)</code> command.
<code>release -n</code>	Marks the file to never be released. You can set this attribute from within the <code>archiver.cmd</code> file, or the superuser can use the <code>release(1)</code> command to set it.
<code>release -p</code>	Marks the file for partial release. You can set this attribute from within the <code>archiver.cmd</code> file or by using the <code>release(1)</code> command.
<code>stage -a</code>	Marks the file for associative staging. You can set this attribute from within the <code>archiver.cmd</code> file or by using the <code>stage(1)</code> command.
<code>stage -n</code>	Marks the file to never be staged. This signifies direct access to removable media cartridges. You can set this attribute from within the <code>archiver.cmd</code> file, or the superuser can use the <code>stage(1)</code> command to set it.
	Not supported on Sun StorEdge QFS shared file system clients.
<code>setfa -D</code>	Marks the file for direct I/O.

TABLE 3-1 User-Specified File Attributes (*Continued*)

File Attribute	Description
<code>setfa -sm</code>	Marks the file for allocation with a stripe width of <i>m</i> .
<code>segment nm stage_ahead x</code>	Marks the file for segmentation. The <i>nm</i> notation indicates that the segment is <i>n</i> megabytes in size. The <code>stage_ahead x</code> attribute indicates the number of segments (<i>x</i>) to be staged ahead. You can use the <code>segment(1)</code> command to set this attribute.

You can set the attributes shown in TABLE 3-1 on both files and directories. After directory attributes are set, files that are created in the directory inherit all the directory attributes. Files created before an attribute is applied to the parent directory do not inherit directory attributes.

If you have purchased the optional WORM-FS package, you can also apply WORM (write once read many) attributes to a file, and set the file's retention period. See "Configuring WORM-FS File Systems" on page 75 for details.

System-Specified File States

TABLE 3-2 shows the various states that the file systems set for a file. These states are stored in the inode.

TABLE 3-2 System-Specified File States

Attribute	Definition
<code>archdone</code>	Indicates that the file's archive requirements have been met. There is no more work the archiver must do on the file. The archiver sets this attribute. It cannot be set by a user. Note that <code>archdone</code> does not necessarily indicate that the file has been archived.
<code>damaged</code>	The file is damaged. The stager or the <code>samfsrestore(1M)</code> command sets this attribute. You can use the <code>undamage(1M)</code> command to reset this attribute to undamaged. If this attribute has been set by the <code>samfsrestore(1M)</code> utility, it means that no archive copies existed for the file at the time a <code>samfsdump(1M)</code> was taken. You can reset this attribute to undamaged, but the file might still be unrecoverable.
<code>offline</code>	The file data has been released. The releaser sets this attribute. You can also set this attribute by using the <code>release(1)</code> command.

Users can gather information about file states by using the `s1s(1)` command, which is described in "Displaying File Information" on page 28.

Displaying File Information

The Sun StorEdge SAM-FS `sls(1)` command extends the standard UNIX `ls(1)` command and provides more information about a file. CODE EXAMPLE 3-1 shows detailed `sls(1)` command output that displays the inode information for file `hgc2`.

CODE EXAMPLE 3-1 `sls(1)` Output in a Sun StorEdge SAM-FS Environment

```
# sls -D hgc2
hgc2:
mode: -rw-r--r--  links: 1  owner: root      group: other
length: 14971  admin id: 0  inode: 30.5
archdone;
segments 3, offline 0, archdone 3, damaged 0;
copy 1: ---- Jun 13 17:14      2239a.48  1t MFJ192
copy 2: ---- Jun 13 17:15      9e37.48  1t AA0006
access: Jun 13 17:08  modification: Jun 13 17:08
changed: Jun 13 17:08  attributes: Jun 13 17:10
creation: Jun 13 17:08  residence: Jun 13 17:08
```

TABLE 3-3 describes the meaning of each row of `sls(1)` output shown in CODE EXAMPLE 3-1.

TABLE 3-3 `sls(1)` Output Explanation

Line Number	Tag	Content
1	mode:	The file's mode and permissions, the number of hard links to the file, the owner of the file, and the group to which the owner belongs.
2	length:	<p>The file's length in bytes, the file's admin ID number, and the file's inode number.</p> <p>By default, the admin ID number is 0. If this number is greater than 0, it indicates the file's accounting category for counting files and blocks. You can set this number to a value greater than 0 even when file system quotas are not enabled on this file system. For information about file system quotas, see "Administering File System Quotas" on page 45.</p> <p>The inode number is a two-part number that contains the inode number itself, followed by a period (.), followed by the inode generation number.</p>
3	archdone;	The file attributes specific to the file. For more information about this line, see the <code>sls(1)</code> man page.

TABLE 3-3 `sls(1)` Output Explanation (*Continued*)

Line Number	Tag	Content
4	segments	<p>The segment index information. This line does not appear unless the file is a segment index. The general format for this line is as follows:</p> <p>segments <i>n</i>, offline <i>o</i>, archdone <i>a</i>, damaged <i>d</i>;</p> <p>segments <i>n</i> shows the total number of data segments for this file. In this example, there are 3.</p> <p>offline <i>o</i> shows the number of data segments offline. In this example, there are no offline segments.</p> <p>archdone <i>a</i> shows the number of segments for which the archiving requirements have been met. In this example, there are 3.</p> <p>damaged <i>d</i> shows the number of damaged segments. In this example, there are no damaged segments.</p>

TABLE 3-3 `sls(1)` Output Explanation (*Continued*)

Line Number	Tag	Content
5, 6	<code>copy 1:</code> , <code>copy 2:</code>	<p>Archive copy lines. The <code>sls(1)</code> command displays one archive copy line for each active or expired archive copy.</p> <p>The four positions in this line indicate the following:</p> <p>1 – Either an expired or an active entry.</p> <ul style="list-style-type: none">• An <code>S</code> indicates that the archive copy is expired. That is, the file was modified and this archive copy is a previous version of the file.• A <code>U</code> indicates that the copy has been unarchived. Unarchiving is the process by which archive entries for files or directories are deleted.• A dash (<code>-</code>) indicates that the archive copy is active and valid. <p>2 – Whether the archive copy is to be rearchived.</p> <ul style="list-style-type: none">• An <code>r</code> indicates that the archive copy is scheduled to be rearchived by the archiver.• A dash (<code>-</code>) indicates that the archive copy is not to be rearchived by the archiver. <p>3 – Unused.</p> <p>4 – Whether the copy is damaged or undamaged.</p> <ul style="list-style-type: none">• A <code>D</code> indicates that the archive copy is damaged. A damaged archive copy is not a candidate for staging.• A dash (<code>-</code>) indicates that the archive copy is not damaged. It is a candidate for staging. <p>The format of the rest of the archive copy line is as follows:</p> <ul style="list-style-type: none">• The date and time the archive copy was written to the archive media.• Two hexadecimal numbers separated by a decimal point (<code>.</code>). The first hexadecimal number (<code>2239a</code>) indicates the position of the beginning of the archive file on the cartridge. The second hexadecimal number (<code>48</code>) is the file byte offset (divided by 512) of this copy in the archive file.• The media type and the Volume Serial Name (VSN) where the archive copy resides.
7	<code>access:</code>	The time the file was last accessed and modified.
8	<code>changed:</code>	The time the file content and the file's attributes were last changed. A <code>retention-end</code> date also appears here if this is a WORM-FS file. See "Configuring WORM-FS File Systems" on page 75.
9	<code>creation:</code>	The time the file was created and became resident in the file system.

Retention Line Explanation

If you are using the optional WORM-FS package, a retention line will also appear in the `sls(1)` output. The format of the retention line is as follows:

```
retention: active retention-period: 3y 0d 0h 0m
```

This indicates whether a retention period has been set for this file and, if so, what its length is. The `retention-end` date indicates the date on which the retention period expires. For more information about using the WORM-FS feature, see “Configuring WORM-FS File Systems” on page 75.

Checksum Line Explanation

If a file has checksum-related attributes (`generate`, `use`, or `valid`), the `sls(1)` command returns a checksum line. You can use the `ssum(1)` command to set these attributes. The format of the checksum line is as follows:

```
checksum: gen use val algo: 1
```

The system displays the preceding line if checksum attributes are set for a file. You can interpret this line as follows:

- If the `generate` attribute is not set, `no_gen` appears in place of `gen`.
- If the `use` attribute is not set, `no_use` appears.
- If the file has been archived and a checksum has been computed, `val` appears.
- If the file has not been archived or if no checksum has been computed, `not_val` appears.
- The keyword `algo` precedes the numeric algorithm indicator that specifies the algorithm that is used to generate the checksum value.

Propagating Configuration File Changes to the System

This section describes how to propagate configuration file changes throughout the system. The procedures describe the propagation of changes for the following files:

- `mcf(4)`

- `defaults.conf`
- `archiver.cmd` file
- `stager.cmd` file

You must perform these procedures under the following circumstances:

- If you update any of these files in order to add, delete, or correct information.
- If you create or update a Sun StorEdge SAM-FS `archiver.cmd`, `defaults.conf`, or `stager.cmd` file after your Sun StorEdge SAM-FS file system is already operational.

The following sections describe these procedures:

- “To Change `mcf(4)` or `defaults.conf(4)` File System Information in a Sun StorEdge SAM-FS Environment” on page 32
- “To Change `mcf(4)` or `defaults.conf(4)` Removable Media Drive Information” on page 33

▼ To Change `mcf(4)` or `defaults.conf(4)` File System Information in a Sun StorEdge SAM-FS Environment

1. Use `vi(1)` or another editor to edit the file and change the file system information.
2. If you are changing the `mcf(4)` file, use the `sam-fsd(1M)` command to check the `mcf(4)` file for errors:

```
# sam-fsd
```

If the output from this command shows errors, correct them before proceeding to the next step.

3. If you are removing or changing information related to one or more file systems, issue a `samcmd(1M)` `aridle` command to idle the archiver for each affected file system defined in the `mcf(4)` file.

Use this command in the following format:

```
samcmd aridle fs.fsname
```

For *fsname*, specify the name of the file system.

4. If you are removing or changing information related to one or more drives, issue a `samcmd(1M) idle` command to idle the archiver for each equipment ordinal assigned to each affected drive in the `mcf(4)` file.

Use this command in the following format:

```
samcmd idle eq
```

For *eq*, specify the Equipment Ordinal number of the drive.

5. Issue the `umount(1M)` command to unmount each file system affected by the changes.

For instructions on unmounting the file system, see “Unmounting a File System” on page 38.

6. Use the `samd(1M) config` command to propagate the changes:

```
# samd config
```

7. Use the `mount(1M)` command to remount the file systems you unmounted.

For more information about these files, see the `defaults.conf(4)` or `mcf(4)` man pages.

▼ To Change `mcf(4)` or `defaults.conf(4)` Removable Media Drive Information

1. Edit the file and change the removable media drive information.
2. If you are changing the `mcf(4)` file, use the `sam-fsd(1M)` command to check the `mcf(4)` file for errors:

```
# sam-fsd
```

If the output from this command shows errors, correct them before proceeding to the next step.

3. If you are removing or changing information related to one or more file systems, issue a **samcmd(1M) aridle** command to idle the archiver for each affected file system defined in the **mcf(4)** file.

Use this command in the following format:

```
samcmd aridle fs.fsname
```

For *fsname*, specify the name of the file system.

4. If you are removing or changing information related to one or more drives, issue a **samcmd(1M) idle** command for the Equipment Ordinal number assigned to each affected drive in the **mcf(4)** file.

Use this command in the following format:

```
samcmd idle eq
```

For *eq*, specify the Equipment Ordinal number of the drive.

5. Use the **samd(1M) stop** command to stop all removable media activity:

```
# samd stop
```

6. Use the **samd(1M) config** command to propagate the changes and restart the system:

```
# samd config
```

7. Use the **samd(1M) start** command to re-start all removable media activity:

```
# samd start
```

For more information about these files, see the **defaults.conf(4)** or **mcf(4)** man pages.

▼ To Change **archiver.cmd(4)** or **stager.cmd(4)** Information

1. Use **vi(1)** or another editor to edit the **archiver.cmd(4)** or **stager.cmd(4)** file.

2. If you are changing an existing `archiver.cmd(4)` file, use the `archiver(1M) -lv` command to validate the changes you made in the `archiver.cmd(4)` file.
3. Save and close the file.
4. Use the `samd(1M) config` command to propagate the file changes and restart the system:

```
# samd config
```

Setting Up Mount Parameters

You can mount a Sun StorEdge SAM-FS file system by using the Solaris OS `mount(1M)` command.

Mount parameters are used to manipulate file system characteristics. There are several ways to specify mount parameters. Methods at the top of the hierarchy override methods lower in the hierarchy. You can specify mount options in the following ways, listed in hierarchical order from the top down:

- With the `mount(1M)` command using command line options. Options specified on the Solaris OS `mount(1M)` command line override other options specified in the `/etc/vfstab` file, directives specified in the `samfs.cmd` file, and system default settings.
- As `/etc/vfstab` file settings.
- In the `samfs.cmd` file using directives.
- As system defaults. The default system settings are the configurable settings already defined for your Solaris OS. You can override the system settings with specifications in the `samfs.cmd` file, in the `/etc/vfstab` file, and in the `mount(1M)` command.

You can also specify mount options by using the `samu(1M)` operator utility or the `samcmd(1M)` command. Mount options enabled or disabled in this way persist until the file system is unmounted.

The following subsections describe ways to specify mount options. The *Sun StorEdge SAM-FS Installation and Upgrade Guide* also includes information about mounting a file system.

The mount(1M) Command

The Solaris OS `mount(1M)` command mounts the file system and enables you to specify settings that override the settings specified in the `/etc/vfstab` file and in the `/etc/opt/SUNWsamfs/samfs.cmd` file. For example, you can specify the stripe width, read-ahead, write-behind, and high and low water marks for disk cache utilization.

One way to use the `mount(1M)` command in conjunction with the `samfs.cmd` file is to use the `samfs.cmd` file as your main location for mount options and to use options on the `mount(1M)` command when experimenting with or tuning your system.

For example, the following command mounts file system `qfs1` at `/work` with `setuid` execution disallowed. The `qfs1` file system name is the Equipment Identifier. This also appears in the `mcf(4)` file's Equipment Identifier field for this file system. To specify more than one mount option, separate each with a comma.

```
# mount -o nosuid qfs1 /work
```

For more information about the `mount(1M)` command, see the `mount_samfs(1M)` man page.

The /etc/vfstab File

Each Sun StorEdge SAM-FS file system that is defined in the `mcf(4)` file must have a line in the `/etc/vfstab` Solaris OS system file. This is required for mounting the file system.

The following is an example of a file system line in the `/etc/vfstab` file:

```
qfs1      -      /qfs      samfs      -      yes      stripe=0
```

From left to right, the fields shown indicate the following:

- The file system family set name.
- The file system to `samfsck(1M)`.
- The mount point.
- The file system type. This is always `samfs`.
- The `samfsck(1M)` pass.
- Mount-at-boot options.
- Mount parameters, separated by commas, without intervening spaces.

The fields in the `/etc/vfstab` file must be separated by either space or tab characters.

The mount parameters field can contain any of the mount parameters listed as arguments to the `-o` option on the `mount_samfs(1M)` man page. These parameters are nearly identical to those that you can specify as directive lines in the `samfs.cmd` file or as arguments to the `-o` option in the `mount(1M)` command. As with the `samfs.cmd` file, you can include specifications for various I/O settings, read-ahead, write-behind, the stripe width, various storage and archive management settings, and other features.

For more information about possible mount parameters, see the `mount_samfs(1M)` man page. For more information about modifying the `/etc/vfstab` file, see the `vfstab(4)` man page.

The `samfs.cmd` File

The `/etc/opt/SUNWsamfs/samfs.cmd` file enables you to specify mount parameters for all of your Sun StorEdge SAM-FS file systems. This file can be useful when you have multiple file systems configured and you want to specify the same mount parameters for all of them.

Using this file enables you to define all mount parameters in one place in an easily readable format. Directives specified toward the beginning of this file are global directives and apply to all Sun StorEdge SAM-FS file systems. The second part of this file enables you to indicate the specific parameters that you want to apply to each individual file system. The ability to specify the common parameters once, and in only one place, differentiates this file from the `/etc/vfstab` file, in which you must specify all mount parameters for each file system.

The mount parameters that can be specified in the `samfs.cmd` file are nearly identical to those that you can specify in the `/etc/vfstab` file or as arguments to the `-o` option with the `mount(1M)` command. The possible mount parameters you can specify pertain to I/O settings, read-ahead, write-behind, the stripe width, various storage and archive management settings, WORM-FS, and other features. For more information about the mount parameters that can be specified in this file, see the `samfs.cmd(4)` man page.

In the `samfs.cmd` file, directives are written one per line. The file can contain comments, which must begin with a pound character (`#`). Characters that appear to the right of the pound character are treated as comments.

For a directive that applies globally to all file systems, place the line before any `fs =` line. For a directive that is specific to a particular file system, start the line with `fs =` and place it after all global directives. Directives specific to a particular file system override global directives.

CODE EXAMPLE 3-2 shows a sample `samfs.cmd` file that sets the low and high water marks for disk cache utilization for all file systems and specifies individualized parameters for two specific file systems.

CODE EXAMPLE 3-2 Example `samfs.cmd` File

```
low = 50
high = 75
fs = samfs1
    high = 65
    writebehind = 512
    readahead = 1024
fs = samfs5
    partial = 64
```

The directives in the `samfs.cmd` file override any default system settings, but arguments to the `mount(1M)` command override any directives in this file. Entries in the `/etc/vfstab` file also override directives specified in the `samfs.cmd` file.

For information about the `mount(1M)` command, see the `mount_samfs(1M)` man page. For information about which directives can be entered in the `samfs.cmd` file, see the `samfs.cmd(4)` man page.

Unmounting a File System

You can use the Solaris OS `umount(1M)` command to unmount Sun StorEdge SAM-FS file systems.

In Sun StorEdge SAM-FS environments, you must issue commands to stop the archiver before unmounting the file system. The following procedure shows you how to idle the archiver and unmount the file system.

▼ To Unmount a Sun StorEdge SAM-FS File System

1. Issue a `samcmd(1M) aridle fs.file-system` command for the file system.

For example:

```
# samcmd aridle fs.samqfs2
```

This stops archiving operations for the file system at a logical place before stopping the daemons.

2. Issue a `samd(1M) stop` command:

```
# samd stop
```

This command kills the `sam-amld` daemon.

3. Unmount the file system:

```
# umount /samqfs
```

Several conditions can be present in a file system at unmounting time, so you might need to issue the `umount(1M)` command a second time. If the file system still does not unmount, use `unshare(1M)`, `fuser(1M)`, or another command in conjunction with the `umount(1M)` command. For more information on unmounting procedures, see the *Sun StorEdge SAM-FS Installation and Upgrade Guide*.

Adding Disk Cache to a File System

To increase the disk cache for a file system, you add disk partitions or disk drives, and then update the `mcf(4)` file and use the `samgrowfs(1M)` command to expand the file system. You do not need to reinitialize or restore the file system.

Note that when adding disks or partitions, the system might update the Equipment Ordinal of the historian. The system automatically generates the Equipment Ordinal of the historian unless you specifically call it out. For more information, see the `historian(7)` man page.

When making changes to the `mcf(4)` file, be aware of the following:

- You can configure up to 252 disk partitions in a file system.
- If you want to add new partitions for data, add them to the `mcf(4)` file after the existing disk partitions.
- Do not change the Equipment Identifier name in the `mcf(4)` file. If the name in the `mcf(4)` file does not match the name in the superblock, the file system can no longer be mounted. Instead, the following message is logged in `/var/adm/messages`:

```
WARNING SAM-FS superblock equipment identifier <id>s on eq <eq>
does not match <id> in mcf
```

▼ To Add Disk Cache to a File System

1. **Use the `umount(1M)` command to unmount the file system you want to expand.**
For more information about unmounting a file system, see “Unmounting a File System” on page 38.
2. **If you want to rename the file system during this procedure, use the `samfsck(1M)` command with its `-R` and `-F` options to rename the file system.**
For more information about this command, see the `samfsck(1M)` man page.
3. **Edit the `/etc/opt/SUNWsamfs/mcf` file to add the disk cache.**
4. **Issue the `sam-fsd(1M)` command to check for errors in the `mcf(4)` file:**

```
# sam-fsd
```

If the output from this command shows errors, correct them before proceeding to the next step.

5. **Issue the `samd(1M) config` command to propagate the `mcf(4)` file changes to the system:**

```
# samd config
```

For more information, see the `samd(1M)` man page.

6. Issue the `samgrowfs(1M)` command on the file system that is being expanded.

For example, type the following command to expand file system `samfs1`:

```
# samgrowfs samfs1
```

If you renamed the file system, run the `samgrowfs(1M)` command using the new name. For more information about this command, see the `samgrowfs(1M)` man page.

7. Mount the file system.

For information about mounting a Sun StorEdge SAM-FS file system, see the `mount_samfs(1M)` man page.

Re-creating a File System

In order to do any of the following, you must re-create the file system:

- Change disks or partitions
- Add disks or partitions
- Remove disks or partitions

This section describes this procedure.

▼ To Back Up and Re-create a File System

1. Back up all site-customized system files and configuration files.

Depending on your software, these files might include `mcf(4)`, `archiver.cmd`, `defaults.conf`, `samfs.cmd`, `inquiry.conf`, and so on. Back up these files for all file systems in your Sun StorEdge SAM-FS environment. Also make sure that you have backup copies of files in the `/etc/opt/SUNWsamfs` directory, files in the `/var/opt/SUNWsamfs` directory, library catalogs, the historian, and any parameter files for network-attached automated libraries.

If you do not know the names and locations of your catalog files, examine the `mcf(4)` file with `vi(1)` or another viewing command and find the first `rb` entry in the `mcf(4)` file. That entry contains the name of the library catalog file. If no catalog file location is specified, then the system is using the default location (`/var/opt/SUNWsamfs/catalog`).

2. Ensure that each file system to be modified is backed up.

File systems should be backed up regularly according to your site's policies. If you are comfortable with the backup files that already exist for your file systems, there is no need to back them up again now. If, however, you need to back up your file systems to preserve information created since the last dump file was created, do so now. For information about how to create a dump file, see the *Sun StorEdge SAM-FS Installation and Upgrade Guide*.

Note that the `samfsdump(1M)` command issues warnings when creating the dump file if it encounters unarchived files in the file system. If warnings are issued, these files need to be archived before unmounting the file systems.

3. Unmount the file system.

For instructions, see “Unmounting a File System” on page 38.

4. If you want to rename the file system during this procedure, use the `samfsck(1M)` command with its `-R` and `-F` options.

For more information, see the `samfsck(1M)` man page.

5. Edit the `/etc/opt/SUNWsamfs/mcf` file to add, change, or remove partitions.

For more information, see “Adding Disk Cache to a File System” on page 39.

6. Type the `sam-fsd(1M)` command to check for errors in the `mcf(4)` file:

```
# sam-fsd
```

If the output from this command indicates that there are errors in the `mcf(4)` file, correct them before proceeding to the next step.

7. Issue the `samd(1M) config` command to propagate the `mcf(4)` file changes to the system:

```
# samd config
```

For more information, see the `samd(1M)` man page.

8. Issue the `sammkfs(1M)` command to re-create the file system.

For example, the following command creates `samfs10`:

```
# sammkfs samfs10
```

9. Issue the `mount(1M)` command to mount the file system.

For information about mounting a Sun StorEdge SAM-FS file system, see the `mount_samfs(1M)` man page.

10. Issue the `cd(1)` command to change to the mount point of the file system.
11. Use the `samfsrestore(1M)` command, or use File System Manager, to restore each file.

Restore from the dump file you had or from the dump file created in Step 1.

For more information, see the `samfsdump(1M)` man page, the File System Manager online help, or the *Sun StorEdge SAM-FS Troubleshooting Guide*.
12. Use the `restore.sh(1M)` script to stage back all files that had been online:

```
# restore.sh log-file mount-point
```

For *log-file*, specify the name of the log file that was created by the `sammkfs(1M)` or the `samfsrestore(1M)` commands.

For *mount-point*, specify the mount point of the file system being restored.

For information about the `restore.sh(1M)` script, see the `restore.sh(1M)` man page.

Administering File System Quotas

This chapter describes how to enable and administer file system quotas. This chapter contains the following sections:

- “Overview” on page 45
 - “Enabling Quotas” on page 48
 - “Checking Quotas” on page 57
 - “Changing and Removing Quotas” on page 59
-

Overview

File system quotas control the amounts of online and total disk space that can be consumed by a specific user, by a group of users, or by a site-determined group of users called an admin set.

Quotas help control the size of a file system by limiting the amount of space and the number of inodes that each user can consume. Quotas can be especially useful on file systems that contain user home directories. After quotas are enabled, you can monitor usage and adjust the quotas as needs change.

A file system provides a user with blocks for data and inodes for files. Each file uses one inode, and file data is stored in a disk allocation unit (DAU). DAU sizes are determined at the time the file system is created. Quotas account for disk usage in multiples of 512 bytes.

The following subsections provide background information about using quotas:

- “Types of Quotas, Quota Files, and Quota Records” on page 46
- “Soft Limits and Hard Limits” on page 47
- “Disk Blocks and File Limits” on page 47

Types of Quotas, Quota Files, and Quota Records

You can set quotas according to user ID, group ID, or an administrator's site-specific grouping. This site-specific grouping is called an *admin set ID*. You can use an admin set ID, for example, to identify a collection of users working on a project for which file system quotas are imposed.

Quotas are enabled when the `quota` mount option is in effect and the system detects the presence of one or more quota files in the file system's root directory. The `quota` mount option is enabled by default. If you mount the file system with `noquota` in effect, quotas are disabled. For more information about mount options, see the `mount_samfs(1M)` man page.

Each quota file contains a sequence of records. Record zero is the record for the system administrator's quotas and resource usage. System administrator quotas are never enforced, but you can use any record, including the system administrator's record, as a template for subsequent records in the quota file. For more information about this practice, see "To Enable or Change Limits for Users, Groups, or Admin Sets Using an Existing Quota File" on page 55.

Record one is the record in the quota file for user one, group one, or admin set ID one, depending on the type of quota file. You can edit record one and all subsequent records in order to set different quotas for different users. TABLE 4-1 shows the quota file names and the quotas they enable in `/root`.

TABLE 4-1 Quota File Names

Quota File Name	Quota Type
<code>.quota_u</code>	UID (system user ID)
<code>.quota_g</code>	GID (system group ID)
<code>.quota_a</code>	AID (system admin set ID)

You can set default quota limits for users by editing record zero in the quota file and allowing the values in record zero to be used as the initial quota settings for all other users. By default, if user quota limits have not been set specifically, the system uses the values in record zero.

Each quota file requires 128 bytes of space. To calculate the necessary size for the initial zero quota file, use the following formula:

$$\begin{aligned} (\text{highest-ID} + 1) \times 128 &= x \\ x / 4096 &= \text{zero quota file size} \end{aligned}$$

Soft Limits and Hard Limits

You can set both soft and hard limits. A hard limit specifies a fixed amount of system resources available for use, which the system never allows a user to exceed. A soft limit specifies a level of system resource use that can be exceeded temporarily, up to the hard limit. The soft limit is never larger than the hard limit.

If a user attempts to allocate resources beyond the hard limit, the operation is aborted. In this case, the operation fails and generates an `EDQUOT` error.

After a user exceeds a soft limit, a timer starts, and the user enters a grace period. While the timer is ticking, the user is allowed to operate above the soft limit. After the user goes below the soft limit, the timer is reset. If the grace period ends and the timer stops without the user's having gone below the soft limit, the soft limit is then enforced as a hard limit.

For example, assume that a user has a soft limit of 10,000 blocks and a hard limit of 12,000 blocks. If the user's block usage exceeds 10,000 blocks and the timer exceeds the grace period, this user is no longer able to allocate more disk blocks on that file system until usage drops below the 10,000-block soft limit.

You, the administrator, can use the `samquota(1M)` command to see the timer value. The `squota(1)` command is a user version of the `samquota(1M)` command. The `squota(1)` user command contains options that users can specify to obtain information about quotas that pertain to them.

Disk Blocks and File Limits

It is possible for a user to exceed an inode quota, without using any blocks, by creating all empty files. It is also possible for a user to use only one inode and still exceed the block quota by creating a file that is large enough to consume all data blocks in the user's quota.

File system quotas are expressed in terms of the number of 512-byte blocks that a user can allocate. However, disk space is allocated to user files in terms of DAUs. The DAU setting is specified by the `-a allocation_unit` option to the `sammkfs(1M)` command. It is preferable to set a block quota to a multiple of the file system DAU. If this is not done, users can allocate only up to the block count, rounded down to the nearest DAU. See "Enabling Default Quota Values" on page 54 for instructions on setting block quotas.

Enabling Quotas

You can enable quotas through a process that includes editing system files, creating quota files, and entering various quota commands.

The following subsections provide more details on how to configure a file system to use quotas and how to enable quotas.

Guidelines for Setting Up Quotas

Before you enable quotas, you should determine how much disk space and how many inodes to allocate to each user. If you want to be sure that the total file system space is never exceeded, you can divide the total size of the file system by the number of users. For example, if three users share a 100-megabyte slice and have equal disk space needs, you could allocate 33 megabytes to each. In environments in which not all users are likely to reach their limits, you might want to set individual quotas so that they add up to more than the total size of the file system. For example, if three users shared a 100-megabyte slice, you could allocate 40 megabytes to each.

You can use the following quota commands, in the formats shown, for displaying quota information:

- The `squota(1)` command is for end users. It enables them to retrieve quota information for themselves on a user, group, or admin set basis.
- The `samquota(1M)` command is for system administrators. It enables you to retrieve quota information or to set quotas. Use the `-U`, `-G`, and `-A` options for a user, a group, or an admin set, respectively. CODE EXAMPLE 4-1 shows this.

CODE EXAMPLE 4-1 Using `samquota(1M)` to Retrieve Information

#	samquota	-U janet	<i>/mount-point</i>	<i>#Prints a user quota</i>
#	samquota	-G pubs	<i>/mount-point</i>	<i>#Prints a group quota</i>
#	samquota	-A 99	<i>/mount-point</i>	<i>#Prints an admin set quota</i>

▼ To Configure a New File System to Use Quotas

Use this procedure if you are creating a new file system and no files currently reside in the file system. To configure an existing file system to use quotas, see “To Configure an Existing File System to Use Quotas” on page 50.

Before you start this procedure, make sure that you do not have the `noquota` mount option specified in your `samfs.cmd` or `/etc/vfstab` files.

1. Become superuser.

2. Create the file system.

To create the file system, follow the steps outlined in the *Sun StorEdge SAM-FS Installation and Upgrade Guide*.

3. Use the `mount(1M)` command to mount the file system.

For example:

```
# mount /qfs1
```

4. Use the `dd(1M)` command to create the quota files.

The arguments to this command depend on the type of quota you are creating, as follows:

- To create admin set quotas, use the following command:

```
# dd if=/dev/zero of=/qfs1/.quota_a bs=4096 count=1
```

- To create group quotas, use the following command:

```
# dd if=/dev/zero of=/qfs1/.quota_g bs=4096 count=1
```

- To create user quotas, use the following command:

```
# dd if=/dev/zero of=/qfs1/.quota_u bs=4096 count=1
```

For more information about the `dd(1M)` command, see the `dd(1M)` man page.

5. Use the `umount(1M)` command to unmount the file system in which the quota files have been created.

For example:

```
# umount /qfs1
```

The file system must be unmounted so it can be remounted and have its quota files read at mount time. For more information about the `umount(1M)` command, see the `umount(1M)` man page.

6. Use the `samfsck(1M)` command to perform a file system check.

In the following example, the `-F` option resets the in-use values in the quota files:

```
# samfsck -F qfs1
```

7. Use the `mount(1M)` command to remount the file system.

The system enables quotas when it detects the presence of one or more quota files in the root directory.

Note – You do not need to include the `quota` mount option in the `/etc/vfstab` or `samfs.cmd` file. The `quota` mount option is enabled by default with the `mount(1M)` command, and quotas are enabled automatically when the system detects the presence of quota files.

For more information about the `mount(1M)` command, see the `mount_samfs(1M)` man page.

8. Use the `samquota(1M)` command to set quotas for users, groups, or admin sets.

Subsequent sections in this chapter provide procedures and show examples of this process. For more information about the `samquota(1M)` command, see the `samquota(1M)` man page.

▼ To Configure an Existing File System to Use Quotas

Use this procedure if you are creating quotas for a file system that is already populated with files. If you are configuring a new file system to use quotas, see “To Configure a New File System to Use Quotas” on page 48.

Before you start this procedure, make sure that you do not have the `noquota` mount option specified in your `samfs.cmd` or `/etc/vfstab` files.

1. Use the `su(1)` command to become superuser.

2. Use the `mount(1M)` command to examine the `/etc/mnttab` file and ensure that the file system is mounted:

```
# mount
```

Make sure that the file system is listed in the mount list that is displayed.

3. Use the `cd(1)` command to change to the root directory of the file system for which quotas are to be enabled.

For example:

```
# cd /oldfs1
```

4. Use the `ls(1) -a` command to retrieve the list of files in this directory and verify that quotas do not already exist on the file system.

If any of the following files are present, quotas have been enabled for this file system: `.quota_u`, `.quota_g`, `.quota_a`.

If any quota type is established for a file system, you can establish any other quota type later. Be careful not to modify existing quota files when adding new ones.

5. If the quota files do not exist for the types of quotas you wish to enforce, use the `dd(1M)` command to create the quota files.

Determine the highest existing ID numbers of the types of quotas you wish to enforce. Make the initial, zero, quota files large enough to hold the records for those IDs; each quota file record requires 128 bytes.

For example, if you want to enable admin set quotas, and the highest admin set ID in use on the file system is 1024, the calculation is as follows:

$$(1024 + 1) \times 128 = 131200$$

$$131200/4096 = 32.031...$$

Use the following command:

```
# dd if=/dev/zero of=/oldfs1/.quota_a bs=4096 count=33
```

For more information about the `dd(1M)` command, see the `dd(1M)` man page.

6. Use the `umount(1M)` command to unmount the file system in which the quota files have been created.

For example:

```
# umount /oldfs1
```

The file system must be unmounted so it can be remounted and have its quota files read at mount time. For more information about unmounting a file system, see “Unmounting a File System” on page 38.

7. Use the `samfsck(1M) -F` command to perform a file system check.

This command updates records allocated in the quota files with correct, current usage information.

For example:

```
# samfsck -F /oldfs1
```

8. Use the `mount(1M)` command to remount the file system in which the quota files have been created.

The system enables quotas when it detects the presence of one or more quota files in the `/root` directory.

You do not need to include the `quota` mount option in the `/etc/vfstab` or `samfs.cmd` file. The `quota` mount option is enabled by default with the `mount(1M)` command, and quotas are enabled automatically when the system detects the presence of quota files.

Note – If quota files are present and if the file system is mounted with quotas disabled, the quota records become inconsistent with actual usages when blocks or files are allocated or freed. If a file system with quotas is mounted and run with quotas disabled, run the `samfsck(1M) -F` command to update the quota file usage counts before again remounting the file system with quotas enabled.

For more information about the `mount(1M)` command, see the `mount_samfs(1M)` man page.

9. Use the `samquota(1M)` command to set quotas for users, groups, or admin sets.

Subsequent sections in this chapter provide procedures and show examples of this process. For more information about the `samquota(1M)` command, see the `samquota(1M)` man page.

▼ To Assign Admin Set IDs to Directories and Files

1. Use the `su(1)` command to become superuser.

2. Set the admin IDs.

Use the `samchaid(1M)` command to change the admin set IDs for the directory or file, as follows:

- To set IDs for a file or directory, specify the directory name or path. For example:

```
# samchaid 100 admin.dir
```

- To set IDs for a directory tree, use the -R and (if necessary) the -h options. The -R option specifies a recursive operation, and the -h option changes links, not targets. For example:

```
# samchaid -R -h 22 /qfs1/joe /qfs1/nancee
```

For more information about the `samchaid(1M)` command, see the `samchaid(1M)` man page.

Setting Infinite Quotas

An infinite quota is a kind of special quota. Users with infinite quotas are never denied access to any available file system resource. You can set infinite quota values into record zero of the user, group, or admin set ID quota files and then use this record as the default value for a new user, group, or admin set ID.

▼ To Set an Infinite Quota

- Use the `samquota(1M)` command to set the quota limit to zero.

For example:

```
# samquota -U fred -b 0:h -f 0:h /qfs1
```

You can use the `samquota(1M)` command to set infinite quotas for particular users, groups, or admin set IDs by setting zero values for all hard and soft limits. CODE EXAMPLE 4-2 shows how to set infinite quotas.

CODE EXAMPLE 4-2 Setting Infinite Quotas

```
# samquota -G sam -b 0:s,h -f 0:s,h /sam6
# samquota -G sam /sam6
```

		Online Limits		Total Limits				
Type	ID	In Use	Soft	Hard	In Use	Soft	Hard	
/sam6								
Files	group	101	339	0	0	339	0	0

CODE EXAMPLE 4-2 Setting Infinite Quotas (Continued)

Blocks group	101	248	0	0	2614	0	0
Grace period			0s			0s	
---> Infinite quotas in effect.							

Enabling Default Quota Values

You can use the `samquota(1M)` command to enable a default quota for a user, group, or admin set. This is accomplished through default limits in user, group, or admin set zero.

▼ To Enable Default Quota Values for Users, Groups, or Admin Sets

- Use the `samquota(1M)` command.

For example, the following `samquota(1M)` command sets default quotas for all admin set IDs:

```
# samquota -A 0 -b 12000:s -b 15000:h -b 12G:s:t -b 15G:h:t \  
-f 1000:s -f 1200:h -t 1w /qfs1
```

The preceding command sets any user's uninitialized admin set quota limits as follows:

- The soft online block limit (`-b limit:s`) is set to 12,000 blocks.
- The hard online block limit (`-b limit:h`) is set to 15,000 blocks.
- The total soft block limit (`-b limit:s:t`) is set to 12 gigablocks.
- The total hard block limit (`-b limit:h:t`) is set to 15 gigablocks.
- The soft file limit (`-f limit:s`) is set to 1000 files.
- The hard file limit (`-f limit:h`) is set to 1200 files.
- The grace period (`-t limit`) is set to one week.

Note – If a quota record already exists, the existing values remain in effect. This occurs, for example, if the admin group already has blocks assigned to it.

You can set similar default quotas for users or groups by specifying `-U 0` or `-G 0`, respectively, in place of `-A 0`.

For more information about the `samquota(1M)` command, see the `samquota(1M)` man page.

Enabling Limits

You can use the `samquota(1M)` command to enable a set of limits for a particular user, group, or admin set.

▼ To Enable Limits for Users, Groups, or Admin Sets

- Use the `samquota(1M)` command.

CODE EXAMPLE 4-3 shows commands that enable limits for users, groups, and admin sets, respectively.

CODE EXAMPLE 4-3 Quota Commands

```
# samquota -U joe -b 15000:s -b 20000:h -b 12G:s:t -b 15G:h:t \  
-f 500:s -f 750:h -t 3d /qfs1  
# samquota -G proj -b 15000:s -b 20000:h -b 12G:s:t -b 15G:h:t \  
-f 500:s -f 750:h -t 3d /qfs1  
# samquota -A 7 -b 15000:s -b 20000:h -b 12G:s:t -b 15G:h:t \  
-f 500:s -f 750:h -t 3d /qfs1
```

For more information about the `samquota(1M)` command, see the `samquota(1M)` man page.

▼ To Enable or Change Limits for Users, Groups, or Admin Sets Using an Existing Quota File

After quotas are established, you can use an existing quota file as a template when creating limits for another user, group, or admin set. The following procedure shows this. You can also use this procedure to change any of the quota settings.

1. Use the `samquota(1M)` command to retrieve a quota file, and direct the output to a temporary file.

Use the `-e` option with one or more of the following additional options: `-U userID`, `-G groupID`, or `-A adminsetID`.

CODE EXAMPLE 4-4 shows how to create and retrieve file `quota.group` to use as a template.

Note – You can use a group quota entry as a template to create a user quota entry.

CODE EXAMPLE 4-4 File `quota.group`

```
# samquota -G sam -e /sam6 > /tmp/quota.group  
# cat /tmp/quota.group  
  
# Type ID
```

CODE EXAMPLE 4-4 File quota.group (Continued)

```
# Online Limits Total Limits
# soft hard soft hard
# Files
# Blocks
# Grace Periods
#
samquota -G 101 \
-f 200:s:o -f 300:h:o -f 200:s:t -f 300:h:t \
-b 40000:s:o -b 60000:h:o -b 40000000:s:t -b 60000000:h:t \
-t 0s:o -t 0s:t /sam6
```

2. Use an editor to edit the temporary file you just created.

CODE EXAMPLE 4-5 shows the file that was generated in Step 1 opened in the vi(1) editor. Group ID 101 has been changed to 102. This has the effect of generating a command to copy the quotas set for group 101 to group 102.

CODE EXAMPLE 4-5 File quota.group After Editing

```
# Type ID
# Online Limits Total Limits
# soft hard soft hard
# Files
# Blocks
# Grace Periods
#
samquota -G 102 \
-f 200:s:o -f 300:h:o -f 200:s:t -f 300:h:t \
-b 40000:s:o -b 60000:h:o -b 40000000:s:t -b 60000000:h:t \
-t 1d:o -t 1d:t /sam6
```

3. Save the file and exit the editor.

4. To apply the changes made in the editor, execute the file using the shell.

For example:

```
# sh -x /tmp/quota.group
```

In this example, the -x option directs the shell to echo the commands it executes. You can omit the -x option if desired.

Checking Quotas

After you have enabled disk and inode quotas, you can check these quotas. The `samquota(1M)` command is an administrator command that generates a quota report on an individual user, group, or admin set. The `squota(1)` command is a user command that enables users to check their own individual quotas.

▼ To Check for Exceeded Quotas

- 1. **Become superuser.**
- 2. **Use the `samquota(1M)` command to display the quotas in effect for mounted file systems.**
 - To display user quotas, specify the following command:

```
# samquota -U userID [ file ]
```

For *userID*, specify the numeric user ID or user name of the user whose quotas are being examined.

For *file*, specify a specific file system for the selected user, group, or admin set. The *file* argument can also be the name of any file in the file system. Typically, *file* is the name of the root directory of the file system.

Example 1. CODE EXAMPLE 4-6 retrieves user `hm1259`'s quota statistics in the `sam6` file system on the server and displays output indicating that this user is not exceeding his quota.

CODE EXAMPLE 4-6 Checking for Exceeded Quotas for User `hm1259`

# samquota -U hm1259 /sam6								
		Online Limits			Total Limits			
	Type	ID	In Use	Soft	Hard	In Use	Soft	Hard
/sam6								
Files	user	130959	13	100	200	13	100	200
Blocks	user	130959	152	200	3000	272	1000	3000
Grace period				0s			0s	

Example 2. CODE EXAMPLE 4-7 retrieves user `memil`'s quota statistics in all mounted Sun StorEdge QFS file systems and displays output indicating that this user is exceeding the quota. Note the plus sign (+) in the `Blocks` row of the output. The

plus sign would appear in the Files row, too, if the soft quota limit were being exceeded for files.

CODE EXAMPLE 4-7 Checking for Exceeded Quotas for User `memil`

# samquota -U memil								
	Type	ID	In Use	Online Limits		In Use	Total Limits	
				Soft	Hard		Soft	Hard
/sam6								
Files	user	130967	4	500	750	4	500	750
Blocks	user	130967	41016+	40000	50000	41016	50000	50000
Grace period				1w			0s	
---> Warning: online soft limits to be enforced in 6d23h36m45s								
/sam7								
Files	user	130967	4	500	750	4	500	750
Blocks	user	130967	4106	40000	50000	4106	50000	50000
Grace period				1w			0s	

If a hard limit has been exceeded, or if the soft limit has been exceeded and the grace period has expired, the offending In Use field is marked with an asterisk character (*). If a quota record’s limits are determined to be inconsistent (for example, if a soft limit is larger than a hard limit), an exclamation point is used to mark the field, and all allocation operations are prevented.

- To display group quotas, specify the following command:

```
# samquota -G groupID [ file ]
```

For *groupID*, specify the numeric group ID or the group name for the group of users whose quotas are being examined. For *file*, specify a specific file system for the selected group. The *file* argument can also be the name of any file in the file system. Typically, *file* is the name of the root directory of the file system.

For example, the following command retrieves user quota statistics for the group turtles in the qfs3 file system:

```
# samquota -G turtles /qfs3
```

- To display admin set quotas, specify the following command:

```
# samquota -A adminsetID [ file ]
```

For *adminsetID*, specify the numeric admin set ID of the site-specific administrator set whose quotas are being examined. For *file*, specify a specific file system for the selected admin set. The *file* argument can also be the name of any file in the file system. Typically, *file* is the name of the root directory of the file system.

For example, the following command retrieves user quota statistics for the admin set 457 in all mounted Sun StorEdge QFS file systems:

```
# samquota -A 457
```

Changing and Removing Quotas

You can change quotas to adjust the amount of disk space or number of inodes allocated to users. You can also remove quotas from users or from an entire file system. The following subsections describe how to change and remove quotas:

- “To Change the Grace Period” on page 59
- “Changing the Grace Period Expiration” on page 61
- “To Inhibit Additional File System Resource Allocations” on page 63
- “To Remove a File System’s Quotas” on page 65
- “To Correct Quotas” on page 66

▼ To Change the Grace Period

You can use the `samquota(1M)` command to change the soft time limit grace period.

1. Use the `samquota(1M)` command to retrieve quota statistics for a user, group, or admin set.

See “To Check for Exceeded Quotas” on page 57 for instructions.

Example. CODE EXAMPLE 4-8 retrieves information about group `sam` and shows that this group is over its soft limit.

CODE EXAMPLE 4-8 Exceeding a Soft Limit

# samquota -G sam /sam6								
		ID	In Use	Online Limits		In Use	Total Limits	
Type				Soft	Hard		Soft	Hard
/sam6								
Files	group	101	32	2000	2000	32	2000	2000

CODE EXAMPLE 4-8 Exceeding a Soft Limit (Continued)

```
Blocks group 101      41888*  40000 60000000      43208 60000000 60000000
Grace period                1w              1w
---> Online soft limits under enforcement (since 30s ago)
```

2. Examine the output from the `samquota(1M)` command and determine what the new limits should be.
3. Use the `samquota(1M)` command to change the soft time limit grace period.

CODE EXAMPLE 4-9 shows using the `samquota(1M)` command options to use.

CODE EXAMPLE 4-9 Using `samquota(1M)` to Change Soft Time Limit Grace Periods

```
# samquota -U userID -t interval file
# samquota -G groupID -t interval file
# samquota -A adminID -t interval file
```

The arguments for these commands are as follows:

- *userID* is the numeric user ID or user name of the user whose quotas are being changed.
- *groupID* is the numeric group ID or the group name for the group of users whose quotas are being changed.
- *adminID* is the numeric admin set ID of the site-specific administrator set whose quotas are being changed.
- *interval* is the interval to be used for the grace period. Specify an integer number for interval to indicate the quantity, and then specify a unit multiplier, if desired. The default unit multiplier is s, which indicates seconds. You can also specify w (for weeks), d (for days), h (for hours), or m (for minutes).
- *file* is the specific file system for the selected user, group, or admin set. The file argument can also be the name of any file in the file system. Typically, file is the name of the root directory of the file system

For example, suppose that you wanted to change the grace period for user `mem11`. CODE EXAMPLE 4-10 shows the `samquota(1M)` command that you would use to verify the quotas and its output.

CODE EXAMPLE 4-10 Changing the Grace Period

```
# samquota -U memil /sam6
```

	Type	ID	In Use	Online Limits		In Use	Total Limits	
				Soft	Hard		Soft	Hard
/sam6								
Files	user	130967	4	500	750	4	500	750
Blocks	user	130967	41016+	40000	50000	41016	50000	50000
Grace period				3d			0s	
---> Warning: online soft limits to be enforced in 2d23h59m7s								

You would enter the following command to shorten the grace period.

```
# samquota -U memil -t 1d /sam6
```

CODE EXAMPLE 4-11 shows the `samquota(1M)` command that you would then use to verify the new quotas.

CODE EXAMPLE 4-11 Verifying the New Quotas

```
# samquota -U memil /sam6
```

	Type	ID	In Use	Online Limits		In Use	Total Limits	
				Soft	Hard		Soft	Hard
/sam6								
Files	user	130967	4	500	750	4	500	750
Blocks	user	130967	41016+	40000	50000	41016	50000	50000
Grace period				1d			0s	
---> Warning: online soft limits to be enforced in 23h58m31s								

Changing the Grace Period Expiration

If a user has exceeded the soft quota limit, changing the grace period itself does not modify the expiration timer of any grace periods that have already started. If the grace period is already in effect, you can use the `samquota(1M)` command to modify the grace period in one of the following ways:

- **Clear the grace period timer** – The next time the user allocates a file or block while still over a soft limit, the grace period timer is reset, and the grace period restarts.

CODE EXAMPLE 4-12 shows the command used to clear the timer so it starts counting the next time a user in group `sam` attempts to allocate a block or file in `/sam6`.

CODE EXAMPLE 4-12 Clearing the Timer

```
# samquota -G sam -x clear /sam6
Setting Grace Timer: continue? y
# samquota -G sam /sam6
```

	Type	ID	In Use	Online Limits		In Use	Total Limits	
				Soft	Hard		Soft	Hard
/sam6								
Files	group	101	32	2000	2000	32	2000	2000
Blocks	group	101	41888+	40000	60000000	43208	60000000	60000000
Grace period				1w			1w	

---> Warning: online soft limits to be enforced in 6d23h59m56s

- **Reset the grace period timer** – When an expiration period is reset, the timer is reset, and the grace period restarts.

CODE EXAMPLE 4-13 resets the grace period.

CODE EXAMPLE 4-13 Resetting the Grace Period Timer

```
# samquota -G sam -x reset /sam6
Setting Grace Timer: continue? y
# samquota -G sam /sam6
```

	Type	ID	In Use	Online Limits		In Use	Total Limits	
				Soft	Hard		Soft	Hard
/sam6								
Files	group	101	32	2000	2000	32	2000	2000
Blocks	group	101	41888	40000	60000000	43208	60000000	60000000
Grace period				1w			1w	

---> Warning: online soft limits to be enforced in 6d23h59m52s

- **Set the grace period to a value** – The timer is set to a value, and it starts counting down immediately from that value. There are no restrictions on this value. The value can be larger than the grace period.

CODE EXAMPLE 4-14 sets a very long expiration period.

CODE EXAMPLE 4-14 Setting a Very Long Grace Period

```
# samquota -G sam -x 52w /sam6
Setting Grace Timer: continue? y
# samquota -G sam /sam6
```

	Type	ID	In Use	Online Limits		In Use	Total Limits	
				Soft	Hard		Soft	Hard
/sam6								
Files	group	101	32	2000	2000	32	2000	2000
Blocks	group	101	41888	40000	60000000	43208	60000000	60000000
Grace period				52w			52w	

CODE EXAMPLE 4-14 Setting a Very Long Grace Period *(Continued)*

```
/sam6
Files group 101 32 2000 2000 32 2000 2000
Blocks group 101 41888+ 40000 60000000 43208 60000000 60000000
Grace period 1w 1w
---> Warning: online soft limits to be enforced in 51w6d23h59m54s
```

- **Expire the grace period timer** – The timer is set to expire immediately.
CODE EXAMPLE 4-15 expires the grace period.

CODE EXAMPLE 4-15 Expiring the Grace Period Timer

```
# samquota -G sam -x expire /sam6
Setting Grace Timer: continue? y
# samquota -G sam /sam6

Type ID In Use Online Limits In Use Total Limits
Soft Hard Soft Hard
/sam6
Files group 101 32 2000 2000 32 2000 2000
Blocks group 101 41888 40000 60000000 43208 60000000 60000000
Grace period 1w 1w
---> Online soft limits under enforcement (since 6s ago)
```

▼ **To Inhibit Additional File System Resource Allocations**

When the file system detects that quota values are not consistent for a user, group, or admin set, it prevents that user, group, or admin set from using any more system resources. You can inhibit file system resource allocations by creating inconsistent quota values. For example, you can inhibit further allocation if the hard block or file limits are lower than the soft block or file limits, or if a user’s soft limit is larger than the user’s hard limit.

The file system treats an inconsistent quota setting as a special quota. You can set inconsistent quota values into record zero of the user, group, or admin set ID quota files, and from there they can become the default values for new users, groups, or admin set IDs.

The following procedure shows how to inhibit further system resource allocations for a user, group, or admin set.

1. Become superuser.

2. Obtain, save, and examine current quota information.

CODE EXAMPLE 4-16 shows how to use the `samquota(1M)` command to retrieve current group quota information for group `sam` and write it to a backup file.

CODE EXAMPLE 4-16 Retrieving Group Quota Information

```
# samquota -G sam -e /sam6 | & tee restore.quota.sam

# Type   ID
#           Online Limits
#           soft      hard
# Files
# Blocks
# Grace Periods
#
samquota -G 101 \
  -f      2000:s:o -f      2000:h:o          -f      2000:s:t -f      2000:h:t \
  -b      40000:s:o -b 60000000:h:o          -b 60000000:s:t -b 60000000:h:t \
          -t      1w:o                      -t      1w:t \
  -x 51w6d23h59m:o                      -x clear      /sam6
```

To obtain quota information about a user quota, specify the `-U userID` option in place of the `-G` option. To obtain quota information about an admin set quota, specify the `-A adminID` option in place of the `-G` option.

3. Use the `samquota(1M)` command to set soft quotas to nonzero quotas and hard quotas to zero quotas.

The following command sets the quotas for group `sam` to be inconsistent:

```
# samquota -G sam -f 1:s -f 0:h -b 1:s -b 0:h /sam6
```

To make the quotas for users or admin sets inconsistent, specify the `-U userID` or `-A adminID` option in place of the `-G` option.

4. Use the `samquota(1M)` command to verify your changes.

CODE EXAMPLE 4-17 shows this.

CODE EXAMPLE 4-17 Verifying Changed Quotas

```
# samquota -G sam /sam6

          Type   ID   In Use   Online Limits   In Use   Total Limits
          /sam6   Soft   Hard   Soft   Hard
Files group  101    32!     1       0    32!     1       0
```


CODE EXAMPLE 4-17 Verifying Changed Quotas (Continued)

Blocks group	101	41888!	1	0	43208!	1	0
Grace period			1w			1w	
---> Quota values inconsistent; zero quotas in effect.							

In the preceding output, a zero quota is in effect. Note the exclamation point characters (!), which indicate the over-quota condition in the output.

- 5. Use the `sh(1)` and `samquota(1M)` commands to restore the group’s quota to what it was before the file/block allocation was inhibited and then to verify the changed quotas.

CODE EXAMPLE 4-18 shows these commands.

CODE EXAMPLE 4-18 Restoring the Group Quota

# sh restore.quota.sam								
Setting Grace Timer: continue? y								
Setting Grace Timer: continue? y								
# samquota -G sam /sam6								
			Online Limits			Total Limits		
	Type	ID	In Use	Soft	Hard	In Use	Soft	Hard
/sam6								
Files	group	101	32	2000	2000	32	2000	2000
Blocks	group	101	41888+	40000	60000000	43208	60000000	60000000
Grace period			1w		1w			
---> Warning: online soft limits to be enforced in 6d23h59m54s								

To perform this operation on a user quota, specify the `-U userID` option in place of the `-G` option. To perform this operation on an admin set quota, specify the `-A adminID` option in place of the `-G` option.

▼ To Remove a File System’s Quotas

To remove or disable quotas for a file system, you need to disable quotas in the mount process.

- 1. Use the `su(1)` command to become superuser.
- 2. (Optional) Use a text editor to add the `noquota` mount option to the `/etc/vfstab` or `samfs.cmd` file.

Alternatively, you can specify `noquota` as an option later, when you issue the mount command. See Step 4.

3. If the file system is mounted, use the `umount(1M)` command to unmount the file system.

For example:

```
# umount /myfs
```

If you have difficulty unmounting the file system, see “Unmounting a File System” on page 38.

4. Remount the file system using the `mount(1M)` command.

If you did not perform Step 2, include the `noquota` option with the `mount(1M)` command.

For example:

```
# mount -o noquota /myfs
```

5. Dispense with the quota files by doing one of the following:

- If you expect to reinstate the quota feature later and therefore do not want to destroy the quota files, unmount the file system, run the `samfsck(1M)` command with its `-F` option on the file system, and remount the file system again with the `noquota` mount option removed.
- If you do not expect to reinstate the quota feature at a later date, or if you want to reclaim the space consumed by the quota files, use the `rm(1)` command to remove the `.quota_u`, `.quota_g`, and `.quota_a` files. For example:

```
# rm /myfs/.quota_[agu]
```

▼ To Correct Quotas

1. Become superuser.
2. If the file system is mounted, use the `umount(1M)` command to unmount the file system.

For example:

```
# umount /myfs
```

If you have difficulty unmounting the file system, see “Unmounting a File System” on page 38.

3. Use the `samfsck(1M) -F` command to perform a file system check.

The `samfsck(1M)` command updates records allocated in the quota files with correct, current usage information.

For example:

```
# samfsck -F myfs
```

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

For example:

```
# mount /myfs
```


Advanced Topics

This chapter discusses advanced topics that are beyond the scope of basic system administration and usage. This chapter contains the following sections:

- “Using Daemons, Processes, and Tracing” on page 69
- “Using the `setfa(1)` Command to Set File Attributes” on page 73
- “Configuring WORM-FS File Systems” on page 75
- “Accommodating Large Files” on page 83
- “Understanding I/O Types” on page 84
- “Increasing File Transfer Performance for Large Files” on page 85
- “Setting the Write Throttle” on page 88
- “Setting the Flush-Behind Rate” on page 89
- “Tuning the Number of Inodes and the Inode Hash Table” on page 90

Using Daemons, Processes, and Tracing

It is useful to have an understanding of system daemons and processes when you are debugging. This section describes the Sun StorEdge SAM-FS daemons and processes. It also provides information about daemon tracing.

Daemons and Processes

All Sun StorEdge SAM-FS daemons are named in the form `sam-daemon_named`. Processes are named in a similar manner; the difference is that they do not end in the lowercase letter `d`.

TABLE 5-1 shows some of the daemons and processes that can run on your system. Others, such as `sam-genericd` and `sam-catserverd`, might also be running, depending on system activities.

TABLE 5-1 Daemons and Processes

Process	Description
<code>sam-archiverd</code>	Automatically archives Sun StorEdge SAM-FS files. This process runs as long as the Sun StorEdge SAM-FS file system is mounted.
<code>sam-fsd</code>	Master daemon.
<code>sam-rftd</code>	Transfers data between multiple Sun StorEdge SAM-FS host systems.
<code>sam-robotd</code>	Starts and monitors automated library media changer control daemons.
<code>sam-scannerd</code>	Monitors all manually mounted removable media devices. The scanner periodically checks each device for inserted archive media cartridges.
<code>sam-releaser</code>	Attempts to release disk space occupied by previously archived files on Sun StorEdge SAM-FS file systems until a low water mark is reached. The releaser is started automatically when a high water mark is reached on disk cache and stops when it has finished releasing files. This is a process, not a daemon.
<code>sam-stagealld</code>	Controls the associative staging of Sun StorEdge SAM-FS files.
<code>sam-stagerd</code>	Controls the staging of Sun StorEdge SAM-FS files.
<code>sam-rpcd</code>	Controls the remote procedure call (RPC) application programming interface (API) server process.

When you run Sun StorEdge SAM-FS software, `init` starts the `sam-fsd` daemon as part of `/etc/inittab` processing. The daemon is started at `init` levels 0, 2, 3, 4, 5, and 6. It should restart automatically in case of failure.

When running Sun StorEdge SAM-FS software, the `sam-fsd` daemon creates the following processes:

- `sam-archiverd`. The `sam-archiverd` daemon starts the `sam-arcopy` and the `sam-arfind` processes.
- `sam-catserverd`. Issuing a `samd(1M) stop` command stops this daemon.
- `sam-rftd`.
- `sam-initd`.
- `sam-robotd`. Issuing a `samd(1M) stop` command stops this daemon.
- `sam-scannerd`. Issuing a `samd(1M) stop` command stops this daemon.
- `sam-stagealld`.
- `sam-stagerd`.

Trace Files

Several Sun StorEdge SAM-FS processes can write messages to trace files. These messages contain information about the state and progress of the work performed by the daemons. The messages are primarily used by Sun Microsystems staff members to improve performance and diagnose problems. The message content and format are subject to change from release to release.

Trace files can be used in debugging. By default, trace files are not enabled. You can enable trace files by editing the `defaults.conf` file. You can enable tracing for all processes, or you can enable tracing for individual processes. For information about the processes that you can trace, see the `defaults.conf(4)` man page.

By default, trace files are written to the `/var/opt/SUNWsamfs/trace` directory. In that directory, the trace files are named for the processes (`archiver`, `catserver`, `fsd`, `ftpd`, `recycler`, `sharefsd`, and `stager`). You can change the names of the trace files by specifying directives in the `defaults.conf` configuration file. You can also set a limit on the size of a trace file and rotate your tracing logs. For information about controlling tracing, see the `defaults.conf(4)` man page.

Trace File Content

Trace file messages contain the time and source of the message. The messages are produced by events in the processes. You can select the events by using directives in the `defaults.conf` file.

The default events are as follows:

- Customer notification `syslog` or `notify` file messages
- Nonfatal program errors
- Fatal `syslog` messages
- Process initiation and completion
- Other miscellaneous events

You can also trace the following events:

- Memory allocations
- Interprocess communication
- File actions
- Operator messages
- Queue contents when changed
- Other miscellaneous events

The default message elements (program name, process id (PID), and time) are always included and cannot be excluded. Optionally, the messages can also contain the following elements:

- The date (The time is always included.)
- The source file name and line number
- The event type

Trace File Rotation

To prevent trace files from growing indefinitely, the `sam-fsd` daemon monitors the size of the trace files and periodically executes the following command:

```
/opt/SUNWsamfs/sbin/trace_rotate
```

This script moves the trace files to sequentially numbered copies. You can modify this script to suit your operation. Alternatively, you can provide this function using `cron(1)` or some other facility.

Determining Which Processes Are Being Traced

To determine which processes are being traced currently, enter the `sam-fsd(1M)` command at the command line. CODE EXAMPLE 5-1 shows the output from this command.

CODE EXAMPLE 5-1 `sam-fsd(1M)` Command Output

```
# sam-fsd
Trace file controls:
sam-amld      /var/opt/SUNWsamfs/trace/sam-amld
               cust err fatal misc proc date
               size    0    age 0
sam-archiverd /var/opt/SUNWsamfs/trace/sam-archiverd
               cust err fatal misc proc date
               size    0    age 0
sam-catserverd /var/opt/SUNWsamfs/trace/sam-catserverd
               cust err fatal misc proc date
               size    0    age 0
sam-fsd       /var/opt/SUNWsamfs/trace/sam-fsd
               cust err fatal misc proc date
               size    0    age 0
sam-rftd      /var/opt/SUNWsamfs/trace/sam-rftd
               cust err fatal misc proc date
               size    0    age 0
```


CODE EXAMPLE 5-1 `sam-fsd(1M)` Command Output (*Continued*)

<code>sam-recycler</code>	<code>/var/opt/SUNWsamfs/trace/sam-recycler</code> cust err fatal misc proc date size 0 age 0
<code>sam-sharefsd</code>	<code>/var/opt/SUNWsamfs/trace/sam-sharefsd</code> cust err fatal misc proc date size 0 age 0
<code>sam-stagerd</code>	<code>/var/opt/SUNWsamfs/trace/sam-stagerd</code> cust err fatal misc proc date size 0 age 0
<code>sam-serverd</code>	<code>/var/opt/SUNWsamfs/trace/sam-serverd</code> cust err fatal misc proc date size 0 age 0
<code>sam-clientd</code>	<code>/var/opt/SUNWsamfs/trace/sam-clientd</code> cust err fatal misc proc date size 0 age 0
<code>sam-mgmt</code>	<code>/var/opt/SUNWsamfs/trace/sam-mgmt</code> cust err fatal misc proc date size 0 age 0

For more information about enabling trace files, see the `defaults.conf(4)` man page and the `sam-fsd(1M)` man page.

Using the `setfa(1)` Command to Set File Attributes

Sun StorEdge SAM-FS file systems enable end users to set performance attributes for files and directories. Applications can enable these performance features on a per-file or per-directory basis. The following sections describe how the application programmer can use these features to select file attributes for files and directories, to preallocate file space, to specify the allocation method for the file, and to specify the disk stripe width.

For more information about implementing the features described in the following subsections, see the `setfa(1)` man page.

Selecting File Attributes for Files and Directories

The `setfa(1)` command sets attributes on a new or existing file. The file is created if it does not already exist.

You can set attributes on a directory as well as a file. When using `setfa(1)` with a directory, files and directories created within that directory inherit the attributes set in the original directory. To reset attributes on a file or directory to the default, use the `-d` (default) option. When the `-d` option is used, first attributes are reset to the default and then other attributes are processed.

Preallocating File Space

An end user can preallocate space for a file. This space is associated with a file so that no other files in the file system can use the disk addresses allocated to this file. Preallocation ensures that space is available for a given file, which avoids a file-system-full condition. Preallocation is assigned at the time of the request rather than when the data is actually written to disk.

Note that space can be wasted by preallocation of files. If the file size is less than the allocation amount, the kernel allocates space to the file from the current file size up to the allocation amount. When the file is closed, space below the allocation amount is not freed.

You can preallocate space for a file by using the `setfa(1)` command with either the `-L` or the `-l` (lowercase letter L) option. Both options accept a file length as their argument. Use the `-L` option for an existing file, which can be empty or contain data. Use the `-l` option for a file that has no data yet. If you use the `-l` option, the file cannot grow beyond its preallocated limit.

For example, to preallocate a 1-gigabyte file named `/qfs/file_alloc`, type the following:

```
# setfa -l 1g /qfs/file_alloc
```

After space for a file has been preallocated, truncating a file to 0 length or removing the file returns all space allocated for a file. There is no way to return only part of a file's preallocated space to the file system. In addition, if a file is preallocated in this manner, there is no way to extend the file beyond its preallocated size in future operations.

Selecting a File Allocation Method and Stripe Width

By default, a file uses the allocation method and stripe width specified at mount time (see the `mount_samfs(1M)` man page). However, an end user might want to use a different allocation scheme for a file or directory. The user could do this by using the `setfa(1)` command with the `-s` (stripe) option.

The allocation method can be either round-robin or striped. The `-s` option specifies the allocation method and the stripe width, as shown in TABLE 5-2.

TABLE 5-2 File Allocations and Stripe Widths

-s Option	Allocation Method	Stripe Width	Explanation
0	Round-robin	Not applicable	The file is allocated on one device until that device has no space.
1-255	Striped	1-255 DAUs	The file is striped across all disk devices with this number of DAUs per disk.

The following example shows how to create a file explicitly by specifying a round-robin allocation method:

```
# setfa -s 0 /qfs/100MB.rrabin
```

The following example shows how to create a file explicitly by specifying a striped allocation method with a stripe width of 64 DAUs (preallocation is not used):

```
# setfa -s 64 /qfs/file.stripe
```



Configuring WORM-FS File Systems

Write Once Read Many (WORM) technology is used in many applications because of the integrity of the data and the accepted legal admissibility of stored files that use the technology. Beginning with release 4, update 3, of the Sun StorEdge SAM-FS software, a WORM-FS feature became available as an add-on package called `SUNWsamfswm`. In the 4U4 software release the WORM-FS interface has been modified to be compatible with the new Sun StorEdge 5310 NAS appliance. The

existing WORM-FS interface using `ssum` will no longer be supported. Previous enhancements to the `sfs` and `sfind` utilities supporting the WORM-FS interface remain unchanged.

Note – The WORM-FS feature is licensed separately from the Sun StorEdge SAM-FS file system. Contact your local Sun sales representative for information about obtaining the WORM-FS package.

The WORM-FS feature offers default and customizable file-retention periods, data and path immutability, and subdirectory inheritance of the WORM setting.

Enabling the WORM-FS Feature

Use the `worm_capable` mount option to enable the WORM-FS feature. This option can be placed in `/etc/vfstab` or typed on the command line when the file system is mounted. The `worm_capable` attribute is stored in the mount table and enables WORM files to be created in directories anywhere in the file system.

Note – You must have system administration privileges to set the `worm_capable` mount option in `/etc/vfstab`.

shows the two WORM-FS mount options. The file system `samfs1` mounted at `/samfs1` is WORM-capable and has the default retention period for files set to 60 minutes.

CODE EXAMPLE 5-2 Using WORM-FS Mount Options

```
# cat /etc/vfstab
#device    device  mount  FS    fsck    mount  mount
#to mount  to fsck  point  type  pass    at boot options
#
fd         -       /dev/fdfd -      no      -
/proc      -       /proc  proc  -       no      -
/dev/dsk/c0t0d0s1 - -      swap  -       no      -
samfs1     -       /samfs1samfs- yes     worm_capable,def_retention=60
swap       -       /tmp   tmpfs- yes     -
```

After the WORM-FS feature has been enabled, the file system's superblock is updated to reflect the WORM capability. Any subsequent attempt to rebuild the file system through `sammkfs` will fail.

The `worm_capable` mount option enables a file system to contain WORM files, but it does not automatically create WORM files. To create a WORM file, you must first make the directory WORM-capable. To do this, create an ordinary directory and then use the WORM trigger command `chmod 4000 directory-name` to set the WORM bit on the directory. The directory can now contain WORM files.

After setting the WORM bit on a parent directory, you can create files in that directory and then use the WORM trigger `chmod 4000 file-name` to set the WORM bit on files that you want to be retained.

Note – Use care when applying the WORM trigger. The file data and path cannot be changed after the file has the WORM feature applied. Once this feature is applied to a file, it is irrevocable.

The WORM-FS feature also includes file-retention periods that can be customized. Assigning a retention period to a file maintains the WORM features in that file for the specified period of time. Do one of the following to set a retention period for a file:

- Advance the file's access time using the `touch` utility, or with a program using the `libc` subroutine `utimes()`. With these methods, the file's retention period is stored in minutes. (After the access time is advanced, the `chmod 4000` command is used to set the WORM trigger.)
- Use the `def_retention` mount option to set a default retention period.
- Use the system-wide default retention period. This is set to 30 days by default, but can be set to any value through the mount option.

shows the creation of a file in a WORM-capable directory, setting of the WORM trigger on the file, and use of the `sls` command to display the file's WORM features. This example uses the default retention period of the file system (60 minutes, as set in).

CODE EXAMPLE 5-3 Creation of a WORM-Capable Directory and WORM File

```
# cd WORM
# echo "This is a test file" >> test
# sls -D
test:
    mode: -rw-r--r--  links: 1  owner: root group: other
    length: 20  admin id: 0  inode: 1027.1
    access: Oct 30 02:50  modification: Oct 30 02:50
    changed: Oct 30 02:50  attributes: Oct 30 02:50
    creation: Oct 30 02:50  residence: Oct 30 02:50

    checksum: gen  no_use  not_val  algo: 0

# chmod 4000 test
# sls -D
test:
    mode: -r--r--r--  links: 1  owner: root group: other
    length: 20  admin id: 0  inode: 1027.1
    access: Oct 30 02:50  modification: Oct 30 02:50
    changed: Oct 30 02:50  retention-end: Oct 30 2005 03:50
    creation: Oct 30 02:50  residence: Oct 30 02:50
    retention: active retention-period: 0y, 0d, 1h, 0m
    checksum: gen  no_use  not_val  algo: 0
```

With the addition of the WORM-FS feature, three states are possible for a file in a Sun StorEdge SAM-FS file system:

- Normal
- Retained
- Expired

The normal state represents the state of an ordinary file in a Sun StorEdge SAM-FS file system. A transition to the retained, or *active*, state occurs when the WORM bit is set on a file. The expired, or *over*, state occurs when the file's retention period is exceeded.

When a retention period is assigned to a file and the WORM trigger is applied to it, the file's path and data are immutable. When the retention period expires, the state is changed to "expired" but the path and data remain immutable.

When a file is in an expired state, only two operations are available:

- Extension of the retention period (The retention period can never be shortened.)

■ Deletion of the file

If the retention period is extended, the file's state returns to "active" and the new end date and duration are set accordingly.

Both hard and soft links to files can be used with the WORM-FS feature. Hard links can be established only with files that reside in a WORM-capable directory. After a hard link is created, it has the same WORM characteristics as the original file. Soft links can also be established, but a soft link cannot use the WORM features. Soft links to WORM files can be created in any directory in a Sun StorEdge SAM-FS file system.

Another attribute of the WORM-FS feature is directory inheritance. New directories that are created under a directory that includes the `worm_capable` attribute inherit this attribute from their parent. The WORM bit can be set on any file whose parent directory is WORM-capable. Ordinary users can set the WORM feature on directories and files that they own or have access to by using normal UNIX permissions.

Note – If a WORM-capable directory contains no WORM files, it can be deleted.

Setting a File's Retention Period Using `touch`

You use the `touch` utility to set or extend a file's retention period. To set the retention period, you must first advance the file's access time using `touch`, and then apply the WORM trigger using the `chmod` command.

CODE EXAMPLE 5-4 shows the use of the touch utility to set a file's retention period followed by the application of the WORM trigger.

CODE EXAMPLE 5-4 Using touch and chmod to Set the Retention Period

```
# touch -a -t200508181125 test
# sls -D
test:
  mode: -rw-r--r--  links: 1  owner: root      group: root
  length:          0  admin id: 0  inode:      1027.1
  access:          Aug 18 2005  modification: Aug 18 11:19
  changed:         Aug 18 11:19  attributes:   Aug 18 11:19
  creation:        Aug 18 11:19  residence:    Aug 18 11:19

# chmod 4000 test
# sls -D
test:
  mode: -r-Sr--r--  links: 1  owner: root      group: root
  length:          0  admin id: 0  inode:      1027.1
  access:          Aug 18 2005  modification: Aug 18 11:19
  changed:         Aug 18 11:19  retention-end: Aug 18 2005 11:25
  creation:        Aug 18 11:19  residence:    Aug 18 11:19
  retention:       active          retention-period: 0y, 0d, 0h, 6m
```

The -a option for touch is used to change the access time of the file. The -t option specifies what time is to be used for the access time field. The format for the time argument is [[CC]YY]MMDDhhmm[.SS], as follows:

- [CC] – The first two digits of the year.
- [YY] – The second two digits of the year.
- MM – The month of the year (01-12).
- DD – The day of the month (01-31).
- hh – The hour of the day (00-23).
- mm – The minute of the hour (00-59).
- [SS] – The second of the minute (00-61).

The CC, YY, and SS fields are optional. If CC and YY are not given, the default is the current year. See the touch manpage for more information on these options.

To set the retention period to permanent retention, set the access time to its largest possible value: 203801182214.07.

Using `sls` to View WORM-FS Files

Use the `sls` command to view WORM file attributes. The `-D` option shows whether a directory is WORM-capable. Use this option on a file to display when the retention period began, when it will end, the current retention state, and the duration as specified on the command line.

The start of the retention period is stored in the file's `changed` attribute field. The end of the retention period is stored in the file's `attribute time` field. This time is displayed as a calendar date. An additional line in the `sls` output shows the retention period state and duration.

CODE EXAMPLE 5-5 shows an example of how `sls -D` displays a file's retention status.

CODE EXAMPLE 5-5 Using `sls` to Find a File's Retention Status

```
sls -D test
test:
  mode: -r-Sr--r--  links: 1  owner: root group: root
  length: 5  admin id: 0  inode: 1027.1
  access: Aug 18 2005  modification: Aug 18 11:19
  changed: Aug 18 11:19  retention-end: Aug 18 2005 11:25
  creation: Aug 18 11:19  residence: Aug 18 11:19
  retention: active retention-period: 0y, 0d, 0h, 6m
```

In this example, the retention state is active, as shown by the `retention: active` designation, meaning that the file has the WORM bit set. The retention period started on August 18, 2005, at 11:19 and will end on August 18, 2005, at 11:25. The retention period was specified to be 0 years, 0 days, 0 hours, and 6 minutes.

Using touch to Extend a File's Retention Period

The touch utility can also be used to extend a file's retention period by advancing the file's access time. CODE EXAMPLE 5-6 shows an example of using touch to extend a file's retention period.

CODE EXAMPLE 5-6 Using touch to Extend a File's Retention Period

```
# sls -D test
test:
  mode: -r-Sr--r--  links: 1  owner: root      group: root
  length:           0  admin id: 0  inode: 1029.1
  access:           Aug 18 11:35  modification: Aug 18 11:33
  changed:          Aug 18 11:33  retention-end: Aug 18 2005 11:35
  creation:         Aug 18 11:33  residence: Aug 18 11:33
  retention:        over          retention-period: 0y, 0d, 0h, 2m
# touch -a -t200508181159 test
# sls -D
test:
  mode: -r-Sr--r--  links: 1  owner: root      group: root
  length:           0  admin id: 0  inode: 1029.1
  access:           Aug 18 11:35  modification: Aug 18 11:33
  changed:          Aug 18 11:33  retention-end: Aug 18 2005 11:59
  creation:         Aug 18 11:33  residence: Aug 18 11:33
  retention:        active        retention-period: 0y, 0d, 0h, 26m
```

In this example the retention period was extended to Aug 18, 2005 at 11:59AM, which is 26 minutes from the time the WORM trigger was initially applied.

Using sfind to Find WORM-FS Files

Use the sfind utility to search for files that have certain retention periods. The following sfind options are available:

- **-ractive** – Finds files whose retention period is active.
- **-rover** – Finds files whose retention periods have expired.
- **-rafter *date*** – Finds files whose retention period will end after the specified date. The date is specified as YYYYMMDDHHmm, where YYYY is the year, MM is the month, DD is the day, HH is the hour, and mm is minutes.

For example, CODE EXAMPLE 5-7 shows the command to find files whose retention period expires after 12/24/2004 at 15:00.

CODE EXAMPLE 5-7 Using `sfind` to Find All WORM Files That Expire After a Certain Date

```
# sfind -rafter 200412241500
```

- `-rremain time` – Finds files that have retention periods with at least the specified amount of time left. The time is specified as `MyNdOhPm`, where `M`, `N`, `O`, and `P` are arbitrary non-negative integers and `y`, `d`, `h`, and `m` represent the number of years, days, hours, and minutes, respectively.

For example, shows the command to find files for which more than 1 year, 10 days, 5 hours, and 10 minutes remain before expiration.

CODE EXAMPLE 5-8 Using `sfind` to Find All WORM Files With More Than a Specified Time Remaining

```
# sfind -rremain 1y10d5h10m
```

- `-rlonger time` – Finds files that have retention periods longer than the specified amount of time. The time is specified as `MyNdOhPm`, where `M`, `N`, `O`, and `P` are arbitrary non-negative integers and `y`, `d`, `h`, and `m` represent the number of years, days, hours, and minutes, respectively.

For example, shows the command to find files that have retention periods longer than 10 days.

CODE EXAMPLE 5-9 Using `sfind` to Find All WORM Files With Longer Than a Specified Retention Period

```
# sfind -rlonger 10d
```

- `-rpermanent` – Finds files whose retention period is permanent.

Accommodating Large Files

- When manipulating very large files, pay careful attention to the size of disk cache that is available on the system. If you try to write a file that is larger than your disk cache, the program blocks, waiting for space that might never exist, because the available disk space is insufficient to handle the request.

If your application must write a file that is larger than the disk cache, you can segment the file with the `segment(1)` command. For more information about the `segment(1)` command, see the `segment(1)` man page or see the *Sun StorEdge SAM-FS Storage and Archive Management Guide*.

Understanding I/O Types

The Sun StorEdge SAM-FS file systems support paged I/O, direct I/O, and switching between the I/O types. The following sections describe these I/O types.

Paged I/O

When paged I/O is used, user data is cached in virtual memory pages, and the kernel writes the data to disk. The standard Solaris OS interfaces manage paged I/O. Paged I/O (also called buffered or cached I/O) is selected by default.

Direct I/O

Direct I/O is a process by which data is transferred directly between the user's buffer and the disk. This means that much less time is spent in the system. For performance purposes, specify direct I/O only for large, block-aligned, sequential I/O.

The `setfa(1)` command and the `sam_setfa(3)` library routine both have a `-D` option that sets the direct I/O attribute for a file or directory. If applied to a directory, files and directories created in that directory inherit the direct I/O attribute. After the `-D` option is set, the file uses direct I/O.

You can also select direct I/O for a file by using the Solaris OS `directio(3C)` function call. If you use the function call to enable direct I/O, the setting lasts only while the file is active.

To enable direct I/O on a file-system basis, do one of the following:

- Specify the `-o forcedirectio` option with the `mount(1M)` command.
- Put the `forcedirectio` keyword in the mount option column of the `/etc/vfstab` file, or use it as a directive in the `samfs.cmd` file.

For more information, see the `setfa(1)`, `sam_setfa(3)`, `directio(3C)`, `samfs.cmd(4)`, and `mount_samfs(1M)` man pages.

I/O Switching

By default, paged I/O is performed, and I/O switching is disabled. However, the Sun StorEdge SAM-FS file systems support automatic I/O switching, a process by which a site-defined amount of paged I/O occurs before the system switches automatically to direct I/O.

I/O switching should reduce page cache usage on large I/O operations. To enable I/O switching, use `samu(1M)`, or use the `dio_wr_consec` and `dio_rd_consec` parameters as directives in the `samfs.cmd` file or as options with the `mount(1M)` command.

For more information about these options, see the `mount_samfs(1M)` or `samfs.cmd(4)` man pages.

Increasing File Transfer Performance for Large Files

Sun StorEdge SAM-FS file systems are tuned to work with a mix of file sizes. You can increase the performance of disk file transfers for large files by enabling file system settings.

Note – Sun recommends that you experiment with performance tuning outside of a production environment. Tuning these variables incorrectly can have unexpected effects on the overall system.

If your site has a Sun Enterprise Services (SES) support contract, please inform SES if you change performance tuning parameters.

▼ To Increase File Transfer Performance

1. Set the maximum device read/write directive.

The `maxphys` parameter in the Solaris `/etc/system` file controls the maximum number of bytes that a device driver reads or writes at any one time. The default value for the `maxphys` parameter can differ, depending on the level of your Sun Solaris OS, but it is typically around 128 kilobytes.

Add the following line to `/etc/system` to set `maxphys` to 8 megabytes:

```
set maxphys = 0x800000
```

2. Set the SCSI disk maximum transfer parameter.

The `sd` driver enables large transfers for a specific file by looking for the `sd_max_xfer_size` definition in the `/kernel/drv/sd.conf` file. If this definition does not exist, the driver uses the value defined in the `sd` device driver definition, `sd_max_xfer_size`, which is 1024 x 1024 bytes.

To enable and encourage large transfers, add the following line at the end of the `/kernel/drv/sd.conf` file:

```
sd_max_xfer_size=0x800000;
```

3. Set the fibre disk maximum transfer parameter.

The `ssd` driver enables large transfers for a specific file by looking for the `ssd_max_xfer_size` definition in the `/kernel/drv/ssd.conf` file. If this definition does not exist, the driver uses the value defined in the `ssd` device driver definition, `ssd_max_xfer_size`, which is 1024 x 1024 bytes.

Add the following line at the end of the `/kernel/drv/ssd.conf` file:

```
ssd_max_xfer_size=0x800000;
```

4. Reboot the system.

5. Set the `writebehind` parameter.

This step affects paged I/O only.

The `writebehind` parameter specifies the number of bytes that are written behind by the file system when paged I/O is being performed on a Sun StorEdge SAM-FS file system. Matching the `writebehind` value to a multiple of the RAID's read-modify-write value can increase performance.

This parameter is specified in units of kilobytes and is truncated to an 8-kilobyte multiple. If set, this parameter is ignored when direct I/O is performed. The default `writebehind` value is 512 kilobytes. This value favors large-block, sequential I/O.

Set the `writebehind` size to a multiple of the RAID 5 stripe size for both hardware and software RAID-5. The RAID-5 stripe size is the number of data disks multiplied by the configured stripe width.

For example, assume that you configure a RAID-5 device with three data disks plus one parity disk (3+1) with a stripe width of 16 kilobytes. The `writebehind` value should be 48 kilobytes, 96 kilobytes, or some other multiple, to avoid the overhead

of the read-modify-write RAID-5 parity generation.

For Sun StorEdge SAM-FS file systems, the DAU (`sammkfs(1M) -a` command) should also be a multiple of the RAID-5 stripe size. This allocation ensures that the blocks are contiguous.

You should test the system performance after resetting the `writebehind` size. The following example shows testing timings of disk writes:

```
# timex dd if=/dev/zero of=/sam/myfile bs=256k count=2048
```

You can set the `writebehind` parameter from a mount option, from within the `samfs.cmd` file, from within the `/etc/vfstab` file, or from a command within the `samu(1M)` utility. For information about enabling this from a mount option, see the `-o writebehind=n` option on the `mount_samfs(1M)` man page. For information about enabling this from the `samfs.cmd` file, see the `samfs.cmd(4)` man page. For information about enabling this from within `samu(1M)`, see the `samu(1M)` man page.

6. Set the `readahead` parameter.

This step affects paged I/O only.

The `readahead` parameter specifies the number of bytes that are read ahead by the file system when paged I/O is being performed on a Sun StorEdge SAM-FS file system. This parameter is specified in units of kilobytes and is truncated to an 8-kilobyte multiple. If set, this parameter is ignored when direct I/O is performed.

Increasing the size of the `readahead` parameter increases the performance of large file transfers, but only to a point. You should test the performance of the system after resetting the `readahead` size until you see no more improvement in transfer rates. The following is an example method of testing timings on disk reads:

```
# timex dd if=/sam/myfile of=/dev/null bs=256k
```

You should test various `readahead` sizes for your environment. The `readahead` parameter should be set to a size that increases the I/O performance for paged I/O, but is not so large as to hurt performance. It is also important to consider the amount of memory and number of concurrent streams when you set the `readahead` value. Setting the `readahead` value multiplied by the number of streams to a value that is greater than memory can cause page thrashing.

The default `readahead` value is 1024 kilobytes. This value favors large-block, sequential I/O. For short-block, random I/O applications, set `readahead` to the typical request size. Database applications do their own read-ahead, so for these applications, set `readahead` to 0.

The `readahead` setting can be enabled from a mount option, from within the `samfs.cmd` file, from within the `/etc/vfstab` file, or from a command within the `samu(1M)` utility. For information about enabling this setting from a mount option,

see the `-o readahead=n` option on the `mount_samfs(1M)` man page. For information about enabling this setting from the `samfs.cmd` file, see the `samfs.cmd(4)` man page. For information about enabling this setting from within `samu(1M)`, see the `samu(1M)` man page.

7. Set the stripe width.

The `-o stripe=n` option with the `mount(1M)` command specifies the stripe width for the file system. The stripe width is based on the disk allocation unit (DAU) size. The *n* argument specifies that *n* x DAU bytes are written to one device before writing switches to the next device. The DAU size is set when the file system is initialized by the `sammkfs(1M) -a` command.

If `-o stripe=0` is set, files are allocated to file system devices using the round-robin allocation method. With this method, each file is completely allocated on one device until that device is full. Round-robin is the preferred setting for a multistream environment. If `-o stripe=n` is set to an integer greater than 0, files are allocated to file system devices using the stripe method. To determine the appropriate `-o stripe=n` setting, try varying the setting and taking performance readings. Striping is the preferred setting for turnkey applications with a required bandwidth.

You can also set the stripe width from the `/etc/vfstab` file or from the `samfs.cmd` file.

For more information about the `mount(1M)` command, see the `mount_samfs(1M)` man page. For more information about the `samfs.cmd` file, see the `samfs.cmd(4)` man page.

Setting the Write Throttle

By default, the Sun StorEdge SAM-FS file systems set the

`-o wr_throttle=n` option to the `mount(1M)` command to 16 megabytes. The `-o wr_throttle=n` option limits the number of outstanding write kilobytes for one file to *n*.

If a file has *n* write kilobytes outstanding, the system suspends an application that attempts to write to that file until enough bytes have completed the I/O to allow the application to be resumed.

If your site has thousands of streams, such as thousands of NFS-shared workstations accessing the file system, you can tune the `-o wr_throttle=n` option in order to avoid flushing excessive amounts of memory to disk at once. Generally, the number

of streams multiplied by 1024 x the *n* argument to the `-o wr_throttle=n` option should be less than the total size of the host system's memory minus the memory needs of the Solaris OS, as shown in this formula:

$$\text{number-of-streams} \times n \times 1024 < \text{total-memory} - \text{Solaris-OS-memory-needs}$$

For turnkey applications, you might want to use a size larger than the default 16,384 kilobytes, because this keeps more pages in memory.

Setting the Flush-Behind Rate

Two mount parameters control the flush-behind rate for pages written sequentially and for stage pages. The `flush_behind` and `stage_flush_behind` mount parameters are read from the `samfs.cmd` file, the `/etc/vfstab` file, or the `mount(1M)` command.

The `flush_behind=n` mount parameter sets the maximum flush-behind value. Modified pages that are being written sequentially are written to disk asynchronously to help the Sun Solaris Volume Manager (VM) layer keep pages clean. To enable this feature, set *n* to be an integer from 16 through 8192. By default, *n* is set to 0, which disables this feature. The *n* argument is specified in kilobyte units.

The `stage_flush_behind=n` mount parameter sets the maximum stage flush-behind value. Stage pages that are being staged are written to disk asynchronously to help the Sun Solaris VM layer keep pages clean. To enable this feature, set *n* to be an integer from 16 through 8192. By default, *n* is set to 0, which disables this feature. The *n* argument is specified in kilobyte units.

For more information about these mount parameters, see the `mount_samfs(1M)` man page or the `samfs.cmd(4)` man page.

Tuning the Number of Inodes and the Inode Hash Table

The Sun StorEdge SAM-FS file system enables you to set the following two tunable parameters in the `/etc/system` file:

- `ninodes`
- `nhino`

To enable nondefault settings for these parameters, edit the `/etc/system` file, and then reboot your system.

The following subsections describe these parameters in more detail.

The `ninodes` Parameter

The `ninodes` parameter specifies the maximum number of default inodes. The value for `ninodes` determines the number of in-core inodes that Sun StorEdge SAM-FS software keeps allocated to itself, even when applications are not using many inodes.

The format for this parameter in the `/etc/system` file is as follows:

```
set samfs:ninodes = value
```

The range for *value* is from 16 through 2000000. The default value for `ninodes` is one of the following:

- A value that is equal to the `ncsize` setting. The `ncsize` parameter is a Solaris tuning parameter that specifies the number of entries in the directory name look-up cache (DNLC). For more information about `ncsize`, see the *Solaris Tunable Parameters Reference Manual*.
- 2000. The file systems set `ninodes` to 2000 if the `ncsize` setting is zero or out of range.

The nhino Parameter

The `nhino` parameter specifies the size of the in-core inode hash table.

The format for this parameter in the `/etc/system` file is as follows:

```
set samfs:nhino = value
```

The range for *value* is 1 through 1048756. *value* must be a nonzero power of 2. The default value for `nhino` is one of the following:

- A value that is equal to the `ninodes` value divided by 8 and then, if necessary, rounded up to the nearest power of 2. For example, assume that the following line exists in `/etc/system`:

```
set samfs:ninodes 8000
```

For this example, if `nhino` is not set, the system assumes 1024, which is 8000 divided by 8 and then rounded up to the nearest power of 2 (2^{10}).

- 512. The file systems set `nhino` to 512 if the `ninodes` setting is out of range.

When to Set the `ninodes` and `nhino` Parameters

When searching for an inode by number (after obtaining an inode number from a directory or after extracting an inode number from an NFS file handle), a Sun StorEdge SAM-FS file system searches its cache of in-core inodes. To speed this process, the file system maintains a hash table to decrease the number of inodes it must check.

A larger hash table reduces the number of comparisons and searches, at a modest cost in memory usage. If the `nhino` value is too large, the system is slower when undertaking operations that sweep through the entire inode list (inode syncs and unmounts). For sites that manipulate large numbers of files and sites that do extensive amounts of NFS I/O, it can be advantageous to set these parameter values to larger than the defaults.

If your site has file systems that contain only a small number of files, it might be advantageous to make these numbers smaller than the defaults. This could be the case, for example, if you have a file system into which you write large single-file `tar(1)` files to back up other file systems.

Troubleshooting Sun StorEdge QFS

This appendix describes some tools and procedures that can be used to troubleshoot issues with the Sun StorEdge SAM-FS file system. Specifically, it contains the following topics:

- “Checking File System Integrity and Repairing File Systems” on page 93

Checking File System Integrity and Repairing File Systems

Sun StorEdge SAM-FS file systems write validation data in the following records that are critical to file system operations: directories, indirect blocks, and inodes. If the file system detects corruption while searching a directory, it issues an EDOM error, and the directory is not processed. If an indirect block is not valid, it issues an ENOCSI error, and the file is not processed. TABLE A-1 summarizes these error indicators.

TABLE A-1 Error Indicators

Error	Solaris OS Meaning	Sun StorEdge SAM-FS Meaning
EDOM	Argument is out of domain.	Values in validation records are out of range.
ENOCSI	No CSI structure is available.	Links between structures are invalid.

In addition, inodes are validated and cross checked with directories.

You should monitor the following files for error conditions:

- The log file specified in `/etc/syslog.conf` for the errors shown in TABLE A-1.

- The `/var/adm/messages` file for device errors.

If a discrepancy is noted, you should unmount the file system and check it using the `samfsck(1M)` command.

Note – The `samfsck(1M)` command can be issued on a mounted file system, but the results cannot be trusted. Because of this, you are encouraged to run the command on an unmounted file system only.

▼ To Check a File System

- Use the `samfsck(1M)` command to perform a file systems check.

Use this command in the following format:

```
samfsck -V family-set-name
```

For *family-set-name*, specify the name of the file system as specified in the `mcf(4)` file.

You can send output from `samfsck(1M)` to both your screen and to a file by using it in conjunction with the `tee(1)` command, as follows.

- C shell:

```
# samfsck -V family-set-name |& tee file
```

- Bourne shell:

```
# samfsck -V family-set-name 2>&1 | tee file
```

Nonfatal errors returned by `samfsck(1M)` are preceded by `NOTICE`. Nonfatal errors are lost blocks and orphans. The file system is still consistent if `NOTICE` errors are returned. You can repair these nonfatal errors during a convenient, scheduled maintenance outage.

Fatal errors are preceded by `ALERT`. These errors include duplicate blocks, invalid directories, and invalid indirect blocks. The file system is not consistent if these errors occur. Notify Sun if the `ALERT` errors cannot be explained by a hardware malfunction.

If the `samfsck(1M)` command detects file system corruption and returns `ALERT` messages, you should determine the reason for the corruption. If hardware is faulty, repair it before repairing the file system.

For more information about the `samfsck(1M)` and `tee(1)` commands, see the `samfsck(1M)` and `tee(1)` man pages.

▼ To Repair a File System

1. **Use the `umount(1M)` command to unmount the file system.**

Run the `samfsck(1M)` command when the file system is not mounted. For information about unmounting a file system, see “Unmounting a File System” on page 38.

2. **Use the `samfsck(1M)` command to repair a file system. If you are repairing a shared file system, issue the command from the metadata server.**

You can issue the `samfsck(1M)` command in the following format to repair a file system:

```
# samfsck -F -V fsname
```

For *fsname*, specify the name of the file system as specified in the `mcf(4)` file.

Using the samu(1M) Operator Utility

This chapter shows how to use `samu(1M)` to control the devices configured within your Sun StorEdge SAM-FS environment. Many `samu(1M)` displays are useful only for sites using the storage and archive management mechanism.

This chapter contains the following sections:

- “Overview” on page 97
- “Operator Displays” on page 100
- “Operator Commands” on page 150

The operations that you can perform from within `samu(1M)` can also be performed by using the `samcmd(1M)` command. For more information about `samcmd(1M)`, see the `samcmd(1M)` man page.

Overview

The `samu(1M)` operator utility requires a display terminal that displays a minimum of 24 lines by 80 characters wide. The utility includes the following features:

- Displays that enable you to monitor Sun StorEdge SAM-FS devices and file system activity.
- Commands that enable you to select displays, set display options, control access to and the activity of devices, and take snapshots of display windows.
- Commands that enable you to tune a running Sun StorEdge SAM-FS file system.

The display windows shown in this chapter are representative examples. The exact format and amount of information displayed on your terminal can be different depending on your terminal model and the devices configured in your Sun StorEdge SAM-FS environment.

The following sections describe how to start and stop `samu(1M)`, interact with the utility, access the help windows, and view operator displays.

▼ To Invoke `samu(1M)`

- To start `samu(1M)`, type the `samu(1M)` command from the command line:

```
# samu
```

The system starts `samu(1M)` and shows the help display. This is the default initial display. To view a different `samu(1M)` display, follow the steps in “To Display a `samu(1M)` Screen” on page 98.

The `samu(1M)` utility enables you to select its initial display. For more information about the `samu(1M)` command line options, see the `samu(1M)` man page.

Note – `samu(1M)`, like the `vi(1)` editor, is based on the `curses(3CURSES)` routine. If you have trouble invoking `samu(1M)`, make sure that your terminal type is defined correctly.

▼ To Display a `samu(1M)` Screen

The `samu(1M)` command accepts options on its command line for displaying different `samu(1M)` screens.

1. **Type a colon (:) to bring up the `samu(1M)` prompt.**

After you type in the colon, the following appears in the lower left:

```
Command:
```

2. Type the letter that corresponds to the display you want to view and press return.

For example, to view the `v` display, type a `v` and press Return after the `Command:` prompt.

For a complete list of letters to type and the displays to which they correspond, see “(h) - Help Display” on page 111.

▼ To Stop `samu(1M)`

- **To exit `samu(1M)`, type one of the following:**

- `q`
- `:q`

The `samu(1M)` operator utility exits and returns you to the command shell.

Interacting With `samu(1M)`

Interacting with `samu(1M)` is similar to interacting with the UNIX `vi(1)` editor with respect to paging forward or backward, entering commands, refreshing the display, and quitting the utility.

Each display has its own section in this chapter, and each display section shows the control key sequences you can use to navigate in that display. The `samu(1M)` man page summarizes the control key navigation sequences.

The last line of the display window shows the command and display error messages. If a command error occurs, automatic display refreshing halts until the next operator action.

Entering a Device

Each device included in the Sun StorEdge SAM-FS environment is assigned an Equipment Ordinal (for example, 10) in the `mcf(4)` file. Many `samu(1M)` commands reference a specific device using that Equipment Ordinal.

Example. The syntax for the `:off` command is as follows:

```
:off eq
```

For `eq`, type the Equipment Ordinal for the device you are trying to address.

Getting Online Help

When you start `samu(1M)`, the system automatically displays the first help screen. This help screen differs depending on whether you have the Sun StorEdge QFS or the Sun StorEdge SAM-FS software installed.

For more information about the help (h) display, see “(h) - Help Display” on page 111.

▼ To Access Online Help From a Display Screen

- **Type** `:h`

To move forward or backward from one screen to the next, type the following key sequence:

- Press `Ctrl-f` to page the display forward.
- Press `Ctrl-b` to page the display backward to previous pages.

You can return to the help display at any time by pressing the `h` key.

Operator Displays

You can view the `samu(1M)` operator displays by pressing the key corresponding to each display. The lowercase keys `a` through `w` display operational information.

Note – The uppercase `samu(1M)` displays (`A`, `D`, `C`, `F`, `I`, `J`, `L`, `M`, `N`, `P`, `R`, `S`, `T`, and `U`) are designed to be used at a customer site only with the assistance of a member of the technical support staff.

This chapter does not describe these uppercase displays as thoroughly as the lowercase displays.

For displays that overflow the screen area, the word `more` appears on the bottom of the screen display, indicating that the display contains additional information. You can use `Ctrl-f` to page forward and see more content.

CODE EXAMPLE 5-10 contains the word `more`, indicating that more information appears on subsequent screens.

CODE EXAMPLE 5-10 samu(1M) Screen Indicating More Text Can Be Obtained

xb54	54	exb8505	pt03	0	yes	2	0	on	
lt55	55	dlt2000	pt02	1	yes	4	0	on	ml65
hp56	56	hpc1716	pt01	1	yes	3	0	on	hp70
hp57	57	hpc1716	pt01	1	yes	4	0	on	hp70
more									

If `samu(1M)` prompts you to enter a device, enter its associated Equipment Ordinal. The configuration display (c) shows Equipment Ordinals for all removable media devices. To control all displays, use the control keys listed for the display.

The following sections describe the operator displays in alphabetical order. Examples are provided, and when necessary, displays are followed by a table describing the fields displayed.

(a) - Archiver Status Display

The a display shows the archiver status.

You can invoke this display differently, depending on what you need to view, as follows:

- To display an archiver status summary, which shows the status of the archiver on a per-file-system basis, type the command with the following format:

```
Command: a
```

- To display archiving details for a specific file system, type the command with the following format:

```
Command: a filesystem
```

For *filesystem*, specify the name of a file system.

Navigation

TABLE 5-3 shows the control keys you can use in the `a` display.

TABLE 5-3 Control Keys for the `a` Display

Key	Function
Ctrl-b	Previous file system
Ctrl-f	Next file system
Ctrl-d	Page <i>arcopies</i> forward (bottom portion)
Ctrl-u	Page <i>arcopies</i> backward (bottom portion)

TABLE 5-4 shows the control keys you can use in the `:a filesystem` display.

TABLE 5-4 Control Keys for the `:a filesystem` Display

Key	Function
Ctrl-b	Previous file system
Ctrl-f	Next file system

Sample Display

CODE EXAMPLE 5-11 shows activity and statistics for a single file system in the summary display.

CODE EXAMPLE 5-11 `samu(1M) a` Display

```
Archiver status                      samu 4.4 07:44:02 August 8 2005

sam-archiverd:  Waiting for resources

sam-arfind:  samfs1 mounted at /sam1
Waiting until 2005-05-08 07:54:02 to scan .inodes

sam-arfind:  samfs2 mounted at /sam2
Waiting until 2005-05-08 07:52:57 to scan .inodes

sam-arfind:  qfs1 mounted at /qfs1
Waiting until 2005-05-08 07:44:33 to scan .inodes

sam-arfind:  qfs2 mounted at /qfs2
Waiting until 2005-05-08 07:53:21 to scan .inodes
```

CODE EXAMPLE 5-11 samu(1M) a Display (Continued)

```
sam-arfind: qfs3 mounted at /qfs3
Waiting until 2005-05-08 07:44:11 to scan .inodes

sam-arfind: qfs4 mounted at /qfs4
Waiting until 2005-05-08 07:53:35 to scan .inodes

sam-arfind: shareqfs1 mounted at /shareqfs1
Shared file system client. Cannot archive.

sam-arfind: shareqfs2 mounted at /shareqfs2
Shared file system client. Cannot archive.

sam-arcopy: qfs4.arset5.1.83 dt.DAT001
Waiting for volume dt.DAT001
```

Field Descriptions

TABLE 5-5 shows the fields in the detail display.

TABLE 5-5 samu(1M) a Display Field Descriptions

Field	Description
samfs1 mounted at	Mount point.
regular files	Number of regular files and their total size.
offline files	Number of offline files and their total size.
archdone files	Number of archdone files and size. Indicates that the archiver has completed processing and can perform no further processing for archdone files. Files marked as archdone have been processed for archiving but have not necessarily been archived.
copy1	Number of files and total size for archive copy 1.
copy2	Number of files and total size for archive copy 2.
copy3	Number of files and total size for archive copy 3.
copy4	Number of files and total size for archive copy 4.
Directories	Number of directories and total size.
sleeping until	Indicates when archiver runs again.

(c) - Device Configuration Display

The `c` display shows your configuration's connectivity. It lists all device names and Equipment Ordinals.

To invoke the device configuration display, type the command with the following format:

```
Command: c
```

Navigation

TABLE 5-6 shows the control keys you can use in this display.

TABLE 5-6 Control Keys for the `c` Display

Key	Function
Ctrl-b	Page backward
Ctrl-d	Half-page forward
Ctrl-f	Page forward
Ctrl-u	Half-page backward

Sample Display

CODE EXAMPLE 5-12 shows the device configuration display.

CODE EXAMPLE 5-12 samu(1M) `c` Display

Device configuration:				samu	4.4 07:48:11 Sept 8 2005
ty	eq	state	device_name	fs	family_set
sk	100	on	/etc/opt/SUNWsamfs/dcstkconf	100	dcL700
tp	120	off	/dev/rmt/1cbn	100	dcL700
sg	130	on	/dev/rmt/4cbn	100	dcL700
sg	140	on	/dev/rmt/5cbn	100	dcL700
tp	150	off	/dev/rmt/3cbn	100	dcL700
hy	151	on	historian	151	

Field Descriptions

TABLE 5-7 shows the field descriptions for this display.

TABLE 5-7 samu(1M) c Display Field Descriptions

Field	Description
ty	Device type.
eq	Equipment Ordinal of the device.
state	Current operating state of the device. Valid device states are as follows: <ul style="list-style-type: none">• on—The device is available for access.• ro—The device is available for read-only access.• off—The device is not available for access.• down—The device is available only for maintenance access.• idle—The device is not available for new connections. Operations in progress continue until completion.
device_name	Path to the device.
fs	Family Set Equipment Ordinal.
family_set	Name of the storage Family Set or library to which the device belongs.

(C) - Memory Display

The C display shows the content of a specified memory address. To show the content at an address, enter the address in hexadecimal.

To invoke this display, type the following command:

```
Command: C hex-address
```

For *hex-address*, specify the address of a memory location in hexadecimal. For example:

```
Command: C 0x1044a998
```

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

Sample Display

CODE EXAMPLE 5-13 shows the memory display. The output has been truncated for inclusion in this manual.

CODE EXAMPLE 5-13 samu(1M) C Display

Memory	base: 0x1234567				samu 4.4 07:52:25 Sept 8 2005
00000000	80921000	137ff801	edd05e21	40853420x.mP^!@.4
00000010	00a00420	018528b0	01a604e0	02840640 (0.&.`'...@
00000020	02d030a1	a0853420	0080a0a0	100a6fff	.P0! .4o.
00000030	f6921000	13c65e23	582d0000	0ba01020	v....F^#X-... .
00000040	00c45e20	48c608e0	2fd05e21	40920080	.D^ HF.`'/P^!@...
00000050	037ff801	fa941000	16c45e20	48a600a0	..x.z....D^ H&.
00000060	80921000	137ff801	d5d05e21	40853420x.UP^!@.4
00000070	00a00420	018528b0	01a604e0	02840640 (0.&.`'...@
00000080	02d030a1	c0853420	0080a0a0	100a6fff	.P0!@.4o.
00000090	f6921000	13c65e23	58a01020	00c45e20	v....F^#X . .D^
000000a0	48c608e0	2fd05e21	40920080	037ff801	HF.`'/P^!@.....x.
000000b0	e39405a2	00c45e20	48a600a0	80921000	c..".D^ H&
000000c0	137ff801	bed05e21	40853420	00a00420	..x.>P^!@.4 . . .
000000d0	018528b0	01a604e0	02840640	02d030a1	.. (0.&.`'...@.P0!
000000e0	e0853420	0080a0a0	100a6fff	f6921000	`'.4o.v...
000000f0	13c65e23	58a01020	00c45e20	48c608e0	.F^#X . .D^ HF.`

(d) - Daemon Trace Controls Display

The d display shows the events being traced as specified in the defaults.conf file. For more information about enabling trace files, see the defaults.conf(4) man page.

To invoke this display, type the following command:

```
Command: d
```

Sample Display

CODE EXAMPLE 5-14 shows trace file information. It includes information about the daemons being traced, the paths to the trace files, the events being traced, and information about the size and age of the trace files.

CODE EXAMPLE 5-14 samu(1M) d Display

Daemon trace controls		samu 4.4 07:56:38 Sept 8 2005
sam-amld	/var/opt/SUNWsamfs/trace/sam-amld	
	cust err fatal misc proc debug date	
	size 0 age 0	
sam-archiverd	/var/opt/SUNWsamfs/trace/sam-archiverd	
	cust err fatal misc proc debug date	
	size 0 age 0	
sam-catserverd	/var/opt/SUNWsamfs/trace/sam-catserverd	
	cust err fatal misc proc debug date	
	size 0 age 0	
sam-fsd	/var/opt/SUNWsamfs/trace/sam-fsd	
	cust err fatal misc proc debug date	
	size 0 age 0	
sam-rftd	/var/opt/SUNWsamfs/trace/sam-rftd	
	cust err fatal misc proc debug date	
	size 0 age 0	
sam-recycler	/var/opt/SUNWsamfs/trace/sam-recycler	
	cust err fatal misc proc debug date	
	size 0 age 0	
sam-sharefsd	/var/opt/SUNWsamfs/trace/sam-sharefsd	
	cust err fatal misc proc debug date	
	size 0 age 0	
sam-stagerd	/var/opt/SUNWsamfs/trace/sam-stagerd	
	cust err fatal misc proc debug date	
	size 0 age 0	
sam-serverd	/var/opt/SUNWsamfs/trace/sam-serverd	
	cust err fatal misc proc debug date	
	size 0 age 0	
sam-clientd	/var/opt/SUNWsamfs/trace/sam-clientd	
	cust err fatal misc proc debug date	
	size 0 age 0	

CODE EXAMPLE 5-14 samu(1M) d Display (Continued)

```
sam-mgmt      /var/opt/SUNWsamfs/trace/sam-mgmt
              cust err fatal misc proc debug date
              size    0      age 0
```

(D) - Disk Volume Dictionary

The **D** display shows the disk volume dictionary, which keeps track of the disk media for disk archiving that has been defined in the `diskvols.conf` file. The dictionary contains information about each VSN, including the capacity, space remaining, and flags indicating the status of the VSN. These flags include *unavailable*, *read only*, and *bad media*.

To invoke this display, type the following command:

```
Command:D
```

Sample Display

CODE EXAMPLE 5-15 shows the device configuration display.

CODE EXAMPLE 5-15 samu(1M) D Display

```
Disk volume dictionary samu      4.4 07:48:11 May 8 2005

volumes
magic 340322 version 9 nkeys 2 ndata 2
index  space          capacity    flags    volume
   0   12801654784    17182949376  -----  disk01
   1   12801654784    17182949376  -----  disk02

clients
magic 340322 version 9 nkeys 1 ndata 1
index flags client
   0    0   graul-mn
```

Flags

TABLE 5-8 shows the flags for the D display.

TABLE 5-8 Flags Field for the samu(1M) D Display

Field	Description
l----	Volume is labeled; seqnum file has been created. This is set by the administrator to prevent the software from creating a new seqnum file.
-r---	Volume is defined on a remote host.
--U--	Volume is unavailable.
---R-	Volume is read only.
----E	Media error. Set when the software detects a write error on the disk archive directory.

The diskvols samu(1M) command can be used to set or clear the disk volume dictionary flags. See “The :diskvols volume [+flag | -flag] Command” on page 166.

(f) - File Systems Display

The f display shows the components of your Sun StorEdge SAM-FS file systems.

To invoke this display, type the following command:

Command: **f**

Sample Display

CODE EXAMPLE 5-16 shows the file systems display.

CODE EXAMPLE 5-16 samu(1M) f Display

File systems										samu 4.4 08:11:24 Sept 8 2005										
ty	eq	state	device_name		status		high	low	mountpoint	server										
ms	10	on	samfs1		m----	2----	d	90%	70%	/sam1										
md	11	on	/dev/dsk/c5t8d0s3																	
md	12	on	/dev/dsk/c5t8d0s4																	
md	13	on	/dev/dsk/c5t8d0s5																	
md	14	on	/dev/dsk/c5t8d0s6																	

CODE EXAMPLE 5-16 samu(1M) f Display (Continued)

md	15	on	/dev/dsk/c5t8d0s7						
ms	20	on	samfs2	m----	2----	d	90%	70%	/sam2
md	21	on	/dev/dsk/c5t9d0s3						
md	22	on	/dev/dsk/c5t9d0s4						
md	23	on	/dev/dsk/c5t9d0s5						
md	24	on	/dev/dsk/c5t9d0s6						
md	25	on	/dev/dsk/c5t9d0s7						
ma	30	on	qfs1	m----	2----	d	90%	70%	/qfs1
mm	31	on	/dev/dsk/c5t10d0s0						
md	32	on	/dev/dsk/c5t10d0s1						
ma	40	on	qfs2	m----	2----	d	90%	70%	/qfs2
mm	41	on	/dev/dsk/c5t11d0s0						
md	42	on	/dev/dsk/c5t11d0s1						
ma	50	on	qfs3	m----	2---	r-	90%	70%	/qfs3
mm	51	on	/dev/dsk/c5t12d0s0						
mr	52	on	/dev/dsk/c5t12d0s1						
ma	60	on	qfs4	m----	2---	r-	90%	70%	/qfs4
mm	61	on	/dev/dsk/c5t13d0s0						
mr	62	on	/dev/dsk/c5t13d0s1						
ma	100	on	shareqfs1	m----	2c--	r-	80%	70%	/shareqfs1 spade
mm	101	on	/dev/dsk/c6t50020F2300004655d0s0						
mr	102	on	/dev/dsk/c6t50020F2300004655d0s1						
ma	110	on	shareqfs2	m----	2c--	r-	80%	70%	/shareqfs2 spade
mm	111	on	/dev/dsk/c6t50020F2300004655d0s6						
mr	112	on	/dev/dsk/c6t50020F2300004655d0s7						

Field Descriptions

TABLE 5-9 shows the field descriptions for this display.

TABLE 5-9 samu(1M) f Display Field Descriptions

Field	Description
ty	Device type.
eq	Equipment Ordinal of the device.
state	Current operating state of the device. Valid device states are as follows: <ul style="list-style-type: none">• on—The device is available for access.• ro—The device is available for read-only access.• off—The device is not available for access.• down—The device is available only for maintenance access.• idle—The device is not available for new operations. Operations in progress continue until completion.
device_name	File system name or path to the device.

TABLE 5-9 samu(1M) f Display Field Descriptions (*Continued*)

Field	Description
status	Device status. For a description of status codes, see “Operator Display Status Codes” on page 145.
high	High disk usage threshold percentage.
low	Low disk usage threshold percentage.
mountpoint	Mount point of the file system.
server	Name of the host system upon which the file system is mounted.

(F) - Optical Disk Label Display

The F display shows the label on an optical disk.

To invoke this display, type the following command:

```
Command: F
```

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

(h) - Help Display

The h display shows a summary of the samu(1M) displays available. By default, this is the first display that the system presents when you enter the samu(1M) command at the command line.

To invoke this display, type the following command:

```
Command: h
```

Navigation

TABLE 5-10 shows the control keys you can use in this display.

TABLE 5-10 Control Keys for the h Display

Key	Function
Ctrl-b	Page backward
Ctrl-d	Page forward (top portion)
Ctrl-f	Page forward
Ctrl-u	Page backward (top portion)
Ctrl-k	Toggle path display

Sample Display

There are several pages of help screens, but this manual shows only the first. Subsequent help screens show `samu(1M)` commands.

CODE EXAMPLE 5-17 shows the initial help screen for the Sun StorEdge SAM-FS file system.

CODE EXAMPLE 5-17 `samu(1M)` Initial Help Screen for a Sun StorEdge SAM-FS System

Help information	page 1/15	samu 4.4	08:18:13	Sept 8 2005
Displays:				
a	Archiver status	w	Pending stage queue	
c	Device configuration	C	Memory	
d	Daemon trace controls	F	Optical disk label	
f	File systems	I	Inode	
h	Help information	J	Preview shared memory	
l	License information	K	Kernel statistics	
m	Mass storage status	L	Shared memory tables	
n	Staging status	M	Shared memory	
o	Optical disk status	N	File system parameters	
p	Removable media load requests	P	Active Services	
r	Removable media	R	SAM-Remote	
s	Device status	S	Sector data	
t	Tape drive status	T	SCSI sense data	
u	Staging queue	U	Device table	
v	Robot catalog			
more (ctrl-f)				

(I) - Inode Display

The I display shows the content of inodes.

You can invoke this display differently, depending on what you need to view, as follows:

- To display inodes for an entire file system, type the command with the following format:

Command: **I** *filesystem*

For *filesystem*, specify the name of a file system.

- To display a specific inode, type the command with the following format:

Command: **I** *inode-number*

For *inode-number*, specify the inode number in either hexadecimal or decimal.

Navigation

TABLE 5-11 shows the control keys you can use in this display.

TABLE 5-11 Control Keys for the I Display

Key	Function
Ctrl-b	Previous inode
Ctrl-f	Next inode
Ctrl-k	Advance display format

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

Sample Display

CODE EXAMPLE 5-18 shows the inode display.

CODE EXAMPLE 5-18 samu(1M) I Display

Inode	0x1 (1)	format: file	samu 4.4	08:27:14	Sept 8 2005
	incore: y				
00008100	mode	-r-----	409cdf57	access_time	
00000001	ino	(1)	1d32ea20		
00000001	gen	(1)	4096b499	modify_time	
00000002	parent.ino	(2)	02588660		
00000002	parent.gen	(2)	4096b499	change_time	
00000000	size_u		02588660		
000c0000	size_l	(786432)	4096b443	creation_time	
01000000	rm:media/flags		409a8a7c	attribute_time	
00000000	rm:file_offset		409c0ce6	residence_time	
00000000	rm:mau		00000000	unit/cs/arch/flg	
00000000	rm:position		00000000	ar_flags	
00000000	ext_attrs	-----	00000000	stripe/stride/sg	
00000000	ext.ino	(0)	00000000	media -- --	
00000000	ext.gen	(0)	00000000	media -- --	
00000000	uid	root	00000000	psize (0)	
00000000	gid	root	000000c0	blocks (192)	
00000001	nlink	(1)	00000600	free_ino (1536)	
00011840	status	-n-----			
Extents (4k displayed as 1k):					
00_	000000d0.00	000000e0.00	000000f0.00	00000100.00	00000110.00 00000120.00
06_	00000130.00	00000140.00	00000150.00	00000160.00	00000170.00 00000180.00
12_	00000190.00	000001a0.00	000001b0.00	000001c0.00	00000630.00 00000000.00
18_	00000000.00				

(J) - Preview Shared Memory Display

The J display shows the shared memory segment for the preview queue.

To invoke this display, type the following command:

Command: **J**

Navigation

TABLE 5-12 shows the control keys you can use in this display.

TABLE 5-12 Control Keys for the J Display

Key	Function
Ctrl-b	Page backward
Ctrl-d	Half-page forward
Ctrl-f	Page forward
Ctrl-u	Half-page backward

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

Sample Display

CODE EXAMPLE 5-19 shows the preview shared memory display. This output has been truncated for inclusion in this manual.

CODE EXAMPLE 5-19 samu(1M) J Display

Preview shared memory size: 155648 samu 4.4 08:30:05 Sept 8 2005					
00000000	00040000	00014d58	00000000	00000000MX.....
00000010	00000000	00000000	73616d66	73202d20samfs -
00000020	70726576	69657720	6d656d6f	72792073	preview memory s
00000030	65676d65	6e740000	00026000	00000000	egment....`.....
00000040	00025fff	00000000	00040000	00014d58	.._.....MX
00000050	00000000	00000000	00000000	00000000
00000060	0000d9e0	00000064	00000000	000001b8	..Y'....d.....8
00000070	3f800000	447a0000	0000d820	00000008	?...Dz....X

(K) - Kernel Statistics Display

The K display shows kernel statistics, such as the number of inodes currently in memory.

To invoke this display, type the following command:

Command: **K**

Navigation

TABLE 5-13 shows the control keys you can use in this display.

TABLE 5-13 Control Keys for the K Display

Key	Function
Ctrl-b	Page backward
Ctrl-f	Page forward

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

Sample Display

CODE EXAMPLE 5-20 shows the kernel statistics display.

CODE EXAMPLE 5-20 samu(1M) K Display

Kernel statistics	samu 4.4 08:33:19 Sept 8 2005
module: sam-qfs name: general instance: 0 class: fs	
version	4.4.sam-qfs, gumball 2004-05-07 12:12:04
configured file systems	8
mounted file systems	8
nhino	16384
ninodes	129526
inocount	129527
inofree	128577

(1) - Usage Display

The 1 display shows the usage information for the file system, including the capacity and space used for each library and file system.

To invoke this display, type the following command:

Command: **1**

Sample Display

CODE EXAMPLE 5-21 shows an example of a usage display.

CODE EXAMPLE 5-21 samu(1M) l Display

Usage information		samu	4.4	08:36:27	Sept 8 2005
hostid = 80e69e6e OS name: SunOS Architecture: sparc CPUs: 2 (2 online)					
library 40: capacity	389.3G	bytes space	291.1G	bytes, usage	25%
library 51: capacity	9.5G	bytes space	9.5G	bytes, usage	0%
library 55: capacity	0	bytes space	0	bytes, usage	0%
library 56: capacity	10.7G	bytes space	10.7G	bytes, usage	0%
library totals: capacity	409.5G	bytes space	311.3G	bytes, usage	24%
filesystem samfs3: capacity	54.5M	bytes space	13.4M	bytes, usage	75%
filesystem samfs4: capacity	319.5M	bytes space	298.0M	bytes, usage	7%
filesystem samfs7: capacity	96.6M	bytes space	69.6M	bytes, usage	28%
filesystem samfs6: capacity	5.0G	bytes space	4.9G	bytes, usage	3%
filesystem samfs8: capacity	5.0G	bytes space	4.9G	bytes, usage	2%
filesystem totals: capacity	10.5G	bytes space	10.2G	bytes, usage	3%

Note – In versions of the software before 4U3, this display showed license information for the file system.

(L) - Shared Memory Tables

The L display shows the location of the shared memory tables. It also shows some system defaults that are kept in shared memory.

To invoke this display, type the following command:

```
Command: L
```

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

Sample Display

CODE EXAMPLE 5-22 shows the shared memory tables.

CODE EXAMPLE 5-22 samu(1M) L Display

Shared memory tables				samu 4.4 08:38:31 May 8 2005			
shm ptr tbl:				defaults:			
size	12000	(73728)		optical		mo	
left	44c8	(17608)		tape		lt	
scanner pid	1861			timeout		600	
fifo path	01b0	/var/opt/SUNWsamfs/previews				100	
dev_table	01cc			stages		1000	
first_dev	0450			log_facility		184	
scan_mess	cf50			dio minfilesize		100	
preview_shmid	1			label barcode		FALSE	
flags	0x20000000			barcodes low		FALSE	
preview stages	55776			export unavail		FALSE	
preview avail	100			attended		TRUE	
preview count	0			start rpc		FALSE	
preview sequence	445						
age factor	1			vsu factor		1000	
fs tbl ptr	0xd820			fs count		8	
fseq 10	samfs1	state 0	0 0	0	0		
fseq 20	samfs2	state 0	0 0	0	0		
fseq 30	qfs1	state 0	0 0	0	0		
fseq 40	qfs2	state 0	0 0	0	0		
fseq 50	qfs3	state 0	0 0	0	0		
fseq 60	qfs4	state 0	0 0	0	0		
fseq 100	shareqfs1	state 0	0 0	0	0	0	
fseq 110	shareqfs2	state 0	0 0	0	0	0	

(m) - Mass Storage Status Display

The **m** display shows the status of mass storage file systems and their member drives. This display shows only mounted file systems.

To invoke this display, type the following command:

Command:**m**

Sample Display

CODE EXAMPLE 5-23 shows the m display. Member drives are indented one space and appear directly below the file system to which they belong.

CODE EXAMPLE 5-23 samu(1M) m Display

Mass storage status				samu 4.4 08:41:11 Sept 8 2005						
ty	eq	status	use	state	ord	capacity	free	ra	part	high low
ms	10	m----2----	1%	on		68.354G	68.343G	1M	16	90% 70%
md	11		1%	on	0	13.669G	13.666G			
md	12		1%	on	1	13.669G	13.667G			
md	13		1%	on	2	13.669G	13.667G			
md	14		1%	on	3	13.674G	13.672G			
md	15		1%	on	4	13.674G	13.672G			
ms	20	m----2----	1%	on		68.354G	68.344G	1M	16	90% 70%
md	21		1%	on	0	13.669G	13.667G			
md	22		1%	on	1	13.669G	13.667G			
md	23		1%	on	2	13.669G	13.667G			
md	24		1%	on	3	13.674G	13.672G			
md	25		1%	on	4	13.674G	13.672G			
ma	30	m----2----	4%	on		64.351G	61.917G	1M	16	90% 70%
mm	31		1%	on	0	4.003G	3.988G	[8363840 inodes]		
md	32		4%	on	1	64.351G	61.917G			
ma	40	m----2----	1%	on		64.351G	64.333G	1M	16	90% 70%
mm	41		1%	on	0	4.003G	3.997G	[8382784 inodes]		
md	42		1%	on	1	64.351G	64.333G			
ma	50	m----2---r-	1%	on		64.351G	64.333G	1M	16	90% 70%
mm	51		1%	on	0	4.003G	3.997G	[8382784 inodes]		
mr	52		1%	on	1	64.351G	64.333G			
ma	60	m----2---r-	1%	on		64.351G	64.331G	1M	16	90% 70%
mm	61		1%	on	0	4.003G	3.997G	[8382784 inodes]		
mr	62		1%	on	1	64.351G	64.331G			
ma	100	m----2c--r-	2%	on		270.672G	265.105G	1M	16	80% 70%
mm	101		1%	on	0	2.000G	1.988G	[4168992 inodes]		
mr	102		2%	on	1	270.672G	265.469G			
ma	110	m----2c--r-	3%	on		270.656G	263.382G	1M	16	80% 70%
mm	111		1%	on	0	2.000G	1.987G	[4167616 inodes]		
mr	112		2%	on	1	270.656G	264.736G			

Field Descriptions

TABLE 5-14 shows the field descriptions for this display.

TABLE 5-14 samu(1M) m Display Field Descriptions

Field	Description
ty	Device type.
eq	Equipment Ordinal of the mass storage device.
status	Device status. For a description of status codes, see “Operator Display Status Codes” on page 145.
use	Percentage of disk space in use.
state	Current operating state of the mass storage device.
ord	Ordinal number of the disk device within the storage Family Set.
capacity	Number of 1024-byte blocks of usable space on the disk.
free	Number of 1024-byte blocks of disk space available.
ra	Readahead size in kilobytes.
part	Partial stage size in kilobytes.
high	High disk usage threshold percentage.
low	Low disk usage threshold percentage.

(M) - Shared Memory Display

The M display shows the raw shared memory segment in hexadecimal. This is a device table.

To invoke this display, type the following command:

Command:**M**

Navigation

TABLE 5-15 shows the control keys you can use in this display.

TABLE 5-15 Control Keys for the M Display

Key	Function
Ctrl-b	Page backward
Ctrl-d	Half-page forward
Ctrl-f	Page forward
Ctrl-u	Half-page backward

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

Sample Display

CODE EXAMPLE 5-24 shows the shared memory display. The output has been truncated for inclusion in this manual.

CODE EXAMPLE 5-24 samu(1M) M Display

Shared memory		size: 73728	samu 4.4 08:43:20 May 8 2005		
00000000	00040000	00014d58	00000000	00000000MX.....
00000010	00000000	00000000	73616d66	73202d20samfs -
00000020	73686172	6564206d	656d6f72	79207365	shared memory se
00000030	676d656e	74000000	00012000	000044c8	gment.....DH
00000040	0000dd20	00000000	00000742	00000745	..]B...E
00000050	00000001	00000000	00000000	c0000000@...
00000060	00000001	0001534d	00000000	00000000SM.....
00000070	00000000	00000000	00000000	00000000
00000080	00000000	00000000	00000000	00000000
00000090	20000000	000001b0	000001cc	000004500...L...P
000000a0	0000cf50	00000001	00000001	4c696365	..OP.....Lice
000000b0	6e73653a	204c6963	656e7365	206e6576	nse: License nev
000000c0	65722065	78706972	65732e00	00000000	er expires.....
000000d0	00000000	00000000	00000000	00000000
000000e0	00000000	00000000	00000000	00000000
000000f0	00000000	00000000	00000000	00000000

(n) - Staging Status Display

The `n` display shows the status of the stager for all media. It displays a list of outstanding stage requests.

You can invoke this display differently, depending on what you need to view, as follows:

- To display the staging status for all staging activity, type the command with the following format:

```
Command: n
```

- To display the staging status for a specific media type, type the command with the following format:

```
Command: n mt
```

For *mt*, specify one of the media types shown in the `mcf(4)` man page.

Sample Display

CODE EXAMPLE 5-25 shows the staging status display.

CODE EXAMPLE 5-25 samu(1M) n Display

```
Staging status                                samu 4.4 08:47:16 May  8 2005

Log output to: /var/opt/SUNWsamfs/stager/log

Stage request: dt.DAT001
Loading VSN DAT001

Staging queues
ty pid    user      status    wait files vsn
dt 16097  root      active    0:00     12 DAT001
```

(N) - File System Parameters Display

The N display shows all mount point parameters, the superblock version, and other file system information.

To invoke this display, type the following command:

```
Command: N
```

Navigation

TABLE 5-16 shows the control keys you can use in this display.

TABLE 5-16 Control Keys for the N Display

Key	Function
Ctrl-b	Previous file system
Ctrl-d	Page partitions forward
Ctrl-f	Next file system
Ctrl-i	Detailed status interpretations
Ctrl-u	Page partitions backward

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

Sample Display

CODE EXAMPLE 5-26 shows the file system parameters display.

CODE EXAMPLE 5-26 samu(1M) N Display

File system parameters		samu 4.4 08:55:19 Sept 8 2005	
mount_point	: /sam1	partial	: 16k
fs_type	: 6	maxpartial	: 16k
server	:	partial_stage	: 16384
filesystem name:	samfs1	flush_behind	: 0
eq_type	: 10 ms	stage_flush_beh:	0
state version	: 0 2	stage_n_window	: 262144
(fs,mm)_count	: 5 0	stage_retries	: 3
sync_meta	: 0	stage timeout	: 0
stripe	: 0	dio_consec r,w	: 0 0

CODE EXAMPLE 5-26 samu(1M) N Display (Continued)

mm_stripe	: 1	dio_frm_min r,w:	256 256
high low	: 90% 70%	dio_ill_min r,w:	0 0
readahead	: 1048576	ext_bsize	: 4096
writebehind	: 524288		
wr_throttle	: 16777216		
rd_ino_buf_size	: 16384		
wr_ino_buf_size	: 512		
config	: 0x08520530	mflag	: 0x00000044
status	: 0x00000001		
Device configuration:			
ty	eq state	device_name	fs family_set
md	11 on	/dev/dsk/c5t8d0s3	10 samfs1
md	12 on	/dev/dsk/c5t8d0s4	10 samfs1
md	13 on	/dev/dsk/c5t8d0s5	10 samfs1
md	14 on	/dev/dsk/c5t8d0s6	10 samfs1
md	15 on	/dev/dsk/c5t8d0s7	10 samfs1

(o) - Optical Disk Status Display

The o display shows the status of all optical disk drives configured within the environment.

To invoke this display, type the following command:

Command:**o**

Navigation

TABLE 5-17 shows the control keys you can use in this display.

TABLE 5-17 Control Keys for the o Display

Key	Function
Ctrl-b	Page backward
Ctrl-f	Page forward
Ctrl-b	Page backward
Ctrl-d	Half-page forward

TABLE 5-17 Control Keys for the o Display (Continued)

Key	Function
Ctrl-f	Page forward
Ctrl-k	Select (manual, automated library, both, priority)
Ctrl-u	Half-page backward

Sample Display

CODE EXAMPLE 5-27 shows the optical disk status display.

CODE EXAMPLE 5-27 samu(1M) o Display

Optical disk status				samu	4.4	Thu Oct 11 13:15:40
ty	eq	status	act	use	state	vsn
mo	35	--l---wo-r	1	29%	ready	oper2

Field Descriptions

TABLE 5-18 shows the field descriptions for this display.

TABLE 5-18 samu(1M) o Display Field Descriptions

Field	Description
ty	Device type.
eq	Equipment Ordinal of the optical disk.
status	Device status. For a description of status codes, see “Operator Display Status Codes” on page 145.
act	Activity count.

TABLE 5-18 samu(1M) o Display Field Descriptions (*Continued*)

Field	Description
use	Percentage of cartridge space used.
state	Current operating state of the optical disk. Valid device states are as follows: <ul style="list-style-type: none">• <code>ready</code>—The device is on, and the disk is loaded in the transport; available for access.• <code>notrdy</code>—The device is on, but no disk is present in the transport.• <code>idle</code>—The device is not available for new connections. Operations in progress continue until completion.• <code>off</code>—The device is not available for access.• <code>down</code>—The device is available only for maintenance access.
vsn	Volume serial name assigned to the optical disk, or the keyword <code>nolabel</code> if the volume is not labeled.

(p) - Removable Media Load Requests Display

The `p` display lists information about pending load requests for removable media. You can use the `mt` argument to select either a specific type of media, such as DLT tape, or a family of media, such as tape. The priority display lists the priority in the preview queue, rather than the user queue, and sorts the entries by priority.

It displays mount requests in the following formats:

- Both manual and automated library requests by user
- Both manual and automated library requests by priority
- Manual requests only
- Automated library requests only

You can invoke this display differently, depending on what you need to view, as follows:

- To display mount requests for all removable devices currently selected, type the command with the following format:

Command:**p**

- To display mount requests for devices of a given removable media type, type the command with the following format:

Command:**p** *mt*

For *mt*, specify one of the media types shown in the `mcf(4)` man page.

Navigation

TABLE 5-19 shows the control keys you can use in this display.

TABLE 5-19 Control Keys for the `p` Display

Key	Function
Ctrl-b	Page backward
Ctrl-d	Half-page forward
Ctrl-f	Page forward
Ctrl-k	Toggle between the different display formats
Ctrl-u	Half-page backward

Sample Display

CODE EXAMPLE 5-28 shows the removable media load requests display.

CODE EXAMPLE 5-28 `samu(1M)` `p` Display

Removable media load requests all both samu 4.4 09:14:19 Sept 8 2005							
count: 1							
index	type	pid	user	rb	flags	wait count	vsn
0	dt	15533	root	150	W--f---	0:00	DAT001

Field Descriptions

TABLE 5-20 shows the field descriptions for this display.

TABLE 5-20 `samu(1M)` `p` Display Field Descriptions

Field	Description
index	Index number in the preview table.
type	Device type code assigned to the removable media.
pid	UNIX process identifier. A process identifier of 1 indicates NFS access.
user	Name assigned to the user requesting the load.
priority	Priority of the request.

TABLE 5-20 samu(1M) p Display Field Descriptions (*Continued*)

Field	Description
rb	Equipment Ordinal of the automated library in which the requested VSN resides.
flags	Flags for the device. See TABLE 5-21.
wait	The elapsed time since the mount request was received.
count	The number of requests for this VSN, if it is a stage.
vsn	Volume serial name of the volume.

Flags

TABLE 5-21 shows the flags for the p display.

TABLE 5-21 Flags Field for the samu(1M) p Display

Field	Description
W-----	Write access requested.
-b-----	Entry is busy.
--C----	Clear VSN requested.
---f---	File system requested.
----N--	Media is foreign to the file system.
-----S-	Flip side already mounted.
-----s	Stage request flag.

(P) - Active Services Display

The P display lists the services registered with the Sun StorEdge SAM-FS single port multiplexer.

To invoke this display, type the following command:

Command:**P**

Navigation

TABLE 5-22 shows the control keys you can use in this display.

TABLE 5-22 Control Keys for the P Display

Key	Function
Ctrl-b	Page backward
Ctrl-f	Page forward

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

Sample Display

CODE EXAMPLE 5-29 shows the active services display.

CODE EXAMPLE 5-29 samu(1M) P Display

Active Services	samu	4.4 09:08:33 Sept 8 2005
Registered services for host 'pup':		
sharedfs.qfs2		
sharedfs.qfs1		
2 service(s) registered.		

(r) - Removable Media Status Display

The `r` display enables you to monitor the activity on removable media devices such as tape drives. You can monitor either a specific type of device, such as video tape, or a family of devices such as all tape devices.

You can invoke this display differently, depending on what you need to view, as follows:

- To display the status for all removable media devices, type the command with the following format:

```
Command: r
```

- To display the status for a specific device, type the command with the following format:

Command:**r** *eq*

For *eq*, specify the Equipment Ordinal for the device.

Sample Display

CODE EXAMPLE 5-30 shows the removable media status display.

CODE EXAMPLE 5-30 samu(1M) r Display

Removable media status: all					samu 4.4 09:11:27 Sept 8 2005	
ty	eq	status	act	use	state	vsn
dt	150	--1-----r	0	63%	ready	DAT001

Field Descriptions

TABLE 5-23 shows the field descriptions for this display.

TABLE 5-23 samu(1M) r Display Field Descriptions

Field	Description
ty	Device type.
eq	Equipment Ordinal of the drive.
status	Device status. For a description of status codes, see “Operator Display Status Codes” on page 145.
act	Activity count.
use	Percentage of cartridge space used.

TABLE 5-23 samu(1M) r Display Field Descriptions (*Continued*)

Field	Description
state	Current operating state of the removable media. Valid device states are as follows: <ul style="list-style-type: none">• ready—The device is on, and the disk or tape is loaded in the transport; available for access.• notrdy—The device is on, but no disk or tape is present in the transport.• idle—The device is not available for new connections. Operations in progress continue until completion.• off—The device is not available for access.• down—The device is available only for maintenance access.
vsn	Volume serial name assigned to the volume, or the keyword nolabel if the volume is not labeled. Blank if no volume is present in the transport, or device is off.

(R) - Sun SAM-Remote Information Display

The **R** display shows information and status on Sun SAM-Remote configurations.

To invoke this display, type the following command:

```
Command: R
```

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

(s) - Device Status Display

The **s** display shows the status for all devices configured within the environment.

To invoke this display, type the following command:

```
Command: s
```

Navigation

TABLE 5-24 shows the control keys you can use in this display.

TABLE 5-24 Control Keys for the `s` Display

Key	Function
Ctrl-b	Page backward
Ctrl-d	Half-page forward
Ctrl-f	Page forward
Ctrl-u	Half-page backward

Sample Display

CODE EXAMPLE 5-31 shows the device status display.

CODE EXAMPLE 5-31 `samu(1M) s` Display

Device status		samu	4.4 09:14:05 Sept 8 2005
ty	eq state	device_name	fs status pos
sk	100 on	/etc/opt/SUNWsamfs/dcstkconf	100 m-----r
	stk_dismount(2275) 0, volser 700073		
sg	120 on	/dev/rmt/2cbn	100 -----p
	empty		
sg	130 on	/dev/rmt/5cbn	100 --l----o-r
	Ready for data transfer		
sg	140 on	/dev/rmt/6cbn	100 -----p
	empty		
sg	150 on	/dev/rmt/4cbn	100 -----p
	empty		
hy	151 on	historian	151 -----

Field Descriptions

TABLE 5-25 shows the field descriptions for this display.

TABLE 5-25 samu(1M) s Display Field Descriptions

Field	Description
ty	Device type.
eq	Equipment ordinal of the device.
state	Current operating state of the device.
device_name	Path to the device. For file system devices, this is the file system name.
fs	Equipment Ordinal of the family, set to which the device belongs.
status	Device status. For a description of status codes, see “Operator Display Status Codes” on page 145.

(S) - Sector Data Display

The S display shows raw device data.

To invoke this display, type the following command:

```
Command: S
```

Navigation

TABLE 5-26 shows the control keys you can use in this display.

TABLE 5-26 Control Keys for the S Display

Key	Function
Ctrl-b	Previous sector
Ctrl-d	Page forward (top portion)
Ctrl-f	Next sector
Ctrl-k	Advance display format
Ctrl-u	Page backward (top portion)

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

(t) - Tape Drive Status Display

The `t` display shows the status of all tape drives configured within the environment.

To invoke this display, type the following command:

```
Command: t
```

Navigation

TABLE 5-27 shows the control keys you can use in this display.

TABLE 5-27 Control Keys for the `t` Display

Key	Function
Ctrl-b	Page backward
Ctrl-f	Page forward

Sample Display

CODE EXAMPLE 5-32 shows the tape drive status display.

CODE EXAMPLE 5-32 samu(1M) `t` Display

Tape drive status						samu	4.4 09:21:07 Sept 8 2005
ty	eq	status	act	use	state	vsn	
sg	120	-----p	0	0%	notrdy		
		empty					
sg	130	-----p	0	0%	notrdy		
		empty					
sg	140	-----p	0	0%	notrdy		
		empty					
sg	150	--l-----r	0	41%	ready	700088	
		idle					

Field Descriptions

TABLE 5-28 shows the field descriptions for this display.

TABLE 5-28 samu(1M) t Display Field Descriptions

Field	Description
ty	Device type.
eq	Equipment Ordinal of the drive.
status	Device status. For a description of status codes, see “Operator Display Status Codes” on page 145.
act	Activity count.
use	Percentage of cartridge space used.
state	Current operating state of the removable media. Valid device states are as follows: <ul style="list-style-type: none">• <code>ready</code>—The device is on and the disk or tape is loaded in the transport; available for access.• <code>notrdy</code>—The device is on but no disk or tape is present in the transport.• <code>idle</code>—The device is not available for new connections. Operations in progress continue until completion.• <code>off</code>—The device is not available for access.• <code>down</code>—The device is available only for maintenance access.
vsn	Volume serial name assigned to the volume, or the keyword <code>noLabel</code> if volume is not labeled. Blank if no volume is present in the transport, or device is off.

(T) - SCSI Sense Data Display

The T display shows the SCSI status of a SCSI device.

To invoke this display, type the following command:

Command: **T**

Navigation

TABLE 5-29 shows the control keys you can use in this display.

TABLE 5-29 Control Keys for the T Display

Key	Function
Ctrl-b	Previous equipment
Ctrl-f	Next equipment

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

(u) - Staging Queue Display

The u display lists all files in the staging queue.

To invoke this display, type the following command:

```
Command:u
```

Navigation

TABLE 5-30 shows the control keys you can use in this display.

TABLE 5-30 Control Keys for the u Display

Key	Function
Ctrl-b	Page backward
Ctrl-d	Half-page forward
Ctrl-f	Page forward
Ctrl-k	Display the path on the second line of each entry
Ctrl-u	Half-page backward

Sample Display

CODE EXAMPLE 5-33 shows the staging queue display.

CODE EXAMPLE 5-33 samu(1M) u Display

Staging queue by media type: all						samu 4.4 09:24:23 Sept 8 2005
volumes 1 files 22						
ty	length	fseq	ino	position	offset	vsn
dt	451.611k	20	1030	207cc	473	DAT001
dt	341.676k	20	1031	207cc	7fc	DAT001
dt	419.861k	20	1032	207cc	aa9	DAT001
dt	384.760k	20	1033	207cc	df2	DAT001
dt	263.475k	20	1034	207cc	10f5	DAT001
dt	452.901k	20	1035	207cc	1305	DAT001
dt	404.598k	20	1036	207cc	1690	DAT001
dt	292.454k	20	1037	207cc	19bb	DAT001
dt	257.835k	20	1038	207cc	1c05	DAT001
dt	399.882k	20	1040	207cc	1e0b	DAT001
dt	399.882k	40	1029	208d7	2	DAT001
dt	257.835k	40	1030	208d7	323	DAT001
dt	292.454k	40	1031	208d7	528	DAT001
dt	404.598k	40	1032	208d7	772	DAT001
dt	452.901k	40	1033	208d7	a9d	DAT001
dt	263.475k	40	1034	208d7	e28	DAT001
dt	384.760k	40	1035	208d7	1038	DAT001
dt	419.861k	40	1036	208d7	133b	DAT001
dt	341.676k	40	1037	208d7	1684	DAT001
dt	451.611k	40	1038	208d7	1931	DAT001
dt	161.326k	40	1039	208d7	1cba	DAT001
dt	406.400k	40	1040	208d7	1dfe	DAT001

Field Descriptions

TABLE 5-31 shows the field descriptions for this display.

TABLE 5-31 samu(1M) u Display Field Descriptions

Field	Description
ty	Device type.
length	File length.
fseq	File system equipment number.
ino	The inode number.

TABLE 5-31 samu(1M) u Display Field Descriptions (*Continued*)

Field	Description
position	The position of the archive file on the specific medium.
offset	Offset of the archive file on the specific medium.
vsn	Volume serial name of the volume.

(U) - Device Table Display

The U display shows the device table in a human-readable form.

You can invoke this display differently, depending on what you need to view, as follows:

- To display the device table for all devices, type the command with the following format:

Command:**U**

- To display the device table for a specific device, type the command with the following format:

Command:**U** *eq*

For *eq*, specify the Equipment Ordinal of the device.

Navigation

TABLE 5-32 shows the control keys you can use in this display.

TABLE 5-32 Control Keys for the U Display

Key	Function
Ctrl-b	Previous equipment
Ctrl-f	Next equipment

This display is designed for debugging. It is intended to be used only with the assistance of a Sun Microsystems support staff person.

Sample Display

CODE EXAMPLE 5-34 shows the device table display.

CODE EXAMPLE 5-34 samu(1M) U Display

```
Device table: eq: 10      addr: 00000450  samu 4.4 09:28:40 Sept 8 2005

message:

0004000000014d58 000000000000000000      00000000 delay
0000000000000000 mutex                    00000000 unload_delay
00000aa8 next
73616d66 set:  samfs1
73310000
00000000
00000000
000a000a eq/fseq
08010801 type/equ_type
0000      state
00000000 st_rdev
00000000 ord/model
00000000 mode_sense
00000000 sense
00000000 space
00000000 capacity
00000000 active
00000000 open
00000000 sector_size
00000000 label_address
00000000 vsn:
00000000
00000000
00000000
00000000 status: -----
00000000 dt
73616d66 name: samfs1
```

(v) - Automated Library Catalog Display

The v display shows the location and VSN of all disks or tapes currently cataloged in the automated library.

You can invoke this display differently, depending on what you need to view, as follows:

- To display the catalog for all devices, type the command with the following format:

```
Command: v
```

- To display catalog information for a specific device, type the command with the following format:

```
Command: v eq
```

For *eq*, specify the Equipment Ordinal of the device. Type the keyword *historian* to view the historian catalog.

At certain times, *samu(1M)* prompts for a device to be entered, as follows:

```
Enter robot: eq
```

For *eq*, specify the Equipment Ordinal of the device or press return. Pressing return displays information for the previous device specified.

For a list of all device names and Equipment Ordinals, see “(c) - Device Configuration Display” on page 104.

Navigation

TABLE 5-33 shows the control keys you can use in this display.

TABLE 5-33 Control Keys for the *v* Display

Key	Function
Ctrl-b	Page backward.
Ctrl-d	Next library catalog.
Ctrl-f	Page forward.
Ctrl-i	Detailed, 2-line display format. When you enter Ctrl-i once, it shows times and barcodes. When you enter Ctrl-i a second time, it shows volume reservations on the second line.
Ctrl-k	Advance sort key. After you enter Ctrl-k, you can enter one of the following to select a sort key: <ul style="list-style-type: none"> 1 - sort by slot. 2 - sort by count.

TABLE 5-33 Control Keys for the v Display *(Continued)*

Key	Function
	3 - sort by usage.
	4 - sort by VSN.
	5 - sort by access time.
	6 - sort by barcode.
	7 - sort by label time.
Ctrl-u	Previous automated library catalog.
/	Search for VSN.
%	Search for barcode.
\$	Search for slot.

Sample Display

CODE EXAMPLE 5-35 shows the automated library catalog display.

CODE EXAMPLE 5-35 samu(1M) v Display

Robot VSN catalog by slot : eq 100samu						4.4 09:30:25 Sept 8 2005
count 32						
slot	access time	count	use	flags	ty	vsn
0	2004/05/08 08:35	64	0%	-il-o-b-----	sg	700071
1	2004/05/08 09:08	27	12%	-il-o-b-----	sg	700073
2	2004/05/08 09:12	26	12%	-il-o-b-----	sg	700077
3	2004/05/08 08:39	37	40%	-il-o-b-----	sg	700079
4	2004/05/08 09:16	24	6%	-il-o-b-----	sg	700084
5	2004/05/08 09:18	24	41%	-il-o-b-----	sg	700088
6	none	0	0%	-il-o-b-----	sg	700090
7	none	0	0%	-il-o-b-----	sg	700092
8	none	0	0%	-il-o-b-----	sg	000155
9	none	0	0%	-il-o-b-----	sg	000156
10	none	0	0%	-il-o-b-----	sg	000157
11	none	0	0%	-il-o-b-----	sg	000158
12	none	0	0%	-il-o-b-----	sg	000154
13	none	0	0%	-il-o-b-----	sg	000153
14	none	0	0%	-il-o-b-----	sg	000152

Field Descriptions

TABLE 5-34 shows the field descriptions for this display.

TABLE 5-34 samu(1M) v Display Field Descriptions

Field	Description
Robot VSN catalog	Name of the specified automated library and time the display refreshed.
count	Number of slots allocated in this library's catalog.
slot	Slot number within the specified library.
access time	Time the volume was last accessed.
count	Number of accesses to this volume since the last audit operation.
use	Percentage of space used for the volume.
flags	Flags for the device. See TABLE 5-35 for information about the flags.
ty	Device type.
vsn	Volume serial name of the volume.

Flags

TABLE 5-35 shows the flags from the flags field in TABLE 5-34. In some cases, more than one flag can occur in a field, and one flag overrides the other.

TABLE 5-35 Flags Field for samu(1M) v Display

Flags	Description
A-----	Volume needs audit.
-i-----	Slot in use.
--l-----	Labeled. Overrides N.
--N-----	Unlabeled. This volume is foreign to the environment.
---E-----	Media error. Set when the software detects a write error on a cartridge.
----o-----	Slot occupied.
----C-----	Volume is a cleaning tape. Overrides p.
----p-----	Priority VSN.
-----b-----	Barcode detected.

TABLE 5-35 Flags Field for `samu(1M) v Display` (Continued)

Flags	Description
-----W----	Write protect. Set when the physical write protection mechanism is enabled on a cartridge.
-----R---	Read only.
-----c--	Recycle.
-----d-	Duplicate VSN. Overrides U.
-----U-	Volume unavailable.
-----f	Archiver found volume full.
-----X	Export slot.

(w) - Pending Stage Queue

The `w` display shows queued stage requests for which the volumes have not yet been loaded.

You can invoke this display differently, depending on what you need to view, as follows:

- To display the pending stage queue for all media, type the command with the following format:

```
Command: w
```

- To display the pending stage queue for a specific media type, type the command with the following format:

```
Command: w mt
```

For `mt`, specify one of the media types shown in the `mcf(4)` man page.

Navigation

TABLE 5-36 shows the control keys you can use in this display.

TABLE 5-36 Control Keys for the w Display

Key	Function
Ctrl-b	Page backward
Ctrl-d	Half-page forward
Ctrl-f	Page forward
Ctrl-k	Display the path on the second line of each entry
Ctrl-u	Half-page backward

Sample Display

CODE EXAMPLE 5-36 shows the pending stage queue.

CODE EXAMPLE 5-36 samu(1M) w Display

Pending stage queue by media type: all							samu	4.4	Thu Oct 11 13:20:27
volumes 1 files 13									
ty	length	fseq	ino	position	offset	vsn			
at	1.383M	1	42	3a786	271b	000002			
at	1.479M	1	56	3a786	5139	000002			
at	1018.406k	1	60	3a786	6550	000002			
at	1.000M	1	65	3a786	7475	000002			
at	1.528M	1	80	3a786	99be	000002			
at	1.763M	1	92	3a786	ce57	000002			
at	1.749M	1	123	3a786	11ece	000002			
at	556.559k	1	157	3a786	1532f	000002			
at	658.970k	1	186	3a786	17705	000002			
at	863.380k	1	251	3a786	1dd58	000002			
at	1.268M	1	281	3a786	1f2b7	000002			
at	1.797M	1	324	3a786	23dfa	000002			
at	1.144M	1	401	3a786	2bb6d	000002			

Field Descriptions

TABLE 5-37 shows the field descriptions for this display.

TABLE 5-37 samu(1M) w Display Field Descriptions

Field	Description
ty	Device type.
length	File length.
fseq	File system Equipment Ordinal.
ino	The inode number.
position	The position (in decimal format) of the archive file on the specific medium.
offset	Offset of the archive file on the specific medium.
vsn	Volume serial name of the volume.

Operator Display Status Codes

The operator displays have different status codes for the removable media device displays and the file system displays. The following sections describe these status codes.

Removable Media Device Display Status Codes

The *o*, *r*, *s*, and *t* operator displays show status codes for removable media devices. Status codes are displayed in a 10-position format, reading from left (position 1) to right (position 10).

The status codes in this section do not apply to the *samu(1M)* *f*, *m*, and *v* displays. For information about the status codes for the *f* and *m* displays, see “File System Display Status Codes” on page 146. For information about the status codes for the *v* display, see “(v) - Automated Library Catalog Display” on page 139.

TABLE 5-38 defines the valid status codes for each position.

TABLE 5-38 Removable Media Device Display Status Codes

Status Bit	Meaning for a Device
s-----	Media is being scanned.
m-----	The automated library is operational.
M-----	Maintenance mode.

TABLE 5-38 Removable Media Device Display Status Codes (*Continued*)

Status Bit	Meaning for a Device
-E-----	Device received an unrecoverable error in scanning.
-a-----	Device is in audit mode.
--l-----	Media has a label.
--N-----	Foreign media.
--L-----	Media is being labeled.
---I-----	Waiting for device to idle.
---A-----	Needs operator attention.
----C-----	Needs cleaning.
----U-----	Unload has been requested.
-----R----	Device is reserved.
-----w----	A process is writing on the media.
-----o--	Device is open.
-----P-	Device is positioning (tape only).
-----F-	For automated libraries, all storage slots occupied. For tape and magneto-optical drives, media is full.
-----R	Device is ready and the media is read-only.
-----r	Device is spun up and ready.
-----p	Device is present.
-----W	Device is write protected.

File System Display Status Codes

The `f` and `m` operator displays show status codes for file systems. Status codes are displayed in an 11-position format, reading from left (position 1) to right (position 11).

The status codes in this section do not apply to the `samu(1M)` `c`, `o`, `r`, `s`, `t`, or `v` displays. For information about the status codes for the `c`, `o`, `r`, `s`, and `t` displays, see “Removable Media Device Display Status Codes” on page 145. For information about the status codes for the `v` display, see “(v) - Automated Library Catalog Display” on page 139.

TABLE 5-39 defines the valid status codes for each position.

TABLE 5-39 File System Display Status Codes

Status Bit	Meaning for a File System
m-----	File system is currently mounted.
M-----	File system is being mounted.
-u-----	File system is being unmounted.
--A-----	File system data is being archived.
---R-----	File system data is being released.
----S-----	File system data is being staged.
-----1-----	Sun StorEdge SAM-FS file system version 1.
-----2-----	Sun StorEdge SAM-FS file system version 2.
-----c-----	Sun StorEdge QFS shared file system.
-----W---	Single writer.
-----R--	Multireader.
-----r-	mr devices.
-----d	md devices.

Operator Display Device States

The `c`, `m`, `o`, `r`, `s`, and `t` operator displays show device state codes. These codes represent the current access state for the device. TABLE 5-40 defines the valid state codes.

TABLE 5-40 Operator Display Device States

Device State	Description
on	The device is available for access. For certain displays, this state might be superseded by the states <code>ready</code> or <code>notrdy</code> .
ro	The device is available for read-only access. For certain displays, this state might be superseded by the states <code>ready</code> or <code>notrdy</code> .

TABLE 5-40 Operator Display Device States (*Continued*)

Device State	Description
<code>off</code>	<p>The device is not available for access. For tape and optical disk drives, possible reasons for the device to be in the <code>off</code> state include the following:</p> <ul style="list-style-type: none"> • Cleaning was requested, but no cleaning cartridge was found in the automated library. • The cleaning cartridge cannot be loaded or unloaded from the drive. • Initialization found the drive status to be full, and attempts to clear the drive failed. • The system was unable to clear a cartridge from a drive. • Opening the drive for I/O failed during spin-up. • An error other than <code>NOT READY</code> was received when spinning down the drive for unloading. • Opening the standard tape driver on the drive failed during spin-up.
<code>down</code>	The device is available for maintenance access only.
<code>idle</code>	The device is not available for new connections. Operations in progress continue until completion.
<code>ready</code>	The device is on and the disk or tape loaded in the transport is available for access.
<code>notrdy</code>	The device is on, but no disk or tape is present in the transport.
<code>unavail</code>	The device is unavailable for access and cannot be used for automatic operations. You can continue to use the <code>load(1M)</code> and <code>unload(1M)</code> commands for moving media while the device is in the <code>unavail</code> state.

You can use the `samu(1M)` `down`, `off`, and `on` device state commands to change device states to `down`, `off`, or `on`. You can enter these commands from any `samu(1M)` display, but if you enter them from the `c`, `m`, `o`, `r`, `s`, or `t` display, you can see the device state change in the display. For example, you could set a device state to `off` from within the `P` display, but you would not be able to see the new device state reflected in the display.

The following procedures show what to type to change a device's state from `down` to `on` and from `on` to `down`.

▼ To Change a Drive State from `down` to `on`

1. **Bring up a `samu(1M)` display that shows drive and automated library device states.**

The following `samu(1M)` displays all show device states: `c`, `m`, `o`, `r`, `s`, and `t`.

2. **Visually inspect the display to verify that the device is in the `down` state.**

3. **Type** :off.

Turning the device off halts all activity so the device can be started cleanly in the next step. For example:

```
Command:off eq
```

For *eq*, specify the Equipment Ordinal of the device.

4. **Type** :on.

For example:

```
Command:on eq
```

For *eq*, specify the Equipment Ordinal of the device.

▼ To Change a Drive State from on to down

1. **Bring up a samu(1M) display that shows drive and automated library device states.**

The following samu(1M) displays all show device states: c, m, o, r, s, and t.

2. **Visually inspect the display to verify that the device is in the on state.**

3. **Type** :off.

Turning the device off halts all activity so the device can be stopped cleanly in the next step. For example:

```
Command:off eq
```

For *eq*, specify the Equipment Ordinal of the device.

4. **Type** :down.

For example:

```
Command:down eq
```

For *eq*, specify the Equipment Ordinal of the device.

Operator Commands

The following sections describe the operator commands that you can enter from the `samu(1M)` operator utility's command interface. You can enter the commands from any display.

The following types of operator commands are available:

- "Device Commands" on page 150
- "Sun StorEdge SAM-FS Commands: Archiver Control" on page 151
- "Sun StorEdge SAM-FS Commands: Releaser Control" on page 154
- "Sun StorEdge SAM-FS Commands: Stager Control" on page 155
- "File System Commands: I/O Management" on page 156
- "File System Commands: Direct I/O Management" on page 158
- "File System Commands: Miscellaneous" on page 160
- "Automated Library Commands" on page 163
- "Miscellaneous Commands" on page 165

If you want to enter any operator commands from the Solaris operating system (OS) command line, you must use them as arguments to the `samcmd(1M)` command. For more information about the `samcmd(1M)` command, see the `samcmd(1M)` man page.

In the following subsections, each `samu(1M)` command is prefaced with a colon (:) when it is entered to designate that a command is being entered and not a series of hot keys.

Device Commands

TABLE 5-41 shows the device commands and their actions.

TABLE 5-41 Device Command Actions

Command	Action
<code>down</code>	Terminates operation on device <i>eq</i> .
<code>idle</code>	Restricts access to device <i>eq</i> by preventing new connections to the device. Existing operations continue until completion.
<code>off</code>	Logically turns off device <i>eq</i> .

TABLE 5-41 Device Command Actions (*Continued*)

Command	Action
on	Logically turns on device <i>eq</i> .
unavail	Selects device <i>eq</i> and makes it unavailable for use with the file system. You might set a drive state to <i>unavail</i> , for example, in a disaster recovery situation in which you are trying to load media to restore a file system and you do not want the Sun StorEdge SAM-FS software to attempt to use this drive.
unload	Unloads the mounted media for the specified removable media device <i>eq</i> . For magazine devices, the <i>unload</i> command unloads the mounted cartridge and ejects the magazine.

All of these commands are used in the following format: *:command eq*. For *eq*, specify the Equipment Ordinal of the device.

Sun StorEdge SAM-FS Commands: Archiver Control

TABLE 5-42 shows the archiver commands and their actions.

TABLE 5-42 Archiver Command Actions

Command	Action
aridle	Stops all archiving at the next convenient point. For example, at the end of the current <i>tar(1)</i> file for <i>sam-arcopy</i> operations. You can also use this command to stop all archiving activity for all file systems before unmounting the file systems.
arrerun	Performs a soft restart on the archiver. The archiver daemons are restarted, and all work in progress is recovered.
arrestart	Interrupts the archiver and restarts the archiver. This action occurs regardless of the state of the archiver. Therefore, use <i>arrestart</i> with caution. Some copy operations to archive media might not complete and must be repeated. This wastes space on the media.
armarchreq	Removes an archive request.
arrun	Causes the archiver to begin archiving. This command overrides any existing global <i>wait</i> command in the <i>archiver.cmd</i> file.
arscan	Scans the file system.
arstop	Stops all archiving immediately.
artrace	Performs archiver tracing.

CODE EXAMPLE 5-37 shows the formats for the archiver commands.

CODE EXAMPLE 5-37 Formats for the Archiver Commands

```
:aridle [ dk | rm | fs.fsname ]
:arrerun
:arrestart
:armarchreq fsname.[* | archreq]
:arrun [ dk | rm | fs.fsname ]
:arscan fsname[.dir | ..inodes][int]
:arstop [ dk | rm | fs.fsname ]
:artrace [fs.fsname]
```

The arguments to these commands are optional. If no arguments are specified, all file systems are affected. If arguments are specified, the command takes action based on the type of archive file specified (*dk* or *rm*) or the file system specified. TABLE 5-43 shows the archiver command arguments.

TABLE 5-43 Archiver Command Arguments

Argument	Description
<i>dk</i>	Specifies that this command pertains to disk archive files.
<i>rm</i>	Specifies that this command pertains to removable media files.
<i>fsname</i>	Specifies that this command pertains to a specific file system. Enter a file system name for <i>fsname</i> .
<i>archreq</i>	<p>Specifies the name of a specific archive request file in the following format:</p> <p><i>arset.copy.seq_num</i></p> <p>This file name has three components. Use a period to separate each component. The first component is the name of the archive set. The second component is the copy number (1, 2, 3, or 4). The third component is a sequence number that the archiver assigns. More than one archive request can exist at one time.</p> <p>You can use the <code>showqueue(1M)</code> command to obtain the names of the archive request files in the system. CODE EXAMPLE 5-38 shows how to use this command. Alternatively, you can go to the following directory and list the files present:</p> <p><code>/var/opt/SUNWsamfs/archiver/<i>fsname</i>/ArchReq</code></p>
*	Signifies all files.
<i>dir</i>	Specifies a specific directory name. This is the directory to scan.
<i>.inodes</i>	Specifies that the inodes should be scanned.
<i>int</i>	An integer number of seconds to delay the scan.

CODE EXAMPLE 5-38 shows using the `showqueue(1M)` command to obtain an *archreq* file name that can be used as input to the `armarchreq samu(1M)` command.

CODE EXAMPLE 5-38 Using `showqueue(1M)`

```
# showqueue samfs9
Filesystem samfs9:
Scan list: empty
Archive requests
arset1.2.0 schedule 2005-01-22 16:23:07
    files:697 space:    4.934G flags: offline
(min:    1.000k)    priority: 0 0
    No volumes available
    Drive 1
        Files: 695, bytes:    1.932G (min:    1.000k)
Stage volumes:
    lt.CFX600
    lt.CFX601

arset1.1.1 schedule 2005-01-22 16:23:07
    files:3 space:    6.236M flags:
(min:  826.000k)    priority: 0 0
    No volumes available
    Drive 1
        Files: 3, bytes:    6.236M (min:  826.000k)
```

CODE EXAMPLE 5-38 shows that files `arset1.2.0` and `arset1.1.1` are archive request files.

The `:hwm_archive eq` and `:nohwm_archive eq` Commands

The `hwm_archive` command invokes the archiver when the amount of data in the file system increases to a level above the releaser's high-water mark. You can set the releaser's high-water mark by using the `thresh` command. For information about the `thresh` command, see "The `:thresh eq high low` Command" on page 153. The `nohwm_archive` command disables this capability and is the default.

For *eq*, specify the Equipment Ordinal for the file system.

The `:thresh eq high low` Command

The `thresh` command sets the high and low thresholds for a file system to control file archiving.

For *eq*, specify the Equipment Ordinal of the storage family set.

For *high*, specify the high threshold.

For *low*, specify the low threshold.

For example, the following command sets the high threshold to 50 percent and the low threshold to 40 percent for the storage Family Set whose file system Equipment Ordinal is 10:

```
:thresh 10 50 40
```

Sun StorEdge SAM-FS Commands: Releaser Control

The following commands enable you to control aspects of the partial release feature. For more information about the partial release feature, see the Releasing chapter in the *Sun StorEdge SAM-FS Storage and Archive Management Guide*.

The `:maxpartial eq value` Command

The `maxpartial` command sets the maximum partial release size for the file system to *value* kilobytes. The partial release size cannot be set larger than this `maxpartial` setting.

For *eq*, specify the Equipment Ordinal for the file system.

For *value*, specify an integer such that $0 \leq \text{value} \leq 2097152$.

The `:partial eq value` Command

The `partial` command sets the number of kilobytes to leave online after release of the file. For more information, see the Releasing chapter in the *Sun StorEdge SAM-FS Storage and Archive Management Guide*.

For *eq*, specify the Equipment Ordinal for the file system.

For *value*, specify the number of kilobytes to leave online. The default size is 16.

Sun StorEdge SAM-FS Commands: Stager Control

The following commands enable you to control staging activities.

The `:partial_stage eq value` Command

The `partial_stage` command sets the partial stage size for the file system to *value* kilobytes. For a file with the partial release attribute, *value* specifies the file offset past which access results in the entire file being staged to disk.

For *eq*, specify the Equipment Ordinal for the file system.

For *value*, specify an integer that is greater than 0 but is less than the *value* specified for the `maxpartial` setting. For more information about the `maxpartial` setting, see “The `:maxpartial eq value` Command” on page 154. For more information about the partial release feature, see the Releasing chapter in the *Sun StorEdge SAM-FS Storage and Archive Management Guide*.

The `:stage_flush_behind eq value` Command

The `stage_flush_behind` command sets the maximum stage flush-behind value. Pages being staged are written to disk asynchronously to help the Solaris VM layer keep the pages clean.

For *eq*, specify the Equipment Ordinal for the file system.

For *value*, specify an integer number of kilobytes such that $0 \leq \textit{value} \leq 8192$. By default, *value*=0, which disables `stage_flush_behind`.

The `:stage_n_window eq value` Command

The `stage_n_window` command works with the `stage(1)` command’s `-n` option. This `samu(1M)` command sets the `stage(1)` command’s `-n` option for the file system to *value*. This command is effective for files read directly from the archive media and for which `stage -n` has been specified. For a file with the `stage -n` attribute set, *value* is the amount of data that is staged to the application’s buffer at any one time.

For *eq*, specify the Equipment Ordinal for the file system.

For *value*, specify an integer such that $64 \leq \textit{value} \leq 2097152$ kilobytes. The default is 256 for all file systems except for the Sun StorEdge QFS shared file system, which is set to the value of the `minallocsz` mount option.

The `:stage_retries` *eq value* Command

The `stage_retries` command sets the number of stage retries attempted per archive copy when certain errors are encountered.

For *eq*, specify the Equipment Ordinal for the file system.

For *value*, specify a number such that $0 \leq \text{value} \leq 20$. When *value*=0, retries are not attempted. The default is 3.

The `:stclear` *mt.vsn* Command

The `stclear` command clears a stage request.

For *mt*, specify the media type; for information about valid media types, see the `mcf(4)` man page.

For *vsu*, specify the volume to mount.

The `:stidle` Command

The `stidle` command idles the stager. Use this command if you want the stager to finish its current tasks and not commence any additional staging.

The `:strun` Command

The `strun` command restarts staging activity. You can use this command to restart the stager after you have issued the `stidle` command.

File System Commands: I/O Management

The following commands enable you to manage I/O characteristics dynamically.

The `:flush_behind` *eq value* Command

The `flush_behind` command sets the maximum `flush_behind` *value*. When set to a value greater than 0, modified pages that are being written sequentially are written to disk asynchronously to help the Solaris kernel layer keep the pages clean. This option sets the maximum `flush_behind` value.

For *value*, specify an integer number of kilobytes such that $0 \leq \text{value} \leq 8192$. By default, *value*=0, which disables `flush_behind`.

For *eq*, specify the Equipment Ordinal for the file system.

The `:force_nfs_async eq` and `:noforce_nfs_async eq` Commands

These commands enable you to control whether the file system caches NFS data written to the server even if NFS has requested that the data be written synchronously through to disk. The `force_nfs_async` command caches NFS data. The `noforce_nfs_async` command, which is the default, synchronously writes data through to disk.

The `force_nfs_async` command is effective only if the file system is mounted as an NFS server and only if the clients are mounted with the `noac` NFS mount option. For more information about mounting an NFS file system, see the `mount_nfs(1M)` man page.

For *eq*, specify the Equipment Ordinal for the file system.



Caution – The `force_nfs_async` option violates NFS protocols. Use this command with caution. In the event of a server interruption, data can be lost. Data is cached on the NFS server and cannot be seen immediately by all the clients if there are multiple NFS servers.

The `:readahead eq contig` Command

The `readahead` command specifies the maximum number of bytes that can be read ahead by the file system.

For *eq*, specify the Equipment Ordinal for the file system.

For *contig*, specify units of 1-kilobyte blocks. This must be an integer such that $1 < contig < 8192$. The *contig* specified is truncated to a multiple of 8 kilobytes. The default *contig* is 8 (131072 bytes).

For example, the following command sets the maximum contiguous block size to 262,144 bytes for the file system defined as Equipment Ordinal 3:

```
:readahead 3 256
```

This value can also be configured in the `samfs.cmd` file by specifying the `readahead` directive. For more information, see the `samfs.cmd(4)` man page.

The `:sw_raid eq` and `:nosw_raid eq` Commands

These options specify whether or not the file system aligns the writebehind buffer. Specify `sw_raid` if the software RAID feature of a package such as Solstice DiskSuite is also used on this file system. The default setting is `nosw_raid`.

For *eq*, specify the Equipment Ordinal for a file system.

The `:writebehind eq contig` Command

The `writebehind` command specifies the maximum number of bytes that can be written behind by a file system.

For *eq*, specify the Equipment Ordinal for a file system.

For *contig*, specify units of 1-kilobyte blocks. This must be an integer such that $1 < contig < 8192$. The default *contig* is 8 (131072 bytes).

For example, the following command sets the maximum contiguous block size to 262,144 bytes for the file system defined as Equipment Ordinal 50:

```
:writebehind 50 256
```

This value can also be configured in the `samfs.cmd` file by specifying the `writebehind` directive. For more information, see the `samfs.cmd(4)` man page.

The `:wr_throttle eq value` Command

The `wr_throttle` command sets the number of outstanding write bytes for one file to *value* kilobytes.

For *eq*, specify the Equipment Ordinal for a file system.

For *value*, specify an integer number of kilobytes. If *value*=0, there is no limit. The default is 16384.

File System Commands: Direct I/O Management

The commands in this section control I/O on Sun StorEdge SAM-FS file systems. They enable you to change the type of I/O for an individual file based on I/O size and history. If direct I/O is specified for a file, for example, through the `setfa(1)` command, these options are ignored and all I/O to regular files is direct, if possible.

These commands refer to both well-aligned and misaligned I/O. *Well-aligned* I/O occurs when the file offset falls on a 512-byte boundary and when the length of the I/O transfer is at least 512 bytes. *Misaligned* I/O occurs when the file offset does not fall on a 512-byte boundary and the length of the transfer is less than 512 bytes.

For more information about I/O and I/O management, see “Advanced Topics” on page 69.

The `:dio_rd_form_min eq value` and `:dio_wr_form_min eq value` Commands

These commands set the lower limits for well-aligned I/O to *value* 1024-byte blocks. Use the `dio_rd_form_min` command to set the *value* for reads, and use the `dio_wr_form_min` command to set the *value* for writes.

For *eq*, specify the Equipment Ordinal for the file system.

For *value*, specify an integer number of 1024-byte blocks to use for the lower limit. By default, *value*=256. If *value*=0, automatic I/O switching is disabled.

The `:dio_rd_ill_min eq value` and `:dio_wr_ill_min eq value` Commands

These commands set the lower limit for misaligned I/O to *value* 1024-byte blocks. Use the `dio_rd_ill_min` command to set the *value* for reads, and use the `dio_wr_ill_min` command to set the *value* for writes.

For *eq*, specify the Equipment Ordinal for the file system.

For *value*, specify an integer number of 1024-byte blocks to use for the lower limit. By default, *value*=256. If *value*=0, automatic I/O switching is disabled.

The `:dio_rd_consec eq value` and `:dio_wr_consec eq value` Commands

These commands set the number of consecutive I/O transfers that can occur, with a buffer size greater than the specified lower limits, to *value* operations.

For *eq*, specify the Equipment Ordinal for the file system.

For *value*, specify the number of consecutive I/O transfers with a buffer size greater than the specified lower limit. The specified lower limit is the *value* of `dio_rd_form_min` for aligned reads or `dio_rd_ill_min` for misaligned reads. By default, *value*=0, which means that no default direct reads occur based on I/O sizes.

For more information, see one or more of the following commands or mount parameters:

- “The `:dio_rd_form_min` *eq* value and `:dio_wr_form_min` *eq* value Commands” on page 159
- “The `:dio_rd_ill_min` *eq* value and `:dio_wr_ill_min` *eq* value Commands” on page 159

The `:dio_szero` *eq* and `:nodio_szero` *eq* Commands

These commands set or clear the direct I/O sparse zeroing mount option.

The `dio_szero` option causes uninitialized areas of sparse files written with direct I/O to be zeroed when the area is accessed. This makes the sparse file behavior the same as that for paged I/O. By default, sparse files written by direct I/O do not have the uninitialized areas zeroed for performance reasons. The default is `nodio_szero`.

For *eq*, specify the Equipment Ordinal for the file system.

The `:forcedirectio` *eq* and `:noforcedirectio` *eq* Commands

These commands enable you to control whether direct I/O be used as the default I/O mode. By default, the I/O mode is buffered and uses the page cache. The `forcedirectio` command enables direct I/O for all transfers. The `noforcedirectio` command enables the default, which is buffered I/O.

For *eq*, specify the Equipment Ordinal for the file system.

When direct I/O is specified, the system transfers data directly between the user’s buffer and disk. Use direct I/O only if the file system is used for large, block-aligned, sequential I/O.

For more information about I/O, see “Advanced Topics” on page 69.

File System Commands: Miscellaneous

The following commands enable you to control leases, allocation sizes, and various other file system characteristics.

The `:abr` *eq* and `:noabr` *eq* Commands

These commands set or clear the Application Based Recovery (ABR) mount option.

For use in an Oracle RAC environment with Sun StorEdge QFS AIO only. These mount options disable or enable ABR of software mirrors. They apply only to Sun StorEdge QFS file systems built on Solaris Volume manager mirrored volumes that support ABR.

For *eq*, specify the Equipment Ordinal for the file system.

The `:dmr eq` and `:nodmr eq` Commands

These commands set or clear the Directed Mirror Reads (DMR) mount option.

For use in an Oracle RAC environment with Sun StorEdge QFS AIO only. These mount options disable or enable DMR of software mirrors. They apply only to Sun StorEdge QFS file systems built on Solaris Volume Manager mirrored volumes that support DMR.

For *eq*, specify the Equipment Ordinal for the file system.

The `:invalid eq interval` Command

The `invalid` command specifies that the file system hold cached attributes for at least *interval* seconds after a file is modified. You can specify this command only if the file system was mounted originally with the `reader` mount option. For information about mount options, see the `mount_samfs(1M)` man page.

For *eq*, specify the Equipment Ordinal for the file system.

For *interval*, specify the number of seconds to hold the attributes after file modification. For example, assume that *interval*=30. In such a file system, if you issue an `ls(1)` command, you might not see a newly created file appear in its output for 30 seconds after it has been created on its writer host.

The `:mm_stripe eq value` Command

The `mm_stripe` command sets the metadata stripe width for the file system to *value* 16-kilobyte disk allocation units (DAUs).

For *eq*, specify the Equipment Ordinal of the file system.

For *value*, specify either 0 or 1. If *value*=1, which is the default, the file system writes one DAU of metadata to one LUN before switching to another LUN. If *value*=0, the metadata is round-robin across all available metadata LUNs.

The `:refresh_at_eof eq` and `:norefresh_at_eof eq` Commands

The `refresh_at_eof` and `norefresh_at_eof` commands can be used for fast updates to a Sun StorEdge QFS multireader file system on hosts that are mounted with the `reader` mount option in a multireader file system. This option ensures that the system refreshes the current file size when the read buffer exceeds the end of file. You can use this, for example, if the writer host system is appending to a file and the reader is issuing `tail(1)` commands with the `-f` option. The default is `norefresh_at_eof`.

For *eq*, specify the Equipment Ordinal of the file system.

The `:suid eq` and `:nosuid eq` Commands

The `suid` and `nosuid` commands control whether running programs are allowed to automatically change their owner IDs. For more information about the implications of using these mount options, see the `suid` and `nosuid` mount option descriptions on the `mount_ufs(1M)` man page and see the `suid(2)` man page.

For *eq*, specify the Equipment Ordinal of the file system.

The `:stripe eq value` Command

The `stripe` command sets the stripe width for the file system to *value* disk allocation units (DAUs). The stripe width specifies that *value* multiplied by the DAU bytes are written to one LUN before switching to the next LUN. You can use the `sammkfs(1M) -a` command to set the DAU size on the file system when it is initialized.

For *eq*, specify the Equipment Ordinal of the file system.

For *value*, specify an integer such that $0 < value < 255$. If *value*=0, files are round-robin on each slice. The default *value* on file systems with an `ms` Equipment Type and on file systems with an `ma` Equipment Type with no striped group (`gXXX`) components is as follows:

- 128 kilobytes/DAU for DAUs < 128 kilobytes
- 1 for DAUs > 128 kilobytes

By default, *value*=0 on a Sun StorEdge QFS shared file system.

By default, *value*=0 on file systems with an `ma` Equipment Type with any striped group (`gXXX`) components.

The system sets *value*=0 if mismatched striped groups exist.

For more information about file system types, see “Design Basics” on page 5 and see “System Configuration Tasks” on page 13.

The `:sync_meta eq value` Command

The `sync_meta` command determines whether metadata is written to disk every time it changes. If you are using this command on Sun StorEdge QFS shared file system, also see “Specifying the Frequency With Which Metadata Is Written: the `sync_meta=n` Option” on page 221.

For *eq*, specify the Equipment Ordinal of the file system.

For *value*, specify either 0 or 1, as follows:

- If *value* is 0, metadata is held in a buffer after it changes. For an unshared Sun StorEdge QFS file system in which higher performance is desired, you can set *value* to 0. In this case, the system performs a delayed write in which metadata is held in a buffer before it is written to disk. This is the default for unshared file systems and for file systems that are not mounted as multireader file systems.
- If *value* is 1, metadata is written to disk every time it changes. This slows performance, but it increases data consistency. This is the default for Sun StorEdge QFS file systems mounted as multireader file systems or as shared file systems. For a Sun StorEdge QFS shared file system, *value* must be set to 1 if failover capability is required.

The `:trace eq` and `:notrace eq` Commands

The `trace` command enables tracing for a file system. The `notrace` command disables tracing. These are global directives that affect all operations. For more information about file system tracing, see the `defaults.conf(4)` man page.

For *eq*, specify the Equipment Ordinal of a file system.

Automated Library Commands

The following commands control media activities in an automated library.

The `:audit [-e] eq [:slot [:side]]` Commands

The `audit` command causes the specified automated library device to mount each volume, read the VSN, and rebuild the library catalog.

If `-e` is specified, and the volume is on a tape cartridge, the tape skips to the end of data (EOD) and updates the space available. Note that the skip to EOD is not interruptible. Under certain conditions, it can take hours to complete.

For *eq*, specify the Equipment Ordinal of an automated library device.

For *slot*, specify the slot number containing the volume you want to load.

For *side*, specify the side of a magneto-optical disk. Must be 1 or 2. This argument is not applicable to tape cartridges.

This command is not supported for network-attached libraries.

The `:export eq:slot` and `:export mt.vsn` Commands

The `export` command causes the specified automated library to export a volume to the mail slot. The volume is identified by its slot position within the automated library.

- If exporting by Equipment Ordinal and slot number, the specified automated library moves the volume to the mail slot. For *eq*, specify the Equipment Ordinal or device name. For *slot*, specify the slot number containing the volume you want to export.
- If exporting by logical identifier, the specified automated library moves the volume to the mail slot. For *mt*, specify the media type; for information about valid media types, see the `mcf(4)` man page. For *vs**n*, specify the volume to export.

The `:import eq` Command

The `import` command causes the specified automated library to enable you to add a cartridge. For *eq*, specify the Equipment Ordinal of the automated library.

The `:load eq:slot [:side]` and `:load mt.vsn` Commands

The `load` command enables you to load by either a physical or a logical identifier, as follows:

- If loading by Equipment Ordinal and slot number, the specified automated library loads the volume into a drive.

For *eq*, specify the Equipment Ordinal or device name.

For *slot*, specify the slot number containing the volume you want to load.

For *side*, specify the side of a magneto-optical disk. Must be 1 or 2. This argument is not applicable to tape cartridges.

- If loading by logical identifier, the specified automated library to load mounts a labeled volume into a drive.

For *mt*, specify the media type; for information about valid media types, see the `mcf(4)` man page.

For *vsn*, specify the volume to mount.

The `:priority pid newpri` Command

The `priority` command sets the load priority for a process. You can specify this command from the removable media mount requests display. For more information, see “(p) - Removable Media Load Requests Display” on page 126.

For *pid*, specify the priority shown in the `p` display.

For *newpri*, specify the priority you want to give the request. This should be an integer number.

Miscellaneous Commands

The following commands enable you to control tracing, open access to a disk device, and perform several other miscellaneous tasks.

The `:clear vsn [index]` Command

The `clear` command clears the specified VSN from the removable media mount requests display. For more information, see “(p) - Removable Media Load Requests Display” on page 126.

For *vsn*, specify the volume to mount. Any process waiting for the VSN mount is aborted.

For *index*, specify the decimal ordinal of the VSN in the removable media display.

The `:devlog eq [option]` Command

The `devlog` command sets one or more events to be logged.

For *eq*, specify the Equipment Ordinal of a device.

For *option*, specify one or more event types. Possible event types are as follows: *all*, *date*, *default*, *detail*, *err*, *event*, *label*, *mig*, *module*, *msg*, *none*, *retry*, *stage*, *syserr*, and *time*. For information about these options, see the `defaults.conf(4)` man page. If no *option* is specified, the system does not change the current events being logged for the *eq* specified.

The `:diskvols volume [+flag | -flag]` Command

The `diskvols` command sets or clears flags in the disk volume dictionary.

For *volume*, specify the volume in the disk volume dictionary.

For *flag*, specify one of the five flags in the `D samu(1M)` display. For information about the disk volume dictionary and the flags see “(D) - Disk Volume Dictionary” on page 108 or the `samu(1M)` man page.

The `:dtrace` Commands

The `dtrace` commands are as follows:

- `:dtrace daemon_name on`
- `:dtrace daemon_name off`
- `:dtrace daemon_name.variable value`

The `dtrace` commands specify various tracing options. TABLE 5-44 shows the tracing control command arguments.

TABLE 5-44 Tracing Command Arguments

Argument	Description
<i>daemon_name</i>	Specify the keyword <i>all</i> or a process name. If the keyword <i>all</i> is specified, the tracing command affects all daemons. If one of the following process names is specified, the tracing command affects that process only: <i>sam-archiverd</i> , <i>sam-catserverd</i> , <i>sam-fsd</i> , <i>sam-rftd</i> , <i>sam-recycler</i> , <i>sam-sharefsd</i> , and <i>sam-stagerd</i> . One of the keywords <i>on</i> or <i>off</i> can be specified after a process name. If <i>on</i> or <i>off</i> is specified, tracing is turned off or on for all processes specified.

TABLE 5-44 Tracing Command Arguments (*Continued*)

Argument	Description
<i>variable value</i>	Many different <i>variable</i> and <i>value</i> arguments can be specified. The <code>defaults.conf(4)</code> man page contains comprehensive information about these arguments. Specify one of the following <i>variable</i> and <i>value</i> combinations: <ul style="list-style-type: none">• <i>file value</i>. For <i>value</i>, specify the name of a file to which trace files can be written. This can be a full path name.• <i>options value</i>. For <i>value</i>, specify a space-separated list of trace options.• <i>age value</i>. For <i>age</i>, specify the trace file rotation age. Note: Do not set this value to two minutes or less. If you do, the rotation will never take place.• <i>size value</i>. For <i>value</i>, specify the size of the trace file at which rotation will begin.

The `:fs fsname` Command

The `fs` command sets the file system to be displayed through the `N` display.

For *fsname*, specify the name of the file system to be examined.

The `:mount mntpt` Command

The `mount` command selects a Sun StorEdge SAM-FS file system. For *mntpt*, specify the mount point of a file system.

The `:open eq` Command

The `open` command enables access to the specified disk device. You must issue this command before you can use the `read` command, disk sector display (`S`), or file label display (`F`).

For *eq*, specify the Equipment Ordinal of a device.

The `:read addr` Command

The `read` command reads the specified sector from the currently opened disk device. You must open the device before it can be read.

For *addr*, specify the hexadecimal sector address.

The `:refresh i` Command

The `refresh` command determines the amount of time between `samu(1M)` screen refreshes.

For *i*, specify a time in seconds.

The `:snap [filename]` Command

The `snap` command sends a snapshot of a display window to *filename*, which is the name of a file to receive the display information.

To aid in problem reporting, you can take a snapshot of all the `samu(1M)` utility's displays. Each new snapshot is appended to the snapshots file. The default file is `snapshots` in the current working directory. The file can be printed, examined using `vi(1)`, or faxed to Sun Microsystems customer support staff.

The `:! shell_command` Command

The `!` command enables you to run a shell command without leaving the `samu(1M)` operator utility.

Glossary

A

- addressable storage** The storage space encompassing online, nearline, offsite, and offline storage that is user-referenced through a Sun StorEdge QFS or Sun StorEdge SAM-FS file system.
- archive media** The media to which an archive file is written. Archive media can be removable tape or magneto-optical cartridges in a library. In addition, archive media can be a mount point on another system.
- archiver** The archive program that automatically controls the copying of files to removable cartridges.
- archive storage** Copies of file data that have been created on archive media.
- audit (full)** The process of loading cartridges to verify their VSNs. For magneto-optical cartridges, the capacity and space information is determined and entered into the automated library's catalog.
- automated library** A robotically controlled device designed to automatically load and unload removable media cartridges without operator intervention. An automated library contains one or more drives and a transport mechanism that moves cartridges to and from the storage slots and the drives.

B

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

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

block size See *DAU*.

C

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

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

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

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

D

data device In a file system, a device or group of devices upon which file data is stored.

DAU Disk allocation unit. The basic unit of online storage. Also called *block size*.

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

device scanner Software that periodically monitors the presence of all manually mounted removable devices and that detects the presence of mounted cartridges that can be requested by a user or other process.

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

direct-attached library An automated library connected directly to a server using a SCSI interface. A SCSI-attached library is controlled directly by the Sun StorEdge SAM-FS software.

direct I/O	An attribute used for large block-aligned sequential I/O. The <code>setfa(1)</code> command's <code>-D</code> option is the direct I/O option. It sets the direct I/O attribute for a file or directory. If applied to a directory, the direct I/O attribute is inherited.
directory	A file data structure that points to other files and directories within the file system.
disk allocation unit	See <i>DAU</i> .
disk buffer	In a Sun SAM-Remote configuration, the buffer on the server system that is used for archiving data from the client to the server.
disk cache	The disk-resident portion of the file system software, used to create and manage data files between online disk cache and archive media. Individual disk partitions or an entire disk can be used as disk cache.
disk space threshold	The maximum or minimum level of disk cache utilization, as defined by an administrator. The releaser controls disk cache utilization based on these predefined disk space thresholds.
disk striping	The process of recording a file across several disks, thereby improving access performance and increasing overall storage capacity. See also <i>striping</i> .
drive	A mechanism for transferring data to and from a removable media volume.

E

Ethernet	A local-area, packet-switched network technology. Originally designed for coaxial cable, it is now found running over shielded, twisted-pair cable. Ethernet is a 10- or 100-Mbytes/second LAN.
extent array	The array within a file's inode that defines the disk location of each data block assigned to the file.

F

family device set	See <i>family set</i> .
family set	A storage device that is represented by a group of independent physical devices, such as a collection of disks or the drives within an automated library. See also <i>storage family set</i> .

FDDI	Fiber-distributed data interface is a standard for data transmission in a local area network that can extend in range up to 200 km (124 miles). The FDDI protocol is based on the token ring protocol.
Fibre Channel	The ANSI standard that specifies high-speed serial communication between devices. Fibre Channel is used as one of the bus architectures in SCSI-3.
file system	A hierarchical collection of files and directories.
file-system-specific directives	Archiver and releaser directives that follow global directives in the <code>archiver.cmd</code> file, are specific to a particular file system, and begin with <code>fs =</code> . File-system-specific directives apply until the next <code>fs =</code> directive line or the end of file is encountered. If multiple directives affect a file system, the file-system-specific directives override the global directives.
FTP	File transfer protocol. An Internet protocol for transferring files between two hosts over a TCP/IP network.

G

global directives	Archiver and releaser directives that apply to all file systems and that appear before the first <code>fs =</code> line.
grace period	For disk quotas, the amount of time for which a user is allowed to create files and allocate storage after reaching the soft limit.

H

hard limit	For disk quotas, the maximum limit on file system resources, blocks, and inodes that users cannot exceed.
-------------------	---

I

indirect block	A disk block that contains a list of storage blocks. File systems have up to three levels of indirect blocks. A first-level indirect block contains a list of blocks used for data storage. A second-level indirect block contains a list of first-level indirect blocks. A third-level indirect block contains a list of second-level indirect blocks.
-----------------------	---

- inode** Index node. A data structure used by the file system to describe a file. An inode describes all the attributes associated with a file other than the name. The attributes include ownership, access, permission, size, and the file location on the disk system.
- inode file** A special file (`.inodes`) on the file system that contains the inode structures for all files resident in the file system. Inodes are 512 bytes long. The inode file is a metadata file, which is separated from file data in the file system.

K

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

L

- LAN** Local area network.
- lease** A function that grants a client host permission to perform an operation on a file for a specified period of time. The metadata server issues leases to each client host. The leases are renewed as necessary to permit continued file operations.
- library** See *automated library*.
- library catalog** See *catalog*.
- local file system** A file system that is installed on one node of a Sun Cluster system and is not made highly available to another node. Also, a file system that is installed on a standalone server.
- LUN** Logical unit number.

M

- mcf** Master configuration file. The file that is read at initialization time that defines the relationships between the devices (the topology) in a file system environment.

media	Tape or optical disk cartridges.
media recycling	The process of recycling or reusing archive media with few active files.
metadata	Data about data. Metadata is the index information used to locate the exact data position of a file on a disk. It consists of information about files, directories, access control lists, symbolic links, removable media, segmented files, and the indexes of segmented files.
metadata device	A device (for example, a solid-state disk or mirrored device) upon which file system metadata is stored. Having file data and metadata on separate devices can increase performance. In the <code>mcf(4)</code> file, a metadata device is declared as an <code>mm</code> device within an <code>ma</code> file system.
mirror writing	The process of maintaining two copies of a file on disjointed sets of disks to prevent loss from a single disk failure.
mount point	The directory on which a file system is mounted.
multireader file system	A single-writer, multireader capability that enables you to specify a file system that can be mounted on multiple hosts. Multiple hosts can read the file system, but only one host can write to the file system. Multiple readers are specified with the <code>-o reader</code> option with the <code>mount(1M)</code> command. The single-writer host is specified with the <code>-o writer</code> option with the <code>mount(1M)</code> command. For more information on the <code>mount(1M)</code> command, see the <code>mount_samfs(1M)</code> man page.

N

name space	The metadata portion of a collection of files that identifies the file, its attributes, and its storage locations.
nearline storage	Removable media storage that requires robotic mounting before it can be accessed. Nearline storage is usually less expensive than online storage, but it takes somewhat longer to access.
network-attached automated library	A library, such as those from StorageTek, ADIC/Grau, IBM, or Sony, that is controlled using a software package supplied by the vendor. The Sun StorEdge SAM-FS file system interfaces with the vendor software using a Sun StorEdge SAM-FS media changer daemon designed specifically for the automated library.
NFS	Network file system. A file system distributed by Sun that provides transparent access to remote file systems on heterogeneous networks.

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

O

offline storage Storage that requires operator intervention for loading.

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

online storage Storage that is immediately available, such as disk cache storage.

P

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

preallocation The process of reserving a contiguous amount of space on the disk cache for writing a file. Preallocation can be specified only for a file that is size zero. For more information, see the `setfa(1)` man page.

pseudo device A software subsystem or driver with no associated hardware.

Q

quota The amount of system resources that a user is allowed to consume.

R

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

recycler A Sun StorEdge SAM-FS utility that reclaims space on cartridges that is occupied by expired archive copies.

- release priority** The priority according to which a file in a file system is released after being archived. Release priority is calculated by multiplication of various weights of file properties and then summation of the results.
- releaser** A Sun StorEdge SAM-FS component that identifies archived files and releases their disk cache copies, thus making more disk cache space available. The releaser automatically regulates the amount of online disk storage according to high and low thresholds.
- remote procedure call** See *RPC*.
- removable media file** A special type of user file that can be accessed directly from where it resides on a removable media cartridge, such as magnetic tape or optical disk cartridge. Also used for writing archive and stage file data.
- robot** The portion of an automated library that moves cartridges between storage slots and drives. Also called a *transport*.
- round robin** A data access method in which entire files are written to logical disks in a sequential fashion. When a single file is written to disk, the entire file is written to the first logical disk. The second file is written to the next logical disk, and so on. The size of each file determines the size of the I/O.
- See also *disk striping* and *striping*.
- RPC** Remote procedure call. The underlying data exchange mechanism used by NFS to implement custom network data servers.

S

- samfsdump** A program that creates a control structure dump and copies all the control structure information for a given group of files. It is analogous to the UNIX *tar(1)* utility, but it does not generally copy file data. See also *samfsrestore*.
- SAM-QFS** A configuration that combines the Sun StorEdge SAM-FS software with the Sun StorEdge QFS file system. SAM-QFS offers a high-speed, standard UNIX file system interface to users and administrators in conjunction with the storage and archive management utilities. It uses many of the commands available in the Sun StorEdge SAM-FS command set as well as standard UNIX file system commands.
- samfsrestore** A program that restores inode and directory information from a control structure dump. See also *samfsdump*.
- SCSI** Small Computer System Interface. An electrical communication specification commonly used for peripheral devices such as disk and tape drives and automated libraries.

small computer system interface	See SCSI.
soft limit	For disk quotas, a threshold limit on file system resources (blocks and inodes) that you can temporarily exceed. Exceeding the soft limit starts a timer. When you exceed the soft limit for the specified time, no further system resources can be allocated until you reduce file system use to a level below the soft limit.
staging	The process of copying a nearline or offline file from archive storage back to online storage.
storage family set	A set of disks that are collectively represented by a single disk family device.
storage slots	Locations inside an automated library in which cartridges are stored when not being used in a drive. If the library is direct-attached, the contents of the storage slots are kept in the automated library's catalog.
striped group	A collection of devices within a file system that is defined in the <code>mcf(4)</code> file as one or more <code>gXXX</code> devices. Striped groups are treated as one logical device and are always striped with a size equal to the disk allocation unit (DAU).
stripe size	The number of disk allocation units (DAUs) to be allocated before writing proceeds to the next device of a stripe. If the <code>stripe=0</code> mount option is used, the file system uses round-robin access, not striped access.
striping	A data access method in which files are simultaneously written to logical disks in an interlaced fashion. SAM-QFS file systems provide two types of striping: "hard striping," using stripe groups, and "soft striping," using the <code>stripe=x</code> mount parameter. Hard striping is enabled when a file system is set up, and requires the definition of stripe groups within the <code>mcf(4)</code> file. Soft striping is enabled through the <code>stripe=x</code> mount parameter, and can be changed for the file system or for individual files. It is disabled by setting <code>stripe=0</code> . Hard and soft striping can both be used if a file system is composed of multiple stripe groups with the same number of elements. See also <i>round robin</i> .
Sun SAM-Remote client	A Sun StorEdge SAM-FS system with a client daemon that contains a number of pseudodevices, and can also have its own library devices. The client depends on a Sun SAM-Remote server for archive media for one or more archive copies.
Sun SAM-Remote server	Both a full-capacity Sun StorEdge SAM-FS storage management server and a Sun SAM-Remote server daemon that defines libraries to be shared among Sun SAM-Remote clients.
superblock	A data structure in the file system that defines the basic parameters of the file system. The superblock is written to all partitions in the storage family set and identifies the partition's membership in the set.

T

- tar** Tape archive. A standard file and data recording format used for archive images.
- TCP/IP** Transmission Control Protocol/Internet Protocol. The internet protocols responsible for host-to-host addressing and routing, packet delivery (IP), and reliable delivery of data between application points (TCP).
- timer** Quota software that keeps track of the period starting when a user reaches a soft limit and ending when the hard limit is imposed on the user.

V

- volume** A named area on a cartridge for sharing data. A cartridge has one or more volumes. Double-sided cartridges have two volumes, one on each side.
- volume overflow** A capability that enables the system to span a single file over multiple volumes. Volume overflow is useful for sites using very large files that exceed the capacity of their individual cartridges.
- VSN** Volume serial name. In the context of archiving to removable media cartridges, the VSN is a logical identifier for magnetic tape and optical disk that is written in the volume label. In the context of archiving to disk cache, this is the unique name for the disk archive set.

W

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

Index

Symbols

! samu(1M) command 168
.inodes file 3

Numerics

64-bit addressing 3

A

abr samu(1M) command 160
additional parameters field 22
admin set, quotas 45, 52
API routines 4, 26
archdone file state 27
archive
 copy 30
 file attributes 26
archive(1) command 25
archiver
 commands, using samu(1M) 151
 samu(1M) display 101
archiver.cmd file
 propagating changes 34
arrrerun samu(1M) command 151
arrestart samu(1M) command 151
armarchreq samu(1M) command 151
arrun samu(1M) command 151
arscan samu(1M) command 151
arstop samu(1M) command 151
artrace samu(1M) command 151
audit samu(1M) command 163

automated library
 commands, using samu(1M) 163
 samu(1M) display 126, 139

B

backing up files 41
Buffered I/O
 see I/O, paged

C

cached
 I/O, see I/O, paged
catalina.out log file 18
checksum 31
clear samu(1M) command 165
conf.sh file 18
configuration
 changes, propagating 31
 creating the mcf file 19
 disk use 6
 file allocation 9
 samu(1M) display 104

D

daemons
 overview 69
 sam-archiverd 70
 sam-catserverd 70
 sam-fsd 70
 sam-genericd 70
 sam-initd 70
 sam-releaser 70

- sam-rftd 70
- sam-robotsd 70
- sam-rpcd 70
- sam-scannerd 70
- sam-stagealld 70
- sam-stagerd 70
- samu(1M) display 106
- tracing 71
- tracing, samu(1M) command 166
- damaged file state 27
- data alignment 7
- DAU 4
 - dual allocation scheme 7
 - overview 6
 - quotas and DAUs 47
 - settings 7
- dd(1M) command 49, 51
- device commands in samu(1M) 150
- device state codes, viewing with samu(1M) 147
- device state field 21
- device status, samu(1M) display 131
- device table, samu(1M) display 138
- devlog samu(1M) command 165
- dio_rd_consec
 - mount option 85
 - samu(1M) command 159
- dio_rd_form_min samu(1M) command 159
- dio_rd_ill_min samu(1M) command 159
- dio_szero samu(1M) command 160
- dio_wr_consec
 - mount option 85
 - samu(1M) command 159
- dio_wr_form samu(1M) command 159
- dio_wr_ill_min samu(1M) command 159
- directio(3C) function call 3, 84
- directory attributes 73
- directory name lookup cache (DNLC) 5
- disk allocation unit, see DAU
- disk cache
 - adding 39
 - exceeding 83
- disk volume dictionary
 - samu(1M) display 108
- disks
 - adding or upgrading 41

- diskvols samu(1M) command 166
- diskvols.conf file
 - samu(1M) display 108
- dmr samu(1M) command 161
- documentation xxii
- down samu(1M) command 150
- dtrace samu(1M) command 166
- dual allocation scheme 7

E

- EDOM error 93
- EDQUOT error 47
- ENOC SI error 93
- equipment
 - identifier field 20
 - ordinal field 20
 - type field 20
- export samu(1M) command 164

F

- family set field 21
- file allocation
 - methods 75
 - preallocating space 74
 - round robin 9
 - striped 9, 10
- file attributes 5, 25, 73
- file retention 31
- file states 27
- file system
 - adding disk cache 39
 - backing up 41
 - basic operations 25
 - capacity 3
 - commands, using samu(1M) 156, 160
 - design 5
 - direct I/O commands, using samu(1M) 158
 - exceeding cache 83
 - features 2
 - file allocation 9
 - file information 28
 - mounting 35
 - ms 21
 - number of files supported 3
 - parameters, samu(1M) display 123
 - quotas, see quotas

- recovery 3
 - recreating 41
 - renaming 40
 - repair 94
 - samu(1M) display 109
 - status codes, viewing with samu(1M) 146
 - unmounting 38
 - usage, samu(1M) display 116
 - volume management 2
- File System Manager**
- adding accounts 15
 - File System Manager Portal Agent 17
 - log files 18
 - overview 13
 - using 14
- files**
- accommodating large files 83
 - large file transfer 85
- flush_behind**
- mount option 89
 - samu(1M) command 156
- force_nfs_async** samu(1M) command 157
- forcedirectio**
- mount option 84
 - samu(1M) command 160
- fs** samu(1M) command 167
- fsck(1M)** command, also see **samfsck(1M)** command 3
- fsmgr** agent command 17
- G**
- grace period for quotas 59, 61
- H**
- hard limit, quotas 47
- hash table 91
- help
- samu(1M) display 111
- hwm_archive** samu(1M) command 153
- I**
- I/O**
- direct 2, 84
 - direct I/O file attribute 26
 - direct I/O samu(1M) commands 158
 - paged 2, 86, 87
 - switching 85
 - tuning 84
- idle samu(1M) command 150
- import samu(1M) command 164
- infinite quotas 53
- inode**
- attributes 25
 - file content 5
 - samu(1M) display 113
- inodes** file 3
- inodes** hash table, tuning 90
- invalid** samu(1M) command 161
- K**
- kernel statistics, samu(1M) display 115
- L**
- library catalog, samu(1M) display 139
- licensing**
- general information xxiii
- load** samu(1M) command 164
- log files**
- File System Manager 18
 - file system troubleshooting 93
 - using the restore.sh(1M) script 43
- ls(1)** command, also see **sls(1)** command 28
- M**
- mass-storage, samu(1M) display 118
- maxpartial samu(1M) command 154
- maxphys parameter 85
- mcf** file
- /dev/dsk entry 20
 - /dev/rmt entry 20
 - /dev/samst entry 20
 - additional parameters field 22
 - configuring 19
 - device state field 21
 - equipment identifier field 20
 - equipment ordinal field 20
 - equipment type field 20
 - error checking 42
 - family set field 21
 - fields 19
 - increasing file system size 40
- md** device 21
- memory, samu(1M) display 105

- messages file 40, 94
- metadata 5
 - content 5
 - separation 5
- mm_stripe samu(1M) command 161
- mount options
 - flush_behind 89
 - for quotas 46
 - stage_flush_behind 89
 - WORM-FS 76
 - wr_throttle 88
- mount parameters 35
- mount samu(1M) command 167
- mount(1M) command 22, 35, 36
- ms file system 21

N

- nhino parameter 90, 91
- ninodes parameter 90, 91
- noabr samu(1M) command 160
- nodio_szero samu(1M) command 160
- nodmr samu(1M) command 161
- noforce_nfs_async samu(1M) command 157
- noforcedirectio samu(1M) command 160
- nohwm_archive samu(1M) command 153
- noquota mount option 65
- norefresh_at_eof samu(1M) command 162
- nosetuid samu(1M) command 162
- nosw_raid samu(1M) command 158
- notrace samu(1M) command 163

O

- off samu(1M) command 150
- offline file state 27
- on samu(1M) command 151
- open samu(1M) command 167
- operating system xix
- operator utility, see samu(1M)
- optical disk label, samu(1M) display 111
- optical disk status
 - samu(1M) display 124

P

- paged I/O, see I/O

- partial samu(1M) command 154
- partial_stage samu(1M) command 155
- partitions (adding, changing, deleting) 41
- pending stage queue, samu(1M) display 143
- portal agent 17
- preallocation of file space 4, 74
- priority samu(1M) command 165
- processes, see daemons
- propagating configuration changes 31

Q

- quota mount option 46
- quota.group file 55
- quotas
 - admin set 45, 52
 - changing 59
 - changing limits 55
 - checking 57
 - commands 48
 - configuring 48, 50
 - correcting 66
 - DAU with quotas 47
 - default 54
 - disk blocks and file limits 47
 - enabling 48
 - enabling limits 55
 - grace period 59, 61
 - hard limit 47
 - infinite 53
 - inhibiting resource allocation 63
 - overview 45
 - quota files 46
 - removing 59, 65
 - soft limit 47
 - viewing group quotas 64
 - zero 53

R

- read samu(1M) command 167
- readahead
 - mount option 87
 - samu(1M) command 157
- recovery of a file system 3
- refresh samu(1M) command 168
- refresh_at_eof samu(1M) command 162
- release file attributes 26

- release(1) command 25
- releaser commands, using samu(1M) 154
- removable media
 - samu(1M) display 126
 - status codes, samu(1M) 145
 - status, samu(1M) display 129
- removing quotas 65
- renaming a file system 40
- repairing a file system 94
- restore.sh(1M) script 43
- retention 31
 - period, WORM-FS 77
- robot, see automated library
- round-robin allocation 9
 - user specified 75

S

- sam_archive(3) API routine 26
- sam_release(3) API routine 26
- sam_segment(3) API routine 26
- sam_setfa(3) API routine 26, 84
- sam_ssum(3) API routine 26
- sam_stage(3) API routine 26
- sam-archiverd daemon 70
- sam-catserverd daemon 70
- samchaid(1M) command 53
- samcmd(1M) aridle command 34, 39
- samcmd(1M) command 35
- samd(1M) command 34, 39, 40, 42, 70
- samfs.cmd file 22, 35, 85
 - example 37
- samfsck(1M) command 21, 93, 94
- sam-fsd daemon 70
- sam-fsd(1M) command 40, 42, 72
- samfsinfo(1M) command 22
- samfsrestore(1M) command 43
- sam-genericd daemon 70
- samgrowfs(1M) command 39, 41
- sam-initd daemon 70
- sammkfs(1M) command 21, 22, 42
- SAM-QFS xix
- samquota(1M) command 48
 - arguments 60
- sam-releaser process 70

- SAM-Remote, samu(1M) display 131
- sam-rftd daemon 70
- sam-robotsd daemon 70
- sam-rpcd daemon 70
- sam-scannerd daemon 70
- sam-stagealld daemon 70
- sam-stagerd daemon 70
- samu(1M) utility
 - archiver commands 151
 - automated library commands 163
 - device commands 150
 - direct I/O commands 158
 - display keys 99
 - file system commands 156, 160
 - help display 111
 - interface 99
 - invocation 97
 - mount command 35
 - operator commands 150
 - operator displays 100
 - overview 97
 - releaser commands 154
 - stager commands 155
 - status codes 145
- SCSI disk, transfers 86
- SCSI status, samu(1M) display 135
- sd.conf file 86
- sd_max_xfer_size definition 86
- sector data, samu(1M) display 133
- segment file attribute 27
- segment(1) command 26, 84
- services, samu(1M) display 128
- setfa file attributes 26
- setfa(1) command 3, 26, 73, 84
- setuid samu(1M) command 162
- sfind(1M) command
 - WORM options 82
- shared memory, samu(1M) display 114, 117, 120
- sls(1) checksum 31
- sls(1) command 25, 28
 - output explanation 28
 - WORM options 81
- snap samu(1M) command 168
- soft limit, quotas 47
- software

- documentation xxii
- licensing xxiii
- operating system requirements xix
- squota(1) command 48
- ssd.conf file 86
- ssd_max_xfer_size definition 86
- ssum(1) command 25
- stage file attributes 26
- stage(1) command 26
- stage_flush_behind
 - mount option 89
 - samu(1M) command 155
- stage_n_window samu(1M) command 155
- stage_retries samu(1M) command 156
- stager
 - commands, using samu(1M) 155
 - pending stage queue samu(1M) display 143
 - status, samu(1M) display 122, 136
- stager.cmd file
 - propagating changes 34
- status codes
 - viewing with samu(1M) 145, 146
- stclear samu(1M) command 156
- stidle samu(1M) command 156
- stripe
 - mount option 88
 - samu(1M) command 162
- stripe widths
 - data disks 8
- striped allocation 10
 - stripe width 75, 88
 - user specified 75
- striping
 - introduction 10
- strun samu(1M) command 156
- superblocks, version 1 and 2 23
- sw_raid samu(1M) command 158
- sync_meta
 - samu(1M) command 163
- syslog.conf file 93
- system file 85

T

- tape drive
 - commands, using samu(1M) 163

- samu(1M) display 134
- technical support xxiii
- thresh samu(1M) command 153
- Tomcat log files 18
- trace file 71
 - directory 71
 - events 71
 - rotation 72
- trace samu(1M) command 163
- trace_rotate(1M) command 72

U

- unavail samu(1M) command 151
- unload samu(1M) command 151
- unloading media, with samu(1M) 151
- unmounting
 - file system 38
- upgrading
 - disks 41

V

- VFS, vnode interface
- vfstab file 22, 35
 - fields 36
- Vnode interface
 - see VFS
- volume management 2

W

- WORM retention 31
- worm_capable mount option 76
- WORM-FS
 - creating a WORM file 78
 - mount options 76
 - options 77
 - overview 75
- wr_throttle
 - mount option 88
 - samu(1M) command 158
- writebehind
 - mount option 86
 - samu(1M) command 158

Z

- zero quotas 53