

IBM TotalStorage™ SAN File System
(based on IBM Storage Tank™ technology)



Administrator's Guide and Reference

Version 1 Release 1

IBM TotalStorage™ SAN File System
(based on IBM Storage Tank™ technology)



Administrator's Guide and Reference

Version 1 Release 1

NOTE

Before using this information and the product it supports, read the general information in Appendix I, "Notices", on page 439.

First Edition (November 2003)

This edition applies to the IBM TotalStorage SAN File System and to all subsequent releases and modifications until otherwise indicated in new editions.

Order publications through your IBM representative or the IBM branch office servicing your locality. Publications are not stocked at the address below.

IBM welcomes your comments. A form for reader's comments is provided at the back of this publication. If the form has been removed, you may address your comments to:

International Business Machines Corporation
Design & Information Development
Department CGFA
PO Box 12195
Research Triangle Park, NC 27709-9990
U.S.A.

You can also submit comments by selecting **Feedback** at www.ibm.com/storage/support.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2003. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this guide vii

Who should use this guide	vii
Notices in this guide	viii
Publications	viii
Web sites	ix

Chapter 1. Overview 1

Administrative server	3
Alerts.	4
Backup and restore	5
Service Alert	9
Common Information Model	10
Clients	11
UNIX-based clients	13
Windows-based clients	14
Cluster	16
Cluster states	19
Components	21
Container	22
Data migration	22
Engines.	23
Filesets	24
FlashCopy images	26
Global namespace	29
Locks and leases.	30
Logs.	30
Master console	32
Metadata server	33
Metadata server states.	35
Storage pools.	37
Remote access	39
Security	39
Administrative security	39
Client security	41
SNMP	42
Storage management	43
File placement	43
Policies and rules	44
Storage devices	46
User roles	46
User interfaces	47
Volumes	49

Chapter 2. Maintaining 51

Replacing master console components	51
Replacing storage engine components	51
Replacing a hot-swap fan.	51
Replacing a hot-swap power supply	52

Chapter 3. Monitoring 57

Creating a report	57
One-button data collection utility	57
Hardware Vital Product Data	58
Software Vital Product Data	59
Viewing statistics	61

Viewing system status.	62
--------------------------------	----

Chapter 4. Migrating data. 63

Estimating the time to migrate	63
Importing data into the SAN File System	64
Stopping a data migration	65
Resuming a data migration	65
Backing out migrated data	66

Chapter 5. Managing 67

Managing alerts and logs.	67
Adding SNMP managers	67
Clearing logs	68
Deleting SNMP managers	69
Modifying SNMP managers	69
Modifying SNMP traps	70
Setting up SNMP traps	70
Viewing logs	71
Viewing SNMP managers.	71
Managing backups	72
Backing up using the LUN method	72
Backing up using the API method	73
Managing clients	75
Displaying the client-driver version	75
Granting root privileges to a client	76
Listing client sessions	76
Listing clients with root privileges.	77
Rediscovering volumes accessible to a client	77
Revoking root privileges from a client	77
Starting a client	78
Stopping a client	80
Viewing client-session details	81
Viewing client-session statistics.	82
Managing the cluster	82
Changing active cluster states	82
Configuring cluster tuning	83
Listing cluster configuration.	83
Starting the cluster	84
Stopping the cluster	84
Viewing cluster details.	85
Viewing cluster statistics	85
Viewing cluster tuning details	85
Viewing the cluster software version	86
Managing disaster recovery	86
Creating a recovery file	87
Deleting a recovery file	87
Listing recovery files	88
Restoring the master console	88
Restoring the engine hardware and operating system	88
Restoring SAN connectivity	89
Restoring SAN File System software	90
Restoring SAN File System cluster configuration	90
Restoring SAN File System metadata	91
Restoring SAN File System clients.	92

Restoring SAN File System user data	93	Changing storage pool settings	123
Managing engines	93	Creating a storage pool	123
Accessing an engine through SSH	94	Deleting a storage pool	124
Collecting diagnostic data on engines.	94	Listing storage pools	124
Listing engines	95	Listing volumes in a storage pool	125
Powering off the engine	95	Setting the default storage pool	125
Powering on the engine	96	Viewing storage pool details	125
Restarting the engine	97	Viewing storage pool statistics.	126
Starting the Configuration/Setup Utility.	98	Managing users	126
Viewing engine fan status	98	Granting SAN File System user access and roles	126
Viewing engine power status	99	Listing user roles	127
Viewing engine statistics	99	Listing users.	127
Viewing the engine status summary.	100	Timing out all user authorizations	128
Viewing engine temperatures	100	Managing volumes and LUNs.	128
Viewing engine timeouts	101	Activating a volume	129
Viewing engine voltage status.	101	Adding LUNs to SAN File System	129
Viewing vital engine data	102	Adding volumes to a storage pool	130
Managing filesets (containers)	102	Changing volume settings	130
Attaching a fileset	103	Listing available LUNs	131
Changing fileset settings.	103	Listing files on a volume	131
Creating a fileset	104	Listing LUNs	131
Deleting a fileset	105	Listing volumes	131
Detaching a fileset.	105	Managing free and allocated storage space	132
Listing filesets	106	Removing volumes from a storage pool	133
Viewing fileset details	106	Suspending a volume	133
Viewing fileset settings	106	Viewing available LUN details	134
Viewing fileset statistics	106	Viewing LUN details	134
Managing FlashCopy images	107	Viewing LUN statistics	134
Creating a FlashCopy image	107	Viewing volume details	135
Deleting a FlashCopy image	108	Viewing volume settings	135
Listing FlashCopy images	108	Viewing volume statistics	135
Reverting to a previous FlashCopy image	108		
Viewing FlashCopy image details	109	Appendix A. Accessibility 137	
Managing Metadata servers	109		
Changing the master Metadata server	110	Appendix B. Commands. 139	
Checking metadata	112	tanktool	140
Listing Metadata servers.	112	tankpasswd	141
Starting a Metadata server	112	User assistance for commands.	142
Starting the Metadata server restart service	113	Administrative commands	142
Stopping a Metadata server.	114	activatevol	148
Stopping the Metadata server restart service	115	addsnmpmgr	149
Viewing Metadata server details	115	attachcontainer	151
Viewing Metadata server networking details	116	buildrscript	153
Viewing Metadata server statistics	116	catlog	154
Viewing Metadata server restart service statistics	116	catpolicy	156
Removing down-level Metadata server software	117	chclusterconfig	157
Managing policies	117	chcontainer	159
Activating a policy	118	chpool.	161
Changing the rules in a policy.	118	chvol	163
Copying a policy	119	clearlog	165
Creating a policy	119	collectdiag	166
Deleting a policy	120	detachcontainer	168
Listing policies	120	exit.	170
Changing the name of a policy	120	help	171
Viewing policy details	121	lsadmuser	172
Viewing policy rules	121	lsautorestart	175
Managing processes	121	lsclient	179
Listing processes	121	lscontainer	183
Viewing process details	122	lsdrfile	187
Viewing process limits	122	lsengine	189
Managing storage pools	122		

lsmage	192
lslun	195
lspolicy	199
lspool	202
lsproc	205
lsserver	207
lssnmpmgr	211
lstrapsetting	212
lsvol	213
mkcontainer	217
mkdrfile	220
mkimage	221
mkpolicy	223
mkpool	225
mkvol	227
quiescecluster	229
quit	231
reportvolfiles	232
resetadmuser	234
restartengine	235
resumecluster	236
reverttoimage	237
rmcontainer	239
rmdrfile	241
rmimage	242
rmpolicy	244
rmpool	245
rmsnmpmgr	246
rmvol	247
setcontainerserver	249
setdefaultpool	251
setmaster	252
setoutput	255
settrap	257
startautorestart	259
startcluster	261
startengine	262
startmetadatacheck	263
startserver	265
statcluster	267
statcontainer	271
statengine	272
statserver	276
stopautorestart	279
stopcluster	280
stopengine	281
stopmetadatacheck	282
stopserver	283
suspendvol	285
upgradecluster	286
usepolicy	287
Client commands	288
AIX-client commands	288
Windows-client command	307
Command modes	311
Naming guidelines	311
Standard format parameters	312
Standard listing parameters	314
Syntax diagram conventions	315

Appendix C. Environment variables 319

Appendix D. Getting help, service, and information 321

Service support	322
Before you call for service	323
Getting help online	323
Getting help by telephone	323

Appendix E. Purchasing additional services. 325

Appendix F. Logs. 327

Administrative log	327
Audit log	328
Event log	329
Security log	330
Server log	330

Appendix G. SAN File System console help panels 333

Using the SAN File System console	333
Overview	333
Tasks and panels	336
Status, help, and tables	340
Add Volumes to a Storage Pool- Add Volumes	344
Add Volumes to a Storage Pool- Introduction	345
Add Volumes to a Storage Pool- Select Storage pool	345
Add Volumes to a Storage Pool- Verify Settings	346
Administrative Log	347
Attach Fileset	348
Audit Log	349
Available LUNs	350
Change Power State of Engine	351
Change State of Cluster	352
Check Metadata	352
Check Metadata Progress	353
Client Privileges	354
Client Sessions	355
Cluster	356
Cluster Log	358
Cluster Properties - Details	360
Cluster Properties - Tuning	361
Cluster Properties - Tuning Details	361
Collect Diagnostic Data	363
Collect Diagnostic Data Progress	364
Create a Fileset	365
Create FlashCopy Images of Filesets- Introduction	366
Create FlashCopy Images of Filesets- Select Filesets	367
Create FlashCopy Images of Filesets- Set Properties	367
Create FlashCopy Images of Filesets- Verify Settings	368
Create a Policy - Add Rules	369
Create a Policy - Edit Rules	371
Create a Policy - High-Level Settings	371
Create a Policy - Introduction	372
Create Recovery File	372
Create a Storage Pool - Add Volumes	373
Create a Storage Pool - Introduction	374
Create a Storage Pool - Set Properties	374

Create a Storage Pool - Verify Settings	374	Server Restart Service	408
Delete Filesets	375	Servers	410
Delete FlashCopy Images	376	SNMP Properties - SNMP Events	412
Delete Storage Pools	376	SNMP Properties - SNMP Managers	412
Detach Filesets	377	Statistics - Client Sessions	413
Details of Client Session	377	Statistics - Cluster	414
Details of Image	379	Statistics - Create Report	415
Details of LUN	379	Statistics - Engines	416
Disaster Recovery	381	Statistics - Filesets	417
Download Client Software	382	Statistics - LUNs	417
Engines	382	Statistics - Report	418
Engines Properties - Fans	384	Statistics - Storage Pools	419
Engines Properties - Power	385	Statistics - Servers	419
Engines Properties - Summary	385	Statistics - Volumes	421
Engines Properties - Temperatures	387	Stop Cluster	422
Engines Properties - Timeouts	389	Stop Servers	423
Engines Properties - Vital Engine Data	390	Storage Pools	423
Engines Properties - Voltages	390	Storage Pool Properties - Available LUNs	425
Fileset Properties - Details	391	Storage Pool Properties - Current Volumes	426
Fileset Properties - General Settings	392	Storage Pool Properties - Details	426
Filesets	394	Storage Pool Properties - General Settings	427
FlashCopy Images	395	System Overview	428
LUNs	396	Users	433
Policies	397	Using the Help Assistant	433
Policy Properties - Details	398	Volume Properties - Details	435
Policy Properties - Rules	399	Volume Properties - General Settings	435
Processes	399	Volumes	435
Processes Properties - Cluster-Level Details	400		
Processes Properties - Server-Level Details	400	Appendix H. Prerequisites	437
Remove Volumes	401	Supported browsers	437
Restart Service Statistics- Probe Overview	401	Data-migration prerequisites	437
Restart Service Statistics- Test Details	402		
Restart Service Statistics- Tuning Details	403	Appendix I. Notices	439
Revert Fileset to Image	404	Trademarks	440
Roles	405		
Sign on	405	Glossary	443
Security Log	406		
Server Properties - Details	407	Index	447
Server Properties - Networking Details	408		

About this guide

This guide provides an overview of the IBM TotalStorage SAN File System and detailed administration instructions. The information is organized as follows:

- Chapter 1, “Overview”, on page 1 explains how to use the SAN File System console and provides an overview of SAN File System and related concepts.
- Chapter 2, “Maintaining”, on page 51 describes the steps that you need to perform to maintain SAN File System on a regular basis, including replacing components for the master console and engines.
- Chapter 3, “Monitoring”, on page 57 provides information about how to keep a check on system performance through tasks such as creating a report, or viewing statistics.
- Chapter 4, “Migrating data”, on page 63 describes the required steps to perform a migration such as estimating the time to migrate, as well as backing out migrated data.
- Chapter 5, “Managing”, on page 67 gives step-by-step instructions for performing ongoing administrative tasks for SAN File System through the management of alerts, clients, engines, users, volumes, and so forth.
- The appendices provide the following additional information:
 - Accessibility features of the SAN File System console and help system
 - Commands for the Administrative command-line interface
 - Administrative command-line interface environment variables
 - Sources for getting additional help, service and information
 - Logs to assist in monitoring and troubleshooting
 - Information on using the SAN File System console and field descriptions for SAN File System console panels
 - Task prerequisites
 - Information about purchasing additional services
 - Notices

Who should use this guide

This guide should be used by the person or persons assigned to manage SAN File System on a regular basis.

Depending on the specific role assigned within the system (that is, Monitor, Operator, Backup or Administrator), SAN File System managers should have experience in at least the following skills, or have access to personnel with experience in these skills:

- Microsoft® Windows® and Windows Advanced Server (depending on your client environment)
- AIX® (depending on your client environment)
- Linux system administration
- Lightweight Directory Access Protocol (LDAP) management
- Networking and network management
- SAN management
- Command-line interface scripting
- Critical business issues (such as backup, disaster recovery, and security)

Related topics:

- “User roles” on page 46

Notices in this guide

The following notices are contained with the this guide and convey these specific meanings:

Note: These notices provide important tips, guidance, or advice.

Attention: These notices indicate possible damage to programs, devices, or data. An attention notice appears before the instruction or situation in which damage could occur.

CAUTION:

These notices indicate situations that can be potentially hazardous to you. A caution notice appears before the description of a potentially hazardous procedure step or situation.

DANGER

<p>These notices indicate situations that can be potentially lethal or extremely hazardous to you. A danger notice appears before a description of a potentially lethal or extremely hazardous procedure step or situation.</p>
--

Publications

The following publications are available in the SAN File System library. They are provided in softcopy on the *IBM TotalStorage SAN File System Publications CD* that came with your storage engine and at www.ibm.com/storage/support. To use the CD, insert it in the CD-ROM drive. If the CD does not launch automatically, follow the instructions on the CD label.

Note: The softcopy version of these publications are accessibility-enabled for the IBM Home Page Reader.

- *IBM TotalStorage SAN File System Release Notes*
This document provides any changes that were not available at the time the publications were produced. This document is available only from the technical support Web site: www.ibm.com/storage/support
- *IBM Safety Information — Read This First, SD21-0030*
This document provides translated versions of general safety notices and should be read before using this product. This document is provided only in hardcopy.
- *IBM Statement of Limited Warranty*
This publication describes the IBM statement of limited warranty as it applies to the SAN File System Model 1RX storage engine.
- *IBM TotalStorage SAN File System Software License Information*
This publication provides multilingual information regarding the software license for IBM TotalStorage SAN File System Software.
- *IBM TotalStorage SAN File System Administrator's Guide and Reference, GA27-4317*
This publication introduces the concept of SAN File System, and provides instructions for configuring, managing, and monitoring the system using the SAN File System console and Administrative command-line interfaces. This book

also contains a commands reference for tasks that can be performed at the administrative and client command-line interfaces.

- *IBM TotalStorage SAN File System Maintenance and Problem Determination Guide, GA27-4318*

This publication provides instructions for adding and replacing hardware components, monitoring and troubleshooting the system, and resolving hardware and software problems.

Note: This document is intended only for trained personnel.

- *IBM TotalStorage SAN File System Messages Reference, GC30-4076*

This publication contains message description and resolution information for errors that can occur in SAN File System software.

- *IBM TotalStorage SAN File System Planning, Installation and Configuration Guide, GA27-4316*

This publication provides detailed procedures to plan the installation and configuration of SAN File System, set up and cable the hardware, perform the minimum required configuration, migrate existing data, and upgrading software.

- *Rack Installation Instructions*

This publication provides instructions for installing the Model 1RX in a rack.

- *IBM TotalStorage SAN File System System Management API Guide and Reference, GA27-4315*

This publication contains guide and reference information for using the CIM Proxy API, including common and SAN File System-specific information.

Note: This document contains information and procedures intended for only selected IBM Business Partners. Contact your IBM representative before using this publication.

- *Subsystem Device Driver User's Guide for the IBM TotalStorage Enterprise Storage Server and the IBM TotalStorage SAN Volume Controller, SC26-7540*

The Subsystem Device Driver (SDD) provides the multipath configuration environment support for a host system that is attached to an IBM TotalStorage Enterprise Storage Server[®] (ESS), IBM TotalStorage SAN Volume Controller or IBM TotalStorage SAN File System. This book provides step-by-step procedures on how to install, configure, and use SDD for the host systems.

Note: SAN File System supports the version of the Subsystem Device Driver that is shipped with the program product.

- *IBM TotalStorage Translated Safety Notices, GA67-0043*

This publication contains translated versions of hardware caution and danger statements that appear in the publications in this library. Each caution and danger statement has an assigned number that you can use to locate the corresponding statement in your native language.

Web sites

The following Web sites have additional and up-to-date information about SAN File System:

- www.ibm.com/storage/support
- www.ibm.com/storage/software/virtualization/sfs

Related topics:

- “Using the SAN File System console” on page 333

Chapter 1. Overview

This section provides a brief overview of SAN File System. Note that when you are using the SAN File System InfoCenter, for any italicized term in this section, you'll find a corresponding topic under Concepts in the table of contents. If you are using a printed version of this information, you'll also find a corresponding topic in the Concepts section.

What is SAN File System?:

IBM TotalStorage SAN File System is a multiplatform, scalable file system and storage management solution that works with a storage area network (SAN). It uses SAN technology, which allows an enterprise to connect large numbers of devices, such as client and server machines and mass storage subsystems, to a high-performance network.

On a SAN, storage is separated from the computers that use it. With SAN File System, heterogeneous clients can access shared data directly from large, high-performance, high-function storage systems, such as IBM TotalStorage Enterprise Storage Server[®] (ESS). The SAN File System is currently built on a Fibre Channel network, and is designed to provide superior I/O performance for data sharing among heterogeneous computers. It also provides growth capability and simplified storage management.

SAN File System differs from conventional distributed file systems in that it uses a data-access model that requires *clients* to contact servers to obtain only the information they need to locate and access data on *storage devices*, not the data itself. SAN File System clients access data directly from storage devices using the high-bandwidth provided by a Fibre Channel network. Direct data access eliminates server bottlenecks and provides the performance necessary for data-intensive applications.

SAN File System presents a single, *global namespace* to clients where they can create and share data. Data consistency and integrity is maintained through SAN File System's management of distributed *locks* and the use of *leases*. SAN File System provides locks that enable file sharing among SAN File System clients, and when necessary, provides locks that allow clients to have exclusive access to files. A lease determines the maximum period of time that a server guarantees the locks it grants to clients. A client must contact the server before the lease period ends in order to retain its locks.

SAN File System also provides the benefits of automatic *file placement* through the use of *policies and rules*. Based on rules specified in a policy, SAN File System automatically stores data on devices in *storage pools* that are specifically created to provide the capabilities and performance appropriate for how the data is accessed and used.

What's in the box?:

SAN File System is shipped with hardware that is based on the IBM xSeries platform and runs the Linux operating system. The hardware is referred to as the SAN File System *engines*.

An engine, which is one of the *components* of the SAN File System architecture, comes with SAN File System server software preloaded, and is ready for configuration by onsite IBM personnel. Each engine has a *Metadata server*, which serves metadata and locks to clients and performs all metadata, administrative, and *storage management* tasks; an administrative command line interface; and an administrative infrastructure that includes an *Administrative server* and a Web-based user interface from which an administrator can monitor and control the system. The administrative *user interfaces* are built around the Common Information Model (CIM), which also allows third party CIM agents to manage SAN File System.

SAN File System engines and servers run in *clusters*. You can have from two to eight engines in a cluster. IBM onsite personnel install and configure SAN File System up to and including commissioning the Metadata servers to run in the cluster. IBM personnel can also perform *data migration* to migrate your existing data to the SAN File System global namespace.

SAN File System also includes client software that provides clients with local access to the global namespace on your SAN. The client software must be installed on client machines running AIX 5.1 or Windows 2000 Advanced Server. Note that SAN File System clients can also act as servers to a broader clientele. They provide NFS or CIFS access to the global namespace and can host applications such as database servers.

Where do I start?:

After SAN File System is installed, configured, and the Metadata servers are up and running, here are some of the tasks an administrator can perform:

- Set up *administrative security*, including assigning *user roles* to users who will administer SAN File System, to prevent unauthorized access to administrative operations, and set up *client security* to prevent unauthorized access to files in the global namespace.
- Create *filesets*, which are subsets of the entire global namespace. A fileset serves as the unit of workload that can be assigned to a Metadata server in your server cluster. After filesets are created, clients can create regular directories and files within those filesets.
- Assign filesets to Metadata servers to distribute and balance the global namespace workload across all of the servers in your cluster.
- Create storage pools that contain *volumes* that provide specific quality-of-service levels and performance for specific needs, such as storing data for specific applications or business divisions.
- Add volumes, which are LUNs labeled for use by SAN File System, to storage pools so that SAN File System can begin using them to store data.
- Define *policies and rules* that determine where newly-created user files are stored.
- Set up *alerts* and *SNMP* traps to ensure timely notification of significant events, such as those that occur in the server cluster, or conditions, such as a fileset or storage pool running out of space.
- Plan and set up a strategy for *backup and recovery* that includes using standard backup tools already available in your environment and SAN File System features, such as *FlashCopy images* (which are point-in-time copies of filesets that can be used for file-based backups) and disaster recovery files (which are built from scripts and can be used to recreate system metadata if necessary).
- View *logs* to obtain additional information about events and conditions.

Note that when SAN File System is installed, the following items exist:

- The global fileset, which is the root of the global namespace. All filesets, except nested filesets that are attached to other filesets, are attached to the global fileset using a new directory name.
- A system storage pool that is used to store system metadata, which includes information about filesets, storage pools, volumes, and file placement policies. An administrator must add one or more volumes to this storage pool before it can be used.
- A default storage pool that can be used to store all files that are not assigned to a specific user storage pool by a rule in the active policy set. An administrator must also add one or more volumes to this storage pool before it can be used.
- A null policy set that remains active until an administrator creates and activates a new one. The null policy set assigns all files to the default storage pool.

How can I get quick help with problems?:

SAN File System provides integration with the *master console*, which is a serviceability node for SAN File System and other IBM TotalStorage products. The master console enables an optional *call home* feature (which notifies IBM service personnel about any problems that occur in your SAN File System server cluster so that they can respond quickly) and the *remote access* feature (which provides the ability for IBM support personnel who are not located on your premises to assist in diagnosing and repairing failures on a SAN File System engine).

Administrative server

The SAN File System *Administrative server*, which is based on a Web server software platform, is made up of the following parts:

- The GUI Web server, which renders the Web pages that make up the SAN File System console. The console is a Web-based user interface. It can be accessed using any standard Web browser, such as Mozilla or Microsoft® Internet Explorer, that has network access to the engines that host the Metadata servers in a server cluster.
- The Administrative agent, which implements all of the management logic for the GUI, CLI, and CIM interfaces, as well as performing administrative authorization and authentication against LDAP. The Administrative agent processes all management requests initiated by an administrator from the SAN File System console, as well as requests initiated from the administrative command line interface, which is called *tanktool*. The Agent communicates with the SAN File System Metadata servers, the OS, the RSA II card, the LDAP, and Administrative agents on other nodes in the cluster when processing requests.

An Administrative server interacts with a Metadata server through the Administrative agent. When you issue a request, the Administrative agent checks with an LDAP server, which must be installed in your environment, to authenticate the user ID and password and to verify whether the user has the authority (is assigned the appropriate role) to issue a particular request. After authenticating a user, the Administrative agent interacts with the Metadata server on behalf of that user to process the request. This same system of authentication and interaction is also available to third-party CIM clients to manage SAN File System.

To ensure high availability, an Administrative server resides on each engine of a cluster. All requests are received by the Administrative server that runs on the same engine as the master Metadata server. This is the primary Administrative server. However, requests can also be processed by Administrative servers running

on other engines, and all requests are redirected to an Administrative server running on another engine if the primary Administrative server is not available.

Related topics:

- “Security” on page 39
- “Engines” on page 23
- “Metadata server” on page 33
- “Cluster” on page 16
- “User interfaces” on page 47
- “User roles” on page 46

Alerts

An event is a happening in the Metadata server or cluster, such as a change in state from online to offline. An *alert* is a message that can be generated for an event. It informs an administrator about certain conditions, such as a fileset or a storage pool reaching or exceeding its threshold.

Events are recorded as messages in the cluster log. These messages can be viewed by an administrator using the SAN File System console or by using administrative commands.

A trap is a notification mechanism to convey the occurrence of an event. An administrator can choose to set configuration parameters that determine whether Simple Network Management Protocol (SNMP) trap messages are generated for events. SNMP trap messages notify administrators of events asynchronously, and eliminate the need for an administrator to frequently view messages in the cluster log to determine the state of the SAN File System cluster.

Table 1. Alerts, events, and traps

	Alerts	Traps
Description	Warns of a significant event on a Metadata server or cluster. Also informs about condition changes such as a change in state to offline, or approaching storage capacity	Optional notification method that notifies the administrator of events asynchronously
Delivery	Sends a generated message to the terminal	Sends the administrator a generated message directly, either locally or remotely

The first configuration parameter determines where SNMP trap messages are sent. An administrator specifies a list of SNMP Managers that are the recipients of any SNMP trap messages. The list includes the IP address, port number, version of SNMP, and community string for one or two managers. If no SNMP Managers are specified, no SNMP trap messages are sent.

The second parameter specifies which types of event messages also generate SNMP trap messages. An administrator can specify any combination of messages classified as informational, warning, error, or severe. If no severity types are specified, no SNMP trap messages are sent.

In addition, if you have the call home feature activated, that feature generates a specific type of SNMP trap whenever a Metadata server encounters an event that requires notification to IBM support personnel. When a Metadata server encounters such an event, it sends an SNMP trap Protocol Data Unit (PDU) to the Master console, which then parses and converts the trap into a Simple Mail Transfer Protocol (SMTP) e-mail message. The e-mail message is then sent to a known SMTP mail server. Finally, the e-mail message is forwarded to the IBM RETAIN system for processing by support personnel. Note that all of the SNMP-related configuration parameters must be set properly for the call-home feature to work.

Related topics:

- “Adding SNMP managers” on page 67
- “Service Alert” on page 9
- “Master console” on page 32
- “SNMP” on page 42
- “Setting up SNMP traps” on page 70

Backup and restore

Backup is the process of saving copies of your files, and *recovery* is the process of restoring those copies if your original files are damaged or lost. For SAN File System, an administrator must also back up the system metadata, which includes information about fileset attachment points, storage pools, volumes, and file placement policies. This backup data is used to re-create cluster configuration if necessary.

Although SAN File System does not provide specific backup-and-restore operations for files, it supports the use of backup tools that are already present in your environment. For example, if your enterprise currently uses a storage management product such as Tivoli® Storage Manager (TSM), SAN File System clients can use the functions and features of that product to back up and restore files that reside in the SAN File System global namespace.

To perform LUN-based backups, an administrator can use the copy services features that exist in the storage subsystems that SAN File System supports.

To create a backup copy of the system metadata configuration, an administrator can use the Disaster Recovery task on the SAN File System console or an administrative command.

Using a file-based approach:

To back up and restore files in the global namespace using a file-based approach, users can run standard tools and utilities on SAN File System clients.

The first line of defense for scenarios where files have been lost but the overall system remains healthy is the use of the FlashCopy image function in SAN File System. To assist with the backup process, an administrator can choose to create FlashCopy images of filesets that can be backed up at a later time. A FlashCopy image contains read-only copies of the files in a fileset as they exist at a specific point in time.

The FlashCopy image is stored in a special subdirectory named `.flashcopy` under the fileset's root attachment point. After an administrator creates a FlashCopy

image of a fileset, a user can use standard backup tools to back up the files from a SAN File System client by specifying the path to the FlashCopy image instead of the path to the actual files.

Note: Individual files could be copied out of the .flashcopy directory if less than the entire fileset needs restoration.

Users and applications can continue working with the actual files while the backup occurs.

Understanding restrictions for a file-based approach:

Both backup administrators and users on client machines must be aware of restrictions that apply if files are backed up for use on both AIX® and Windows® clients.

First, an AIX user who requests the backups must have read permissions on all files and search permissions on all directories. This is typically a root user. A Windows user must have read permissions on all files and “list folder contents” permissions on all folders.

Second, to avoid losing security attributes, users must organize the file system in a special way. There are two options:

- The first is to ensure that the root of each fileset contains only directories and folders, and use a naming convention for each that makes it clear to users whether a directory contains files created by an AIX client or by a Windows client. The top-level directories can contain any combination of files, links, and directories, just as they can within NFS and CIFS.
- The second is to choose either AIX or Windows as the global namespace default, and to require (by convention) that files, directories, and links created by non-default clients be created in specially named directories. Users can place these specially named directories at any level in the file system; however, any file, link, or directory beneath them must also be created by the same type of client.

In either case, the special naming conventions are for user and administrator benefit only. SAN File System does not interpret them in any way. In addition, if a directory or folder grants search or “list folder contents” permissions to “Other” or “Everyone,” SAN File System does not prevent an administrator or any client user from creating a file that violates the convention.

Therefore, administrators and users must be aware of the naming conventions when performing backups, and ensure that backups of files in specially named directories are performed only from the same type of client that created the files. This means that, in an environment that has both AIX and Windows clients, the backup process must be divided into multiple parts to prevent the loss of security attributes for files. The number of parts could be as few as two, or as many as the number of directories in the global namespace, depending on the capabilities of the native backup utility a client uses.

Sometimes disaster strikes and the system — as well as the FlashCopy images— are unusable. For these disaster recovery scenarios, there are two basic approaches for backup and restoration:

1. The LUN method
2. The API method

While these methods are not mutually exclusive, your disaster recovery plans will primarily involve one or the other.

The LUN method saves and restores data at the device level (that is, a JABOB – Just a Bunch of Bytes – approach); the API method saves and restores data at the file level. For a variety of reasons, the LUN method is simpler to manage, and provides more coverage than the API method. To adopt the LUN method, however, the actual copying and restoring of data must be provided as a service by the underlying storage subsystem. If your storage subsystem meets this requirement, the LUN method is the recommended backup and restore approach. If your storage subsystem provides no such service, the API method is your only available option.

Note: This service does not have to be centralized and homogenous, covering the entire SAN, although such a service simplifies the procedure. You may choose to pursue the LUN method even for a fragmented SAN that requires a piecemeal LUN copy across two or more storage subsystems. In such a scenario, you would be responsible for manually managing those multiple backup sets as though they were a single backup set.

Using the LUN-based approach:

To back up the entire SAN File System global namespace in a single operation and to restore the global namespace as a complete namespace, an administrator can use a LUN-based approach. The administrator can use the copy services features that exist in the storage subsystems that SAN File System supports, such as the FlashCopy feature of the IBM TotalStorage Enterprise Storage Server.

When performing a LUN-based backup, an administrator must be sure to back up both the LUNs used as volumes in user storage pools and the LUNs used as volumes in the system storage pool (which is used for metadata) at the same time.

Before performing a LUN-based backup, an administrator must quiesce the Metadata server cluster. This is required for a consistent backup. An administrator can also choose to stop the cluster (from the SAN File System console or by using the **stopcluster** command) before performing a LUN-based backup.

Using the API-based approach:

Considerations for filesets within directories for the API method.:

There are special API method considerations when filesets reside within directories. From the SAN File System client perspective (and therefore from the backup application perspective), a fileset within a directory looks exactly like a regular subdirectory. From the Metadata server and Administrative server perspective, however, this is a fileset that was attached to an arbitrary subdirectory in the global namespace.

The problem is that backup applications generally create subdirectories on-the-fly when attempting to restore files to a directory that does not exist. In the case of SAN File System, the subdirectory being created might have originally been part of another directory. This problem is avoided by recreating all subdirectory connections prior to restoring all the regular subdirectories and files.

The **mkdrfile** command saves state information that can be used to recreate this portion of the global namespace following some disastrous loss of data.

Unfortunately, regular subdirectories can only be created from a client, therefore, the Metadata server cannot recreate directory trees which contain a mix of filesets within subdirectories, and regular subdirectories. To avoid this problem during disaster recovery scenarios, it is strongly recommended that you only attach filesets to the cluster root directory (usually seen as /mnt/sanfs from the client), and to each other. In other words, as a best practice, do not attach filesets to regular subdirectories. The **mkdrfile** output can then be used to completely restore the top of the global namespace tree before using the client-based backup application to restore the rest of the global namespace.

FlashCopy image considerations for the API method:

The .flashcopy subdirectories created when FlashCopy images are made, are read-only. Unless told otherwise, the client backup application would typically backup the .flashcopy subdirectories along with everything else. At restore time, however, the same backup application would attempt, but not be able, to redeposit the original files into these subdirectories. (You would not want them there anyhow since they would appear to be valid FlashCopy images from the client perspective, when in reality, the metadata needed for the original FlashCopy images would be missing).

One obvious consequence of this behavior is that there is no way, using the API method, to restore your original FlashCopy images if you have lost your metadata in a disaster scenario. You only get the original files restored. However, if your backup application supports the ability to restore files to a directory other than their original location (that is, to the grandparent directory two levels above the ./flashcopy/<flashcopyname> directory), then you have all the ingredients for a highly efficient API method backup, which leverages the FlashCopy image feature.

In any case, periodic FlashCopy images are still highly recommended. They are the most efficient method for quickly backing up and restoring files in scenarios where the metadata is still available.

Backing up and restoring system metadata:

SAN File System manages file data and system metadata separately. When a user backs up a file, only the file's data and attributes are backed up. For disaster recovery purposes, an administrator must back up system metadata (which includes information about fileset attachment points, storage pools, volumes, and policies), separately.

An administrator can create a file that contains a backup copy of system metadata from the SAN File System console or by using the **mkdrfile** command. The file, which is stored in the /usr/tank/server/DR directory on the master Metadata server's local disk, contains everything required to re-create the metadata. When needed, an administrator can use the contents of this file (along with normal restore processes for file data) to re-create the state of the cluster.

To restore system metadata, an administrator processes the information contained in the system metadata backup file using the **bulddrscript** command. This command creates several scripts that the administrator must review to obtain a restore scenario, and then run to re-create the SAN File System configuration. Then, user data files can be restored from SAN File System clients.

An administrator should run the **mkdrfile** and **bulddrscript** commands often enough to ensure that any configuration changes are reflected in the output of

these commands. Note that an administrator should put copies of the output of the **mkdrfile** and **builddrscript** commands in an easily recoverable location on backup media where critical system and application files are kept for backup and restore purposes.

Note: To assist in protecting against the corruption of metadata and other metadata failures, an administrator can check metadata from the SAN File System console or by using the **startmetadachek** command. This command performs a consistency check on the system metadata, and optionally repairs any problems it finds. An administrator can check file metadata for one or more filesets, the system metadata, or both. There is also an option to check only the metadata structure, or to check the metadata structure and its contents.

There are three cases when an administrator might use the check or repair operation: 1) as part of a regularly scheduled cycle of preventive maintenance, 2) in response to an alert that recommends that this operation be performed (extra detail may be supplied that specifies the restore option that an administrator must use to salvage the metadata), and 3) if metadata corruption (or any other SAN File System corruption) is suspected. If the check metadata operation cannot resolve the problem, an administrator must perform a full restore of SAN File System, beginning with restoring the metadata. It is critical that **mkdrfile** is run in order to recover from such a situation.

Related topics:

- “Global namespace” on page 29
- “Metadata server” on page 33
- “FlashCopy images” on page 26
- “Storage pools” on page 37
- “Clients” on page 11

Service Alert

Service Alert is a feature that allows IBM service personnel to respond quickly to any problems that occur in your SAN File System server cluster.

When one of the Metadata servers in your server cluster encounters an event that requires a service alert action, it sends a Simple Network Management Protocol (SNMP) datagram to IBM Director, which then parses and converts the trap into a Simple Mail Transfer Protocol (SMTP) e-mail message. The e-mail message is then sent to the customer’s SMTP mail server. Finally, the e-mail message is forwarded to IBM personnel for processing.

Figure 1 on page 10 shows the Service Alert architecture:

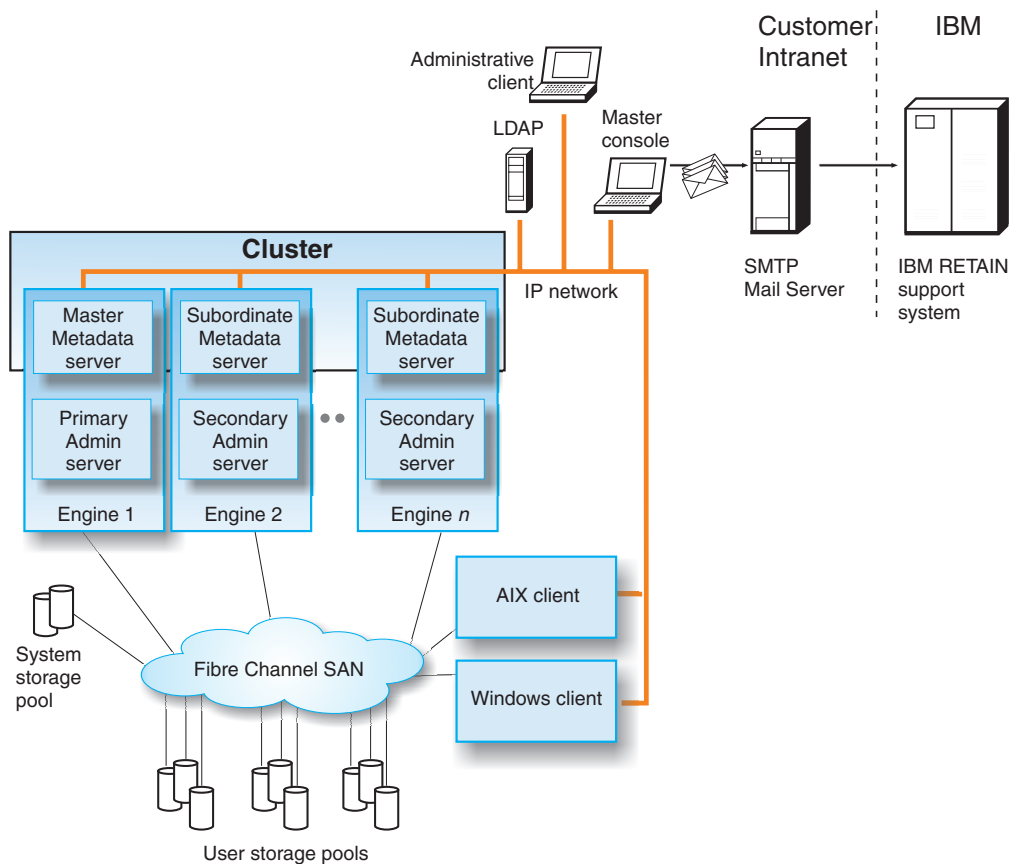


Figure 1. Service Alert architecture

Related topics:

- “Adding SNMP managers” on page 67
- “Alerts” on page 4
- “Master console” on page 32
- “SNMP” on page 42
- “Setting up SNMP traps” on page 70

Common Information Model

The *Common Information Model (CIM)* is a standard for systems management. It specifies a management protocol as well as the schemas for the various subsystems that it can manage.

The SAN File System console (which is a Web-based user interface) and the SAN File System administrative command line interface (tanktool) are built around the Common Information Model. These interfaces use protocols that are compliant with CIM standards to communicate with Metadata servers in a SAN File System server cluster.

When an administrator issues a request, an Administrative agent communicates with an LDAP server (which must be configured and accessible on your network) through a CIM interface to authenticate the administrator’s user ID and password and to verify that the user has the authority to issue a particular request. Each administrator is assigned a user role that allows the administrator a specific level

of access to administrative operations. After authenticating an administrator, the Administrative agent interacts with the Metadata server on behalf of that user to process the request.

The Administrative agent, which also functions as a CIM Object Manager (CIMOM) for SAN File System, allows third-party CIM clients to manage SAN File System.

For more information about CIM, refer to the *System Management API Guide and Reference*, GA27-4315.

Related topics:

- “Publications” on page viii
- “Administrative server” on page 3

Clients

SAN File System is based on a client-server design. A SAN File System *client* is a computer that accesses and creates data that is stored in the SAN File System global namespace. SAN File System uses a cluster of Metadata servers that cooperate with each other to provide the global namespace.

The *SAN File System protocol specification* includes a description of the protocols used between a metadata server and clients running on application servers. It is available at www.ibm.com/storage/software/virtualization/sfs.

Clients access only metadata (such as a file’s location on a storage device) from a Metadata server, and then access data directly from storage devices attached to your high-speed fibre-channel network. This method of data access eliminates server bottlenecks and provides read and write performance that is comparable to that of file systems built on bus-attached, high-performance storage.

SAN File System supports clients that run these operating systems:

- AIX® 5.1 (32-bit only)
- Windows® 2000 Server or Windows 2000 Advanced Server

Client software must be installed on each AIX or Windows client. On an AIX client, the software is a virtual file system (VFS), and on a Windows client, it is an installable file system (IFS). The VFS and IFS provide clients with local access to the global namespace on your SAN. Note that clients can also act as servers to a broader clientele. They can provide NFS or CIFS access to the global namespace and can host applications such as database servers.

A VFS is a subsystem of an AIX client’s virtual file system layer, and an IFS is a subsystem of a Windows client’s file system. A VFS or IFS directs all metadata operations to a Metadata server and all data operations to storage devices attached to your SAN. A VFS or IFS makes the metadata that is visible to a client’s operating system, as well as any applications that the client runs, look identical to metadata read from a native, locally attached file system.

When the global namespace is mounted on an AIX client, it looks like a local file system. When the global namespace is mounted on a Windows client, it appears as another drive letter and looks very much like an NTFS file system. Basically, the SAN File System global namespace looks and acts like any other file system on the client’s system.

Caching metadata, locks, and data:

Caching allows a client to achieve low-latency access to both metadata and data. A client can cache metadata to perform multiple metadata reads locally. The metadata includes the mapping of logical file system data to physical addresses on storage devices attached to your SAN.

A client can also cache locks to allow the client to grant multiple opens to a file locally without having to contact a Metadata server for each operation that requires a lock.

In addition, a client can cache data for small files to eliminate I/O operations to storage devices attached to your SAN. A client performs all data caching in memory. Note that if there is not enough space in the client's cache for all of the data in a file, the client simply reads the data from the shared storage device on which the file is stored. Data access is still fast because the client has direct access to all storage devices attached to your SAN.

Using the direct I/O mode:

Some applications, such as database management systems, use their own sophisticated cache management systems. For such applications, SAN File System provides a direct I/O mode. In this mode, SAN File System performs direct writes to disk, does not cache data, and allows distributed applications on different computers to write data to the same file at the same time. Using the direct I/O mode makes files behave more like raw devices. This gives database systems direct control over their I/O operations, while still providing the advantages of SAN File System, such as the FlashCopy Image feature and file-level backup and restore.

Sharing files:

In a homogenous environment, which means that the clients in your environment are either all AIX or all Windows clients, SAN File System provides access and semantics that are customized for the operating system running on the clients. When files are created and accessed from only Windows clients, all the security features of Windows are available and enforced. When files are created and accessed from only AIX clients, all the security features of AIX are available and enforced.

In a heterogeneous environment, which means that you have both AIX and Windows clients, there is a restricted form of access. When files created on an AIX client are accessed by a Windows client, access is controlled using only the semantics and permissions of the "Other" permission bits in AIX. Similarly, when files created on a Windows client are accessed on an AIX client, access is controlled using only the semantics and permissions of the "Everyone" user group in Windows.

Antivirus software:

If more than one SAN File System client for Windows is running antivirus software that scans directories and files, there is no reason to run this from more than one SAN File System client on the same files. If scans are run from more than one client, they should be scheduled to run at different times to allow better performance of each scan.

Related topics:

- “Components” on page 21
- “Global namespace” on page 29
- “Locks and leases” on page 30
- “AIX” on page 78
- “Windows” on page 79
- “UNIX-based clients”
- “Windows-based clients” on page 14

UNIX-based clients

A *UNIX[®]-based client* is a SAN File System client that runs a UNIX operating system and has the SAN File System client code installed. In this release, SAN File System supports clients running AIX 5.1 (32-bit only).

UNIX-based clients use standard UNIX permission semantics (such as read, write, and execute bits, and owner and group IDs) that make the SAN File System global namespace appear as if it is a local UNIX file system.

The SAN File System client code installed on a UNIX client is called a Virtual File System (VFS). The VFS is a subsystem of an AIX client’s virtual file system layer. It directs all metadata operations to a SAN File System Metadata server and all data operations to storage devices attached to your SAN. A VFS makes the metadata that is visible to a client’s operating system, as well as any applications that run on the client, look identical to metadata read from a native, locally attached file system.

Privileged clients:

A privileged client is a client machine on which root users are given the same privileges on the global namespace as they have on other file systems available on their system. A root user logged in to a privileged AIX client is granted full control over directories, files, and other file system objects created by AIX clients.

If a root user logs in to a client that is not a privileged client, that user’s privileges for the global namespace are reduced to those of “Other.”

Commands:

There is a set of commands (issued from the operating system command line) that is used to set up, start and stop the client and to migrate data. These commands are accessible from each client machine. The client commands are separate from the administrative command line interface. The client commands allow users to perform operations on the client systems, and the administrative command line interface allows administrators to administer all aspects of the SAN File System server.

Constraints and restrictions:

The following are restrictions for AIX client features:

- AIX clients cannot use user IDs or group IDs 999999 and 1000000 for real users or groups; these are reserved IDs used internally by SAN File System.
- Multibyte enablement for items referenced in a file placement policy (storage pools, filesets, parts of file names) is not supported.
- mkfs and mkfifo (named pipes) in AIX are not supported.

Related topics:

- “Clients” on page 11
- “Components” on page 21
- “Global namespace” on page 29
- “Windows-based clients”

Windows-based clients

A *Windows-based client* is a client that runs a Windows® operating system and has the SAN File System client code installed. In this release, SAN File System supports clients running on Windows 2000 Advanced Server.

The SAN File System client code installed on a Windows client is called an Installable File System (IFS). The IFS is a subsystem of a Windows client’s file system. It directs all metadata operations to a SAN File System Metadata server and all data operations to storage devices attached to your storage area network (SAN). An IFS makes the metadata that is visible to a client’s operating system, as well as any applications that run on the client, look identical to metadata read from a native, locally attached file system.

Windows-based clients use a subset of the Windows semantics. The supported semantics are described to Windows as volume properties, which are visible, for example, as properties of the drive within Windows Explorer. The following volume properties are supported by SAN File System:

- NTFS-like access control lists (requires all Windows-based clients to share a common Active Directory domain for users and groups)
- Long names and short names (eight-character names with three-character extensions)
- UNICODE-based file names
- Case-sensitive file names

Case sensitivity:

While the global namespace is case-sensitive, Windows applications can choose to use case sensitive or case insensitive names. This means that case-sensitive applications, such as those making use of Windows support for POSIX interfaces, behave as expected. Native Win32 clients (such as Windows Explorer) get only case-aware semantics. The case specified at time of file creation is preserved, but in general, file names are case-insensitive. For example, Windows Explorer allows the user to create a file named “Hello.c,” but an attempt to create “hello.c” in the same folder will fail because the file already exists. If a Windows-based client accesses a folder that contains two files that are created on a UNIX-based client with names that differ only in case, the behavior depends on whether the application running on the client accesses the global namespace using Win32 APIs or POSIX APIs.

Integration into Windows:

SAN File System differs from NTFS in its degree of integration into the Windows administrative environment. The differences are as follows:

- Disk Management within the Microsoft Management Console shows SAN File System disks as unallocated.
- SAN File System does not support the use of standard Windows write signature on its disks.
- The SAN File System global namespace cannot be assigned a reserved drive letter.

- Disks used for the SAN File System global namespace cannot be allowed to sleep or hibernate.

SAN File System also differs from NTFS in its degree of integration into Windows Explorer and the desktop. The differences are as follows:

- Manual refreshes are required when updates to the SAN File System global namespace are initiated on the Metadata server (such as attaching a new fileset).
- The recycle bin is not supported.
- Distributed link tracing is not supported. This is a technique through which shell shortcuts and OLE links continue to work after the target file is renamed or moved. Distributed link tracking can help a user locate the link sources in case the link source is renamed or moved to another folder on the same or different volume on the same PC, or moved to a folder on any PC in the same domain.
- Sparse file API and change journaling are not supported. This means that SAN File System provides inefficient support for the indexing service accessible through the Windows "Search for files or folders" function.

Privileged clients:

A privileged client is a client machine on which administrative users are given the same privileges on the global namespace as they have on other file systems available on their system. An administrative user logged in to a privileged Windows client is granted full control over directories, files, and other file system objects created by Windows clients.

If an administrative user logs in to a client that is not a privileged client, that user's privileges for the global namespace are reduced to those of "Everyone."

File names:

File names created on UNIX-based clients using characters that are not valid for the Windows file systems (such as colons, slashes, back slashes, asterisks, question marks, double quote mark, less than, greater than, and pipe) are transformed into valid short names. Applications can use the short name to gain access to files.

Commands:

There is a set of commands that is used to set up, start, and stop the client and to migrate data. These commands are accessible from each client machine. The client commands are separate from the administrative command line interface. The client commands allow users to perform operations on the client systems, while the administrative command line interface allows administrators to administer all aspects of the SAN File System server.

Constraints and limitations:

The following features of NTFS are *not* currently supported by SAN File System:

- File compression (for example, a DoubleSpace volume; no support exists for compression on individual files or all files within a folder)
- Encrypted files and directories
- Quotas
- Reparse points (for example, a specialized hook used for certain implementations of hierarchical storage or NTFS junctions)
- Defragmentation and error-checking tools
- Alternate data streams (a concept similar to resource forks in Macintosh)

- Assigning an ACL for the entire drive
- Change journal for file activity
- Scanning all files/directories owned by a particular DID (FSCTL_FIND_FILES_BY_SID)
- Security auditing or SACLs
- Sparse files (as defined by Windows; SAN File System uses sparse allocation techniques, but does not support certain file system APIs required by Windows)

In addition, note these differences:

- Programs that open files using either the 64-bit file ID or the 128-bit object ID (the "FILE_OPEN_BY_FILE_ID" option) will fail. This applies to the NFS server bundled with Microsoft Services for UNIX.
- Symbolic links created on AIX clients are handled specially by SAN File System on Windows clients; they appear as regular files whose contents are not accessible and cannot be deleted.
- SAN File System clients acting as CIFS servers do not support opportunistic locks (oplocks), which is an option used by CIFS clients to improve performance through the use of local cache.

Related topics:

- "Clients" on page 11
- "Components" on page 21
- "Global namespace" on page 29
- "UNIX-based clients" on page 13

Cluster

A SAN File System *cluster* is a set of Metadata servers and engines with one Metadata server running on each engine. The servers in a cluster communicate with each other and with SAN File System clients over your existing IP network. A cluster provides a single point of control for administrative and service operations.

Each cluster has one master Metadata server, which is designated by an administrator, and one or more subordinate Metadata servers. The master server maintains cluster state and is the focal point for most administrative services. The maximum number of Metadata servers and engines that you can have in a cluster is eight.

Note: Although you cannot purchase SAN File System with only one Metadata server and engine, you can run a single engine system if all of the other engines in your cluster have failed (for example, if you have only two engines, and one of them fails), or if you want to bring down all of the engines except one before performing scheduled maintenance tasks. To keep the SAN File System global namespace available, you must set the single remaining active server to be the master server. If the last remaining active server in a cluster is a subordinate server, it eventually goes into a wait state because it cannot contact the master, and the entire global namespace becomes inaccessible.

Server interactions:

The Metadata servers in a cluster interact with each other for a variety of reasons. For example, they exchange *heartbeats*, which are messages sent periodically from one server to another so that each knows that the other is still active. If a server

stops sending heartbeats for a specific period of time (which is set with heartbeat parameters), the other servers reform the server cluster without it.

The master server needs to communicate with its subordinate servers to perform many administrative tasks, such as supplying the servers with the current workload map and querying server status. It also needs to contact them to process requests from administrators to perform tasks such as draining volumes (when an administrator removes a volume from a storage pool) and creating FlashCopy images of filesets.

Subordinate servers need to initiate contact with the master server for specific tasks, such as acquiring more space or obtaining file placement policy information.

Cluster workload:

Each server in a cluster, including the master server, is assigned a *workload*; that is, each server is responsible for providing metadata and locks to clients whenever they request access to data that resides in one or more filesets assigned to that particular server. A *fileset* is a subset of the SAN File System global namespace and serves as the unit of workload for Metadata servers.

During client setup, a client is given the address of one of the Metadata servers in your server cluster for initial contact and server discovery. Then, when the client issues a request to access data, it is automatically directed to the appropriate server to obtain the metadata and locks required to access the data.

Creating and assigning filesets: An administrator can create any number of filesets within the global namespace and assign those filesets to specific servers. Each server receives a workload map from the master server and knows which filesets are assigned to it and to all other servers in the cluster. Fileset assignments are fixed, which means that after an administrator assigns a fileset to a specific Metadata server, it remains assigned to that server until the administrator chooses to reassign it to another Metadata server.

Balancing the workload: An administrator has complete control over how the global namespace workload is distributed. For example, an administrator can choose to assign an equal number of filesets to each server. Assuming that the metadata activity level for each fileset is relatively the same, this balances the workload evenly across all of the servers in the cluster.

To ensure that each server handles a share of the entire workload, the administrator should create at least one fileset for each server. Creating additional filesets provides the administrator with greater flexibility in assigning and reassigning filesets to achieve optimal results.

Files are typically placed in a particular fileset based on some association that the files share, such as all of the files are used by a specific application or are used by a specific group. When distributing filesets among Metadata servers, an administrator can consider whether there are some filesets in which many files are constantly being created, deleted, and updated, thus requiring numerous and frequent metadata updates. If several of these filesets are identified, the administrator can choose to distribute them appropriately among the servers to help further balance the workload.

Handling server and engine failures:

How SAN File System handles a failure within a server cluster depends on whether it is a soft or a hard failure.

Soft failures: A soft failure requires no administrator intervention and is recoverable by simply restarting the server software.

Soft failures are detected in different ways depending on their cause. The Administrative server, which runs on the same engine as the master Metadata server and in standby mode on each subordinate server, provides an optional Metadata server restart service that monitors the Metadata server software and restarts it if necessary. If a Metadata server hangs, an internal server thread detects that condition and enables the rebooting of the software. If the operating system on an engine crashes or hangs, SAN File System can reboot the operating system. Then, the Metadata server restart service is automatically started, and in turn, restarts the Metadata server on the engine. When a Metadata server is restarted after a soft failure, it resumes serving the same workload that it was serving before the failure.

If a subordinate Metadata server fails, only the filesets that are assigned to that server are temporarily unavailable. If the master Metadata server fails, all of the subordinate servers eventually pause, either because they require a service from the master server or because they are no longer receiving a heartbeat from the master server, and the entire global namespace is unavailable until the master server is restarted.

When a client has an active session with a Metadata server and that server fails, the client stops receiving responses for transactions and lease renewal attempts. However, the client remains active. The client loses its lease (which is the amount of time a client can hold its locks on data before contacting the server again to renew its lease) and any locks it had obtained from that server. After the server is restarted, the client can contact the server and renew its lease. Then, it can reassert its locks (get back all the locks it had before the server failure) and refresh its metadata cache as needed. A Metadata server provides a grace period for lock reassertion to allow clients to reassert their locks before allowing other clients to obtain new ones. Applications running on the client experience a pause in service during the restart and recovery period.

Hard failures: A hard failure is one that requires intervention by an administrator, as in the case of a hardware failure. A hard failure is detected through a heartbeat timeout. When a Metadata server stops sending heartbeats for the amount of time specified with heartbeat parameters, an SNMP trap message is generated (if an administrator has set all SNMP-related parameters appropriately and the failure is graceful). An administrator must verify that the server is down and take actions to restart it.

The server must fail gracefully in order to generate an SNMP trap to announce its failure. If the server does not fail gracefully, an operational server will detect that the server failed and generate an SNMP trap reporting the problem. An SNMP trap will not be generated if the failing server is the last server and does not fail gracefully.

An administrator can repair the failed engine and bring it back into service with recovery similar to that of a soft failure. Or, if the problem is more complex, the administrator can reassign the filesets that belong to the server on the failed engine to other Metadata servers in the cluster and continue operations while the failed

engine is repaired. Note that if the failed engine is the one on which the master server resides, the administrator must also set a new master server.

Note: An administrator can reassign the filesets that belong to a Metadata server only when that server is down, and can designate a new master server only if the current master server is down (the SAN File System cluster state is Offline). To ensure that an engine and its Metadata server are down, an administrator can issue a **stopengine** command. This command forcefully shuts down a designated engine.

Related topics:

- “Administrative server” on page 3
- “Changing active cluster states” on page 82
- “Components” on page 21
- “Configuring cluster tuning” on page 83
- “Engines” on page 23
- “Filesets” on page 24
- “Global namespace” on page 29
- “Locks and leases” on page 30
- “Metadata server” on page 33
- “Starting the cluster” on page 84
- “Stopping the cluster” on page 84

Cluster states

SAN File System cluster states can be active, inactive, or unknown.

Table 2. SAN File System cluster states

State type	State name	State description
Inactive	Not running	The cluster cannot perform any functions because none of the servers in the cluster have completed the start up procedure to the point where the cluster is detected as running, (that is, in the “Not added” server state).
	Forming	The cluster has a cluster master and is in the process of forming. This state is always the initial one whenever a cluster is newly formed.

Table 2. SAN File System cluster states (continued)

State type	State name	State description
Active	Offline	<p>One or more servers in the cluster are in the “Offline” state. This state terminates all current client sessions and prevents new client sessions from being started, but still allows all server I/O. The offline state is used for restricting the cluster from client access.</p> <p>Characteristics:</p> <ul style="list-style-type: none"> • No cluster configurations are allowed in this state. Configuration changes will fail. • Metadata activity stops. • Existing locks are maintained. • File extensions (growing a file) are not allowed during this time, which guarantees a consistent metadata view for LUN-based backup. • Only existing USER data blocks from client side can be modified and assume client-application based techniques are used to make that data consistent later. • SAN File System FlashCopy is not available. • Administration command line interface client locks are revoked, caches flushed, and so on.
	Fully quiescent	<p>One or more servers in the cluster are in the “Fully Quiescent” state. Quiescence puts the cluster in a “quiet” client communications mode to allow other operations to occur, such as configuring a cluster or creating a FlashCopy image. Full quiescence cuts off all client communications with the cluster.</p> <p>Characteristics:</p> <ul style="list-style-type: none"> • No cluster configurations are allowed in this state. Configuration changes will fail. • SAN File System FlashCopy is not available. • Administration command line interface client locks are revoked, caches flushed, and so on.
	Partly quiescent	<p>One or more servers in the cluster are in the “Partly Quiescent” state. Quiescence puts the cluster in a “quiet” client communications mode to allow other operations to occur, such as configuring a cluster or creating a FlashCopy image. Partial quiescence allows existing metadata activity and client communication to continue, but prohibits new communication.</p>
	Online	<p>All servers in the cluster are in the “Online” state, meaning that the cluster has fully initialized all of its subsystems, and is serving client requests.</p>
Unknown	Unknown	The cluster state cannot be determined.

Related topics:

- “Cluster” on page 16
- “Creating a FlashCopy image” on page 107
- “Managing the cluster” on page 82

- “Metadata server” on page 33
- “Metadata server states” on page 35

Components

A *component* is one of the elements that make up the entire SAN File System. The following figure illustrates the major SAN File System software components.

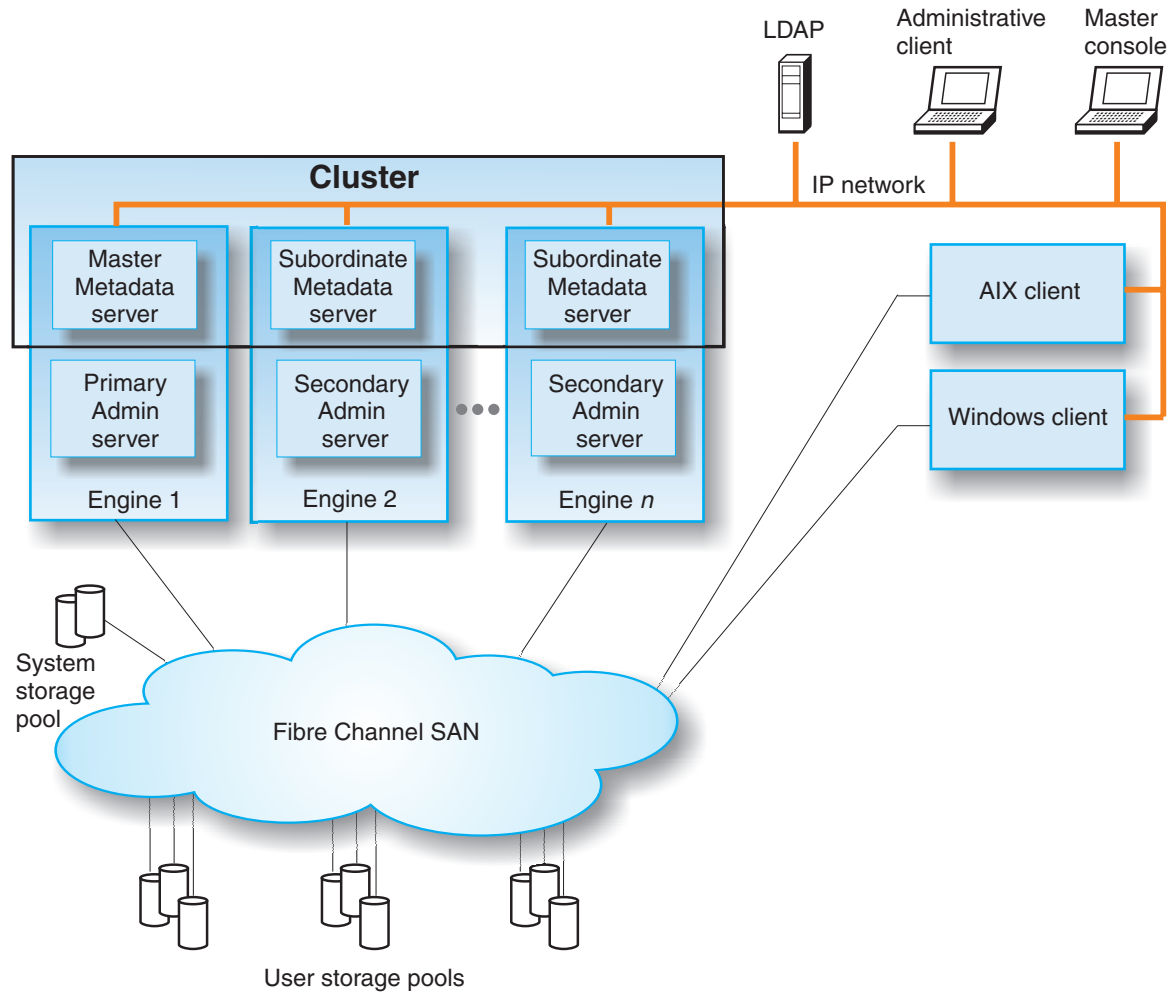


Figure 2. SAN File System components

The Metadata servers and clients communicate over an internal IP network and access data over the SAN.

Note: SAN File System relies on preexisting networking hardware, including an IP network, SAN, network switches, and routers.

A *Metadata server* is a server that runs on a SAN File System engine and performs metadata, administrative, and storage management services. Multiple Metadata servers form a server *cluster*. In a SAN File System server cluster, there is one master Metadata server and one or more subordinate Metadata servers, each running on a separate engine in the cluster. Together these Metadata servers provide clients with shared, coherent access to the SAN File System *global namespace*.

The *metadata* resides on private storage that is shared among all the Metadata servers in the cluster.

The *Administrative server* allows SAN File System to be remotely monitored and controlled through a Web-based user interface, called the *SAN File System console*, using either Netscape 6.2 and later or Internet Explorer 6.0 and later. The Administrative server also processes requests issued from the administrative command line interface, which is called *tanktool*. The Administrative server uses an *LDAP server*, also connected to the internal IP network, to look up authentication and authorization information about the administrative users. The primary Administrative server runs on the same engine as the master Metadata server. It receives all requests issued by administrators and also communicates with the Administrative servers that run on each additional server in the cluster to perform routine requests.

The *clients* access the global namespace through a virtual or installable file system, which is installed on the client machine. These clients can act as servers to a broader clientele, providing Network File System (NFS) or Common Internet File System (CIFS) access to the global namespace or hosting applications (such as database servers or Web-hosting services that use multiple servers).

Related topics:

- “Administrative server” on page 3
- “Clients” on page 11
- “Cluster” on page 16
- “Engines” on page 23
- “Metadata server” on page 33

Container

The term “container” is being phased out of SAN File System in favor of the term “fileset.” “Container” still appears in command names, messages, and other places, although the publications use the newer term “fileset” wherever possible. The term “container” means the same as the term “fileset.”

Related topics:

- “Filesets” on page 24

Data migration

Data migration is the process of moving your data from a source file system to the SAN File System global namespace.

To enable you to use your existing data in SAN File System, a service technician must migrate your data from a source file system to the global namespace. SAN File System provides a data migration utility to help migrate your data quickly and efficiently, while preserving the attributes (such as owner, group, and creation time) of your files. This utility is optimized for bulk data movement of a few large files. It is not intended to replace system utilities, such as tar and cp, for migrating large numbers of small files.

The data migration utility migrates only the following objects. All other objects in the existing file system are ignored.

- Regular files

- Directories
- Symbolic links (for AIX clients only)

The data migration utility attempts to preserve all hard links to regular files. For AIX clients, if SAN File System is set up such that hard links traverse across a fileset, the hard links are replaced with a symbolic link.

SAN File System does not support file-system-based compression (as with NTFS). However, SAN File System supports file-based compression (for example, files produced by utilities such as compress on AIX systems and ZIP utilities on systems running the Microsoft® Windows® operating system). During the migration process, all compressed files are expanded.

Currently, migrating data using the data migration utility is a *disruptive* process. This means that, to guarantee data integrity, you must stop all applications and users from modifying the data being migrated (including database and application servers) until the migration is complete. Only the data being migrated must remain unchanged. To minimize the impact of a migration, a service technician can migrate your data in chunks rather than all at one time.

Note: The data migration utility does not prevent anyone from modifying (for example, editing, moving, or deleting) the data being migrated. You must make sure that no one modifies this data.

If you are migrating an IBM DB2® environment, the procedures vary depending on whether your environment is file-system based or contains raw configuration devices. For a file-system-based environment, a service technician can use the data migration utility to migrate your files, and then reconfigure DB2 to reflect the data movement. For raw configuration devices, the service technician must use the DB2 **unload** command to move data out of the raw devices to a temporary holding location, and then perform a **load** operation to place the data in its location in the SAN File System global namespace. After the data is loaded into the global namespace, it is file based.

The data migration utility is run on a client machine using the client command line interface (CLI). Data from a Windows-based application must be migrated from a Windows-based client. Likewise, data from an AIX application must be migrated from an AIX client.

Related topics:

- “Filesets” on page 24
- “Data-migration prerequisites” on page 437
- Chapter 4, “Migrating data”, on page 63
- “Clients” on page 11

Engines

Within SAN File System, an *engine* is the hardware (which is based on the IBM xSeries® platform) on which a Metadata server and an Administrative server run. Each engine comes preloaded with SAN File System server software and must be configured by onsite IBM personnel. SAN File System supports from two to eight Model 1RX engines.

Note: Although you cannot purchase SAN File System with only one engine, you can run a single-engine system if all of the other engines fail (for example, if

you have only two engines, and one of them fails), or if you want to bring down all of the engines except one before performing scheduled maintenance tasks.

The administrative infrastructure on each engine allows an administrator to monitor and control SAN File System from a standard Web browser or an administrative command line interface. The two major components of the infrastructure are an Administrative agent, which provides access to administrative operations, and a Web server that is bundled with the console services and servlets that render HTML for the administrative browsers. The infrastructure also includes a Service Location Protocol (SLP) daemon, which is used for administrative discovery of SAN File System resources by third-party Common Information Model (CIM) agents, and an IBM Director agent, which detects when the operating system “hangs” or stops unexpectedly and can reboot the operating system.

An administrator can use the SAN File System console, which is the Web-based user interface, or administrative commands to monitor and control an engine.

Related topics:

- “Metadata server” on page 33

Filesets

A *fileset* (except for the global fileset) is a subset of the entire SAN File System global namespace. It serves as the unit of workload for Metadata servers and is also the unit that an administrator specifies to create FlashCopy images that are used in backup procedures.

Creating and attaching filesets:

A fileset has the following properties:

- A fileset name
- A directory path leading to the directory within which the fileset is attached
- A directory name that the fileset is given at the end of the directory path

The fileset attach point is the path formed by combining the directory path and the directory name.

The root of the global namespace is the global fileset. The name of the global fileset is always ROOT. The directory name of the global fileset is specified at SAN File System setup as `sanfs`.

The global fileset is attached to the root level in the namespace hierarchy. An administrator creates filesets and attaches them at specific locations below the global fileset. An administrator can attach a fileset to the global fileset (by specifying a directory path of `sanfs`). An administrator can also attach a fileset to another fileset (for example, by specifying a directory path of `sanfs/fileset1`, where `sanfs/fileset1` is the attach point of the parent fileset). When a fileset is attached to another fileset, it is called a *nested fileset* or a child of a parent fileset.

When creating a fileset, an administrator can also specify a maximum size for the fileset (called a *quota*) and specify whether SAN File System should generate an alert if the size of the fileset reaches or exceeds a specified percentage of the maximum size (called a *threshold*).

Note: The space used by a fileset includes the space used by FlashCopy images. It does not include the space used by any filesets nested within it.

An administrator can detach a fileset and reattach it at the same location or at a different location. If a fileset is reattached at a different location, all the files contained in the fileset are rooted to the new location without any further operations. Note that before a fileset can be detached, any nested filesets must be detached first.

Assigning filesets to Metadata servers:

When creating a fileset, an administrator assigns it to a specific Metadata server in your SAN File System server cluster. That Metadata server is then responsible for providing metadata and locks to clients when they request access to files that reside in that fileset. The fileset-to-server assignment is automatically communicated between SAN File System clients and servers. The client transparently discovers which server to deal with when accessing files on a fileset.

An administrator should create and attach at least one fileset for each Metadata server in your server cluster. However, creating more filesets gives the administrator greater flexibility in distributing filesets among servers to maintain availability and to balance the workload.

Note: An administrator can assign a nested fileset to a different Metadata server than the one to which its parent fileset is assigned. However, if the server to which the parent fileset is assigned becomes inactive, the parent fileset and all of its nested filesets become unavailable until the server comes back up or an administrator reassigns the parent fileset to an active server.

Creating objects in filesets:

From a client perspective, a fileset appears to be a regular directory. Users and applications on SAN File System clients can create objects, such as directories and files, within the fileset.

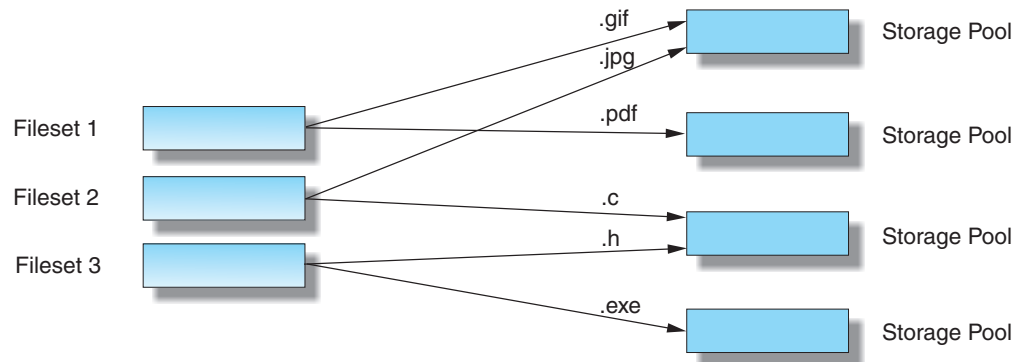
A fileset must be attached to the global namespace and assigned to an active Metadata server before it is available for use by clients. If an administrator detaches an existing fileset, it becomes unavailable to clients until it is attached again. If the Metadata server to which the fileset is assigned becomes inactive, the fileset is unavailable to clients until the server comes back up or an administrator reassigns the fileset to another Metadata server.

Note that users cannot create hard links across fileset boundaries. In addition, a user cannot rename, move, or delete a directory that is the root of a fileset. If a user attempts to perform any of these operations, SAN File System issues an error message.

Placing files in storage pools:

A *storage pool* is a collection of SAN File System volumes that can be used to store either metadata or file data. A fileset can contain files that reside in more than one storage pool. A file is placed in a specific storage pool based on rules in a policy that are used for automatic file placement. There can also be files from more than one fileset in a storage pool. The following illustration shows an example of the

relationship between filesets and storage pools.



Note that when creating file placement policies, an administrator can specify that all files created in a particular fileset are to be stored in a specific storage pool.

Related topics:

- “Alerts” on page 4
- “Attaching a fileset” on page 103
- “Clients” on page 11
- “Creating a fileset” on page 104
- “FlashCopy images”
- “Global namespace” on page 29
- “Listing filesets” on page 106
- “Policies and rules” on page 44
- “Storage pools” on page 37
- “Viewing fileset details” on page 106
- “Viewing fileset settings” on page 106

FlashCopy images

A *FlashCopy[®] image* is a space-efficient, read-only copy of the contents of a fileset in a SAN File System global namespace at a particular point in time. A FlashCopy image can be used with standard backup tools available in your environment to create backup copies of files on tape.

Creating FlashCopy images:

When creating FlashCopy images, an administrator specifies each fileset to be included; the FlashCopy image feature does not automatically include nested filesets. The FlashCopy image operation is performed individually for each fileset. Also note that an administrator cannot create “incremental” FlashCopy images to be used as the basis for incremental backups. A FlashCopy image is simply an image of an entire fileset as it exists at a specific point in time. Note that while a FlashCopy image is being created, all data remains online and available to users and applications.

FlashCopy images are available to clients in a special subdirectory named `.flashcopy` under the root directory of a fileset as shown below.

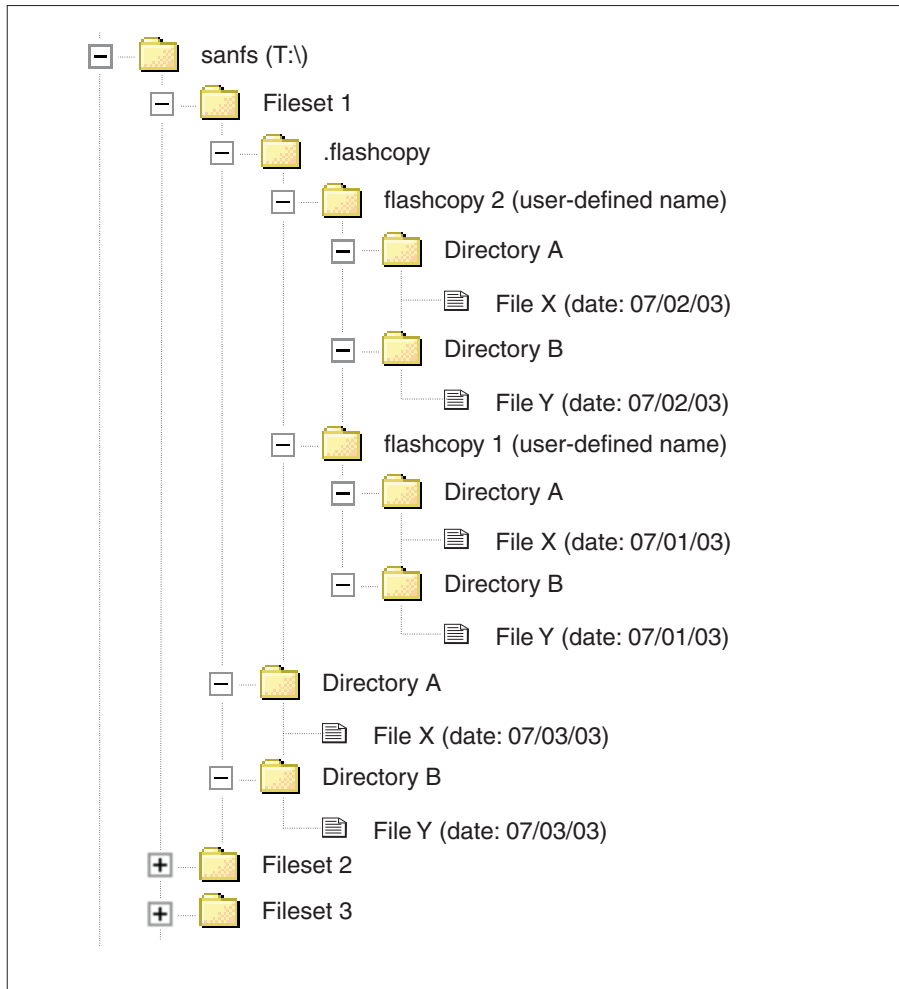


Figure 3. FlashCopy images

In this case, two FlashCopy images, flashcopy 2 and flashcopy 1, have been created under the special .flashcopy directory in Fileset 1. All directories (A, B) and files (X, Y) beneath flashcopy 1 and flashcopy 2 are read-only versions with all other attributes preserved from the original directory or file. Note that flashcopy 1 was created on 07/01/03, and flashcopy 2 was created on 07/02/03. Each reflects the contents of the files on those dates, while the original files have since been updated on 07/03/03.

A fileset can have as many as 32 read-only FlashCopy images. When creating a FlashCopy image for a fileset, the administrator can indicate whether the oldest image should be deleted if creating a new one causes the maximum number of images to be exceeded. Note that once a FlashCopy image is created, its name cannot be changed.

An administrator can list detailed information, such as the following, about each FlashCopy image:

- Its unique name
- Associated fileset name
- Associated fileset state (attached or detached)
- Directory name under which the FlashCopy image is stored
- Directory path for the FlashCopy image

- Description of the FlashCopy image
- Recency (range from 1 to 32)
- Size
- Description

Note: The actual files in a fileset and the FlashCopy images of the files in the fileset share the same file data blocks until a client makes changes to the files. When a client makes a change to a file, such as adding or deleting data, the client performs an operation called *copy on write*. It writes the changed blocks to a new location on disk. At this point, the FlashCopy image points to the old blocks, and the actual file points to the blocks with the new data. Therefore, any access to the FlashCopy image accesses the data blocks as they existed when the FlashCopy image was created, and any access to the actual file accesses the new data blocks.

Backing up files using FlashCopy images:

When performing a backup using standard backup tools available in your environment, a client can specify the path to the FlashCopy image instead of the path to the actual files and continue working with the files while the backup occurs. This procedure produces a consistent backup of the files in the fileset.

Reverting files to a previous FlashCopy image:

Although creating FlashCopy images is *not* a replacement for creating backups of your files to protect your data, in some cases, a user can choose to use a FlashCopy image to revert a file or a set of files to a specific point in time. For example, if a user accidentally deletes a file, restoring it by copying it from a FlashCopy image to another directory instead of restoring it from a backup copy can be faster.

When you restore files from a backup taken from a FlashCopy image, you cannot restore the files to the same location – all FlashCopy image directories are read-only directories. You must restore the files to the directory where the original files resided or to another directory.

Note: For ease of management, it is recommended that you create FlashCopy Images of all filesets at the same point in time, and use a common naming convention to indicate that they represent a set.

Related topics:

- “Backup and restore” on page 5
- “Creating a FlashCopy image” on page 107
- “Deleting a FlashCopy image” on page 108
- “Filesets” on page 24
- “Global namespace” on page 29
- “Listing FlashCopy images” on page 108
- “Reverting to a previous FlashCopy image” on page 108
- “Viewing FlashCopy image details” on page 109

Global namespace

The SAN File System *global namespace* allows shared access to files for both AIX and Windows clients. It ensures that all SAN File System clients have consistent access and a consistent view of the data and files managed by SAN File System.

How the global namespace is organized:

The global namespace is organized into filesets that directly correspond to the namespace's directories and files. Each fileset is available to the client-accessible global namespace at a fileset's attach point. An administrator is responsible for creating filesets and attaching them to directories in the global namespace, which can be done at multiple levels. For instance, the global fileset (for example, ROOT) is attached to the root level in the namespace hierarchy (for example, sanfs), and other filesets are attached at a lower level (for example, sanfs/Notes_ap1/Raleigh_ap3). By defining the path of a fileset's attach point, the administrator is also automatically defining its nesting level in relationship to the other filesets.

How clients access the global namespace:

SAN File System clients mount the global namespace on their systems. After the global namespace is mounted on a client, users can use it just as they do any other file system to access data and to create, update, and delete directories and files.

On an AIX client, the global namespace looks like a local UNIX file system. On a Windows client, it appears as another drive letter and looks similar to an NTFS file system. Basically, the global namespace looks and acts like any other file system on a client's system.

Shared access:

In a homogenous environment (an environment in which all clients run the same operating system), SAN File System provides access and semantics that are customized for the operating system running on the clients. For example, when files are created and accessed from only Windows clients, all the security features of Windows are available and enforced. When files are created and accessed from only AIX clients, all the security features of AIX are available and enforced.

In a heterogeneous environment (an environment in which clients run more than one type of operating system), there is a restricted form of access. For example, when files created on an AIX client are accessed by a Windows client, access is controlled using only the semantics and permissions of the "Other" permission bits in AIX. Similarly, when files created on a Windows client are accessed on an AIX client, access is controlled using only the semantics and permissions of the "Everyone" user in Windows.

Related topics:

- "Clients" on page 11
- "Filesets" on page 24
- "Security" on page 39

Locks and leases

SAN File System uses locks and leases to ensure the consistency and integrity of data in the SAN File System global namespace. The internal locks discussed are in addition to the locks provided with the native file systems, such as flock().

A *lock* is a mechanism that restricts access to data and metadata. The SAN File System protocol provides locks that enable file sharing among SAN File System clients, and, when necessary, provides locks that allow clients to have exclusive access to files. It uses distributed data locks for cache consistency and file access locks to synchronize multiple, concurrent open instances of the same file. SAN File System supports locking semantics that correspond to open modes that are native to Windows and UNIX operating systems. It also supports byte-range locks.

When a client or server fails, SAN File System uses a lease-based safety protocol to ensure data consistency and to protect the structural integrity of the global namespace.

A client obtains a *lease* from a Metadata server as soon as it makes contact with that server. A lease is valid for the period of time set by an administrator using a Metadata server configuration parameter. When a client obtains a lock from a server, that lock is guaranteed to be valid by the server only as long as the client has a valid lease with the server. The server renews a client's lease each time the client contacts the server.

If a client does not contact the server within the specified lease period (for example, because of a temporary network partition), the server can revoke the client's locks. If other clients request locks on the same data, the server revokes the first client's locks and grants new locks to the other clients. If no such requests are made, when the client contacts the server again, it can renew its lease and reassert any locks (get its old locks back) that protect modified but uncommitted data in the client's cache, thus preventing data loss.

A client can also lose its lease because of a server failure. However, when the Metadata server is restarted, the client can renew its lease and reassert its locks. A Metadata server provides a grace period for lock reassertion to allow clients to reassert their locks before allowing other clients to obtain new ones. Clients cannot access any new data until the grace period has ended.

Related topics:

- "Metadata server" on page 33
- "Clients" on page 11

Logs

A *log* is file that contains a record of activity. SAN File System provides the following logs:

- Cluster log
- Event log
- Administrative log
- Audit log
- Security log
- Administrative agent message log

An administrator can view these logs from the SAN File System console or by using the **catlog** command, and can clear the cluster and audit logs using the **clearlog** command.

Cluster log:

The cluster log contains entries from all of the Metadata servers in a cluster. A separate log is maintained on each Metadata server. However, when an administrator issues a request to view the log, entries from all of the separate logs are merged and ordered by date and time to provide the administrator with a cluster-wide view of activities and events.

Event log:

The event log contains a subset of the entries in the cluster log. The entries that appear in the event log are those events, such as changes in server state, that have been configured as alerts. The event log is not stored in a separate physical file, but is generated from entries in the cluster log when a SAN File System administrator issues a request to view the log.

Administrative log:

The administrative log maintains a history of messages generated by the Administrative servers in a cluster. A separate log is maintained on each Administrative server. However, when an administrator issues a request to view the log, entries from all of the separate logs are merged into a single log.

Audit log:

The audit log is located on the master Metadata server engine, and merges with the existing file when the master Metadata server engine is switched. It contains entries for commands issued by administrators (from either the command line interface or the SAN File System console), ordered by date and time. The user ID of the administrator who issued a particular command is included in each entry.

Note: Generally, only commands sent to the Metadata server that change the metadata, cluster configuration, or are significant operations are recorded in the audit log. These commands can be sent using the GUI, CLI, or CIM. For example, list and stat functions are not logged because they do not change the system. Engine functions are not logged because they are not implemented in the metadata server, but making filesets, pools, and metadata checks are logged.

The exception is FlashCopy commands. These commands such as mk, rm, and revert, are logged here even though they do not change the SAN File System configuration. Queries made by a SAN File System administrator are not recorded.

SAN File System administrators can use information in the audit log to help them convert requests made from the SAN File System console into equivalent command line interface instructions or to perform troubleshooting in the case of a failure within the SAN File System system.

Security log:

The security log maintains a history of administrator login activity. A separate log is maintained on each Administrative server. However, when an administrator issues a request to view the log, entries from all of the separate logs are merged into a single log.

Related topics:

- “Viewing logs” on page 71

Master console

The *master console* is a serviceability node for SAN File System and other IBM TotalStorage products. SAN File System includes key features that facilitate integration with the master console. In particular, the master console provides key infrastructure for the remote access and service alert features.

The master console for SAN File System is an IBM @server xSeries 305 Type 8673 Model RA1 (and is ordered as a feature of the SAN File System). It uses the following software:

- Microsoft Windows 2000 Advanced Server edition
- IBM Director Server, version 4.1
- IBM Tivoli® Bonus Pack for SAN Management
- Adobe Acrobat, version 5.0
- The PuTTY openssh package

From the master console, you can access the following components:

- SAN File System console through a Web browser.
- Administrative command-line interface through a Secure Shell (SSH) session.
- Any of the engines in the SAN File System cluster through an SSH session.
- The RSA II card for any of the engines in the SAN File System cluster through a Web browser. In addition, you can use the RSA II Web interface to establish a remote console to the engine, allowing you to view the engine desktop from the master console.
- Any of the SAN File System clients through an SSH session, a telnet session, or a remote display emulation package (such as VNC), depending on the configuration of the client.

Typically, you will use the master console to access the engines as well as the SAN File System console or the Administrative command-line interface. However, if necessary, you can attach a keyboard, monitor, and mouse to an engine.

Using the Remote Access feature of the SAN File System, you can initiate a VPN connection to allow a support engineer to remotely access the master console. You can monitor that access and disconnect the session at any time.

Related topics:

- “Service Alert” on page 9
- “Remote access” on page 39

Metadata server

A *Metadata server* is a server that runs on a SAN File System engine and performs metadata, administrative, and storage management services. In a SAN File System server cluster, there is one master Metadata server and one or more subordinate Metadata servers, each running on a separate engine in the cluster. Together these Metadata servers provide clients with shared, coherent access to the SAN File System global namespace.

All of the servers, including the master Metadata server, share the workload of the SAN File System global namespace. Each is responsible for providing metadata and locks to clients for specific filesets assigned to them by an administrator. They know which filesets belong to which server, and when contacted by a client can direct the client to the appropriate server. They manage distributed locks to ensure the integrity of all of the data within the global namespace.

In addition to providing metadata to clients and managing locks, Metadata servers perform a wide variety of other tasks. They process requests issued by administrators to create and manage filesets, storage pools, volumes, and policy sets, and they enforce the policies defined by administrators to place files in appropriate storage pools and ensure that capacity quotas established for filesets and storage pools are not exceeded.

Performing Metadata service:

There are two types of metadata:

- *File metadata*, which is information that clients need to access files directly from storage devices on your storage area network. File metadata includes permissions, owner and group, access time, creation time, and other file characteristics.
- *System metadata*, which is metadata used by the system itself. System metadata includes information about filesets, storage pools, volumes, and policies. It is stored and managed in a separate system storage pool that is only accessible by the Metadata servers in your server cluster. The Metadata servers perform the reads and writes required to create, distribute, and manage this information.

Distributing locks to clients involves the following:

- Issuing leases that determine the length of time that a server guarantees the locks that it grants to clients.
- Granting locks to clients that allow them shared or exclusive access to files or parts of files. These locks are semi-preemptible, which means that if a client does not contact the server within the lease period, the server can “steal” the client’s locks and grant them to other clients if requested; otherwise, the client can reassert its locks (get its locks back) when it can contact the server again.
- Providing a grace period during which a client can reassert its locks before other clients can obtain new locks if the server itself goes down and then comes back online.

Performing administrative services:

A Metadata server processes requests from administrators (issued from the SAN File System console or by using administrative commands) to perform the following types of tasks:

- Create and manage filesets, which are subsets of the entire global namespace and serve as the units of workload assigned to specific Metadata servers.
- Create and manage volumes, which are LUNs labeled for SAN File System's use in storage pools.
- Create and maintain storage pools (for example, an administrator can create a storage pool that consists of RAID or striped storage devices to meet reliability requirements, and can create a storage pool that consists of random-access or low-latency storage devices to meet high performance requirements).
- Create FlashCopy images of filesets in the global namespace that can be used to make file-based backups easier to perform.
- Define policy sets that contain rules that determine in which storage pools specific files are stored.

Performing storage management services:

A Metadata server performs these storage management services:

- Manages allocation of blocks of space for files on LUNs
- Maintains pointers to the bits of a file that constitute the file
- Evaluates the rules in the active policy set and manages the placement of files in specific storage pools based on those rules
- Issues alerts when filesets and storage pools reach or exceed their administrator-specified thresholds, or returns out-of-space messages if they run out of space

Using configuration parameters:

Each Metadata server has a configuration file that is stored on local storage for each engine in a Metadata server cluster. This configuration file contains settings, which are specified by an administrator, for the following:

Space reclamation interval

Controls how often the master Metadata server reclaims free storage pool partitions allocated by filesets.

Lease period

Specifies the maximum length of time that a server guarantees the validity of the locks it has granted to a client. The client must contact the server before the lease period ends to renew its lease and retain its locks.

List of privileged clients

Lists the clients on which users specified as root or administrator users can have those same privileges on the SAN File System global namespace.

List of SNMP managers

Specifies the IP address, port, version (SNMPv1 or SNMPv2c) and community strings for SNMP trap recipients.

SNMP trap setting

Specifies the filter used on the cluster message log to determine which messages (informational, warning, error, and severe) are also issued as SNMP trap messages.

Related topics:

- "Alerts" on page 4
- "Clients" on page 11
- "Cluster" on page 16

- “Engines” on page 23
- “FlashCopy images” on page 26
- “Locks and leases” on page 30
- “Policies and rules” on page 44
- “Starting a Metadata server” on page 112
- “Storage devices” on page 46
- “Storage pools” on page 37
- “SNMP” on page 42
- “Storage management” on page 43
- “Components” on page 21
- “Volumes” on page 49

Metadata server states

SAN File System Metadata servers present different states depending on whether they are available or unavailable, or whether or not they are part of a cluster.

Table 3. SAN File System Metadata server states

State type	State name	State description
Unavailable	Not running	The server is not running and cannot perform any functions.
	Failed initialization	A fatal error occurred during initialization and the server is suspended. The server remains suspended until an administrator can fix the problem that occurred during initialization.
Available, unclustered	Initializing	The server is running, but has not yet opened its communication ports.
	Not added	The server has not yet been added to the cluster. Because the master server does not know about servers that have not yet been added, this state is only available by logging into that server.
	Joining	The server is joining the cluster (has activated its communication).

Table 3. SAN File System Metadata server states (continued)

State type	State name	State description
Available, clustered	Offline	<p>This state terminates all current client sessions and prevents new client sessions from being started, but still allows all server I/O. The offline state is used for restricting the cluster from client access.</p> <p>Characteristics:</p> <ul style="list-style-type: none"> • No cluster configurations are allowed in this state. Configuration changes will fail. • Metadata activity stops. • Existing locks are maintained. • File extensions (growing a file) are not allowed during this time, which guarantees a consistent metadata view for LUN-based backup. • Only existing USER data blocks from client side can be modified and assume client-application based techniques are used to make that data consistent later. • SAN File System FlashCopy is not available. • Administration command line interface client locks are revoked, caches flushed, and so on.
	Fully quiescent	<p>Quiescence puts the server in a “quiet” client communications mode to allow other operations to occur, such as configuring a server or creating a FlashCopy image. Full quiescence cuts off all client communications with the server.</p> <p>Characteristics:</p> <ul style="list-style-type: none"> • No cluster configurations are allowed in this state. Configuration changes will fail. • SAN File System FlashCopy is not available. • Administration command line interface client locks are revoked, caches flushed, and so on.
	Partly quiescent	<p>Quiescence puts the server in a “quiet” client communications mode to allow other operations to occur, such as configuring a server or creating a FlashCopy image. Partial quiescence allows existing metadata activity and client communication to continue, but prohibits new communication.</p>
	Online	<p>The server has fully initialized all of its subsystems, is a member of a cluster, and is serving client requests.</p>
Unknown	Unknown	The server state cannot be determined.

Properties:

Related topics:

- “Cluster” on page 16
- “Cluster states” on page 19
- “Creating a FlashCopy image” on page 107
- “Managing Metadata servers” on page 109
- “Metadata server” on page 33

Storage pools

A *storage pool* is a collection of SAN File System volumes that can be used to store either metadata or file data. A storage pool typically contains a set of volumes that provide a desired quality of service for a specific use, such as to store all files for a particular application or a specific business division. Note that an administrator must assign one or more volumes to a storage pool before it can be used.

Understanding storage pool types:

SAN File System includes the following three types of storage pools:

System storage pool

The system storage pool contains the system metadata (system and file attributes, configuration information, and Metadata server state) that is accessible to all Metadata servers in your server cluster. There is only *one* system storage pool.

The system storage pool is created when SAN File System is installed. However, an administrator must assign one or more volumes to this storage pool before it can be used.

Note: The first volume assigned to the system storage pool is called the *master volume*. This volume contains the master dbspace within SAN File System, which contains the most critical pages of metadata that SAN File System manages. Internal to the server, the dbspace is organized into various structures that are used by the subsystems within the Metadata servers. It is important to use highly reliable and available LUNs as volumes within the system storage pool so that the server always has a robust copy of this and other system metadata available at all times.

Because the amount of metadata grows as the global namespace grows, you must monitor the system storage pool to ensure that there is always enough volumes assigned to it to accommodate the growth. The system storage pool typically requires approximately 2% to 5% of the total storage capacity that SAN File System manages, but this amount varies depending on your environment. Use the alert features on the system storage pool to ensure that you do not run out of space.

Note: The minimum size of a system volume is 2 GB; therefore, the minimum size of the system storage pool is also 2 GB.

For security and reliability, the volumes assigned to the system storage pool should be accessible only to the server cluster by using a private SAN or a shared SAN with a combination of zoning, LUN masking, or special configuration. For reliability, the volumes should be virtualized RAID arrays (also known as ranks within IBM Enterprise Storage Server®).

User storage pools

User storage pools contain the blocks of data that make up user files. SAN File System stores the data that describes the files (which is called file metadata) separately from the actual file data.

An administrator can create one or more user storage pools, and then create policy sets that contain rules that cause Metadata servers to store data for specific files in the appropriate storage pools.

Default storage pool

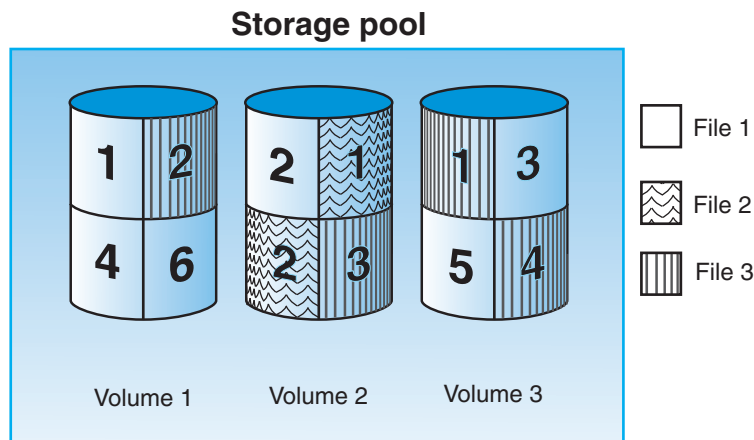
The default storage pool is the user storage pool where a Metadata server stores the data for a file if the file is not assigned to a specific storage pool by a rule in the active policy set.

A default storage pool is created when SAN File System is installed. However, an administrator must assign one or more volumes to it before it can be used. Note that while there can be only one default storage pool at a time, an administrator can designate any user storage pool that has volumes assigned to it to be the default storage pool.

Assigning volumes to storage pools:

Typically, an administrator assigns volumes to storage pools based on their common characteristics, such as device capabilities (availability or performance level) and usage (business division, project, application, location, or customer).

Each storage pool manages its own volumes. Space is allocated to the volumes in each storage pool in a round-robin algorithm (see the following figure), in logical partitions, or in blocks. Logical partitions are allocated to the system storage pool in 16-MB blocks. For user storage pools, including the default storage pool, an administrator can choose to allocate logical partitions in 16, 64, or 256-MB blocks. All logical partitions in the same storage pool must be the same size.



Note: An administrator can choose to have an alert generated whenever a storage pool reaches or exceeds a certain percentage of its maximum capacity. By default, an alert is generated when a storage pool becomes 80% full. An alert is logged every five minutes until one or more volumes are assigned to the storage pool. Note that an administrator can set configuration parameters to cause an SNMP trap message to be generated as well. An SNMP trap notifies an administrator of this condition asynchronously.

Related topics:

- “Alerts” on page 4
- “Creating a storage pool” on page 123
- “File placement” on page 43
- “Filesets” on page 24
- “Listing files on a volume” on page 131
- “Listing storage pools” on page 124
- “Listing volumes in a storage pool” on page 125

- “Policies and rules” on page 44
- “Setting the default storage pool” on page 125
- “Storage management” on page 43
- “Viewing storage pool details” on page 125
- “Volumes” on page 49

Remote access

Remote Access (RA) support is the ability for IBM support personnel who are not located on your premises to assist an administrator or local field personnel in diagnosing and repairing failures with SAN File System. Remote Access support can help to greatly reduce service costs and shorten repair times, which in turn will reduce the impact of any SAN File System failures on your business.

RA provides support personnel with full access to the SAN File System console. This allows the support personnel to query and control the SAN File System Metadata servers and possibly clients, and to access metadata, log, dump, and configuration data.

Remote access does not allow full access with no authentication. The customer must initiate a Virtual Private Network (VPN) connection to allow a support representative to remotely access the master console. From the master console, the support representative can establish a connection to the SAN File System Metadata servers. However, the customer can monitor that access and disconnect the session as desired.

Related topics:

- “Service Alert” on page 9
- “Master console” on page 32

Security

SAN File System provides *administrative security* to protect against unauthorized access to SAN File System administrative operations, and *client security* to protect against unauthorized access to files in the SAN File System global namespace. For more information, see the related topics below.

Related topics:

- “Administrative security”
- “Client security” on page 41

Administrative security

Administrative security protects against unauthorized access to SAN File System administrative operations.

Security for administrative operations includes several different levels of access. These access levels are provided through a combination of site preparation steps, support of industry-standard protocols, and configuration of the SAN File System administrative infrastructure.

The steps to prepare your site include the following:

- Setting up your IP network so that all systems that will be used to administer SAN File System by using a Web browser have Internet protocol access to the systems that host the SAN File System Administrative server and Metadata servers.
- Ensuring that an LDAP server is installed in your environment. An administrator must define users, their authenticating passwords, and SAN File System administration roles on an LDAP server. Then, the administrator must link users to the SAN File System administration roles.

The SAN File System administration roles are as follows:

- **Monitor**—Allows a user to view the state of the system only.
- **Backup**—Allows a user the same privileges of Monitor, plus the ability to perform backup and recovery operations.
- **Operator**—Allows a user the same privileges of Backup, plus the ability to control basic state changes, such as start and stop, for subsystems.
- **Administrator**—Provides a user with full, unrestricted control over administrative operations.

The SAN File System administrative infrastructure supports the Secure Sockets Layer (SSL) as part of its Web server software-platform base. Administrators log in to the SAN File System console, which is the Web-based user interface, through an SSL-secured login, and are authenticated using the LDAP server.

Administrators who want to use the administrative command line interface must first log in to the engine in the cluster on which the command line interface runs, and are authorized using the standard UNIX security mechanisms that are managed by the customer. For administrative command line interface users, an administrator has the option of requiring the use of SSH-based logins to a cluster before the user can use the administrative command line interface (tanktool). Then, to run the command line interface, all access is authorized using the LDAP server.

User authorization timeouts:

Although authorization is checked for every individual SAN File System function, the Administrative agent keeps a cache of LDAP information in order to reduce the number of queries to the LDAP server. When the Administrative agent gets an authentication entry for a particular user, it keeps that information cached for several minutes. Therefore, consecutive operations by a user within a few minutes only require one LDAP lookup.

However, there is an authorization timeout function that asks the Administrative agent to empty the cache, forcing the next function for each user to re-query the LDAP server. This function is useful if a password is changed, or a user is eliminated on the LDAP server, and you want the change to take effect immediately for SAN File System administrative operations. If you do not time out the authorizations, a change will not take effect until the old cache entries expire.

Related topics:

- “Administrative server” on page 3
- “Client security” on page 41
- “Metadata server” on page 33
- “Timing out all user authorizations” on page 128
- “User roles” on page 46

Client security

Client security protects against unauthorized access to files in the SAN File System global namespace. SAN File System clients mount the global namespace on their systems and, subject to security constraints, have complete, shared access to the namespace.

Authenticating and authorizing clients:

Clients are authenticated using external services such as Active Directory for Windows clients and Network Information Services (NIS) for AIX clients. SAN File System does not restrict how authentication is performed, but it does require that all Windows clients share a common definition of users and roles, and that all AIX clients share a common domain (definition of users and groups). SAN File System does perform authorization for file access.

It is also recommended, but not required, that all systems that host a Metadata server in the cluster be part of the shared domain. This makes it easier for administrators to perform client-based activities.

Specifying privileged clients:

SAN File System includes support for a configurable list of privileged clients. A privileged client is a client on which root users in AIX or users with administrator privileges in Windows are given those same privileges for the SAN File System global namespace. A root user logged in to a privileged AIX client is granted full control over directories, files, and other file system objects created by AIX clients, and a user with administrator privileges who is logged in to a privileged Windows client is granted full control over the folders, files, and other file system objects created by Windows clients.

If those same users log in to a client that is not a privileged client, their privileges for the global namespace are reduced to those of “Everyone” for Windows users or “Other” for AIX clients.

Supporting native file system security:

AIX clients: For AIX clients, SAN File System supports the POSIX definition of three sets of three file mode bits—one set for each user, group, and other. The bits in each group represent read, write, and execute/search permissions. Enforcement includes support for setuid and setgid bits, and the X/Open-specified restricted deletion mode (“sticky”) bit used for directories. SAN File System supports commands such as ls and du when they are run against the global namespace.

If a file created by an AIX client has the read and write bits set for user “Other,” all AIX and Windows users can read and write to the file.

Windows clients: For Windows clients, SAN File System supports access control lists (ACLs), which are lists that define permissions for users and groups. An entry in an ACL is called an access control entry (ACE). If a Windows file creates an ACE for user “Everyone,” all AIX and Windows users can access that file.

Sharing files:

When files are created and accessed from a Windows client, all the security features of Windows are available and enforced. When files are created and

accessed from AIX clients, all the security features of AIX are available and enforced. When files created by an AIX client are accessed by a Windows client, access is controlled using only the semantics and permissions of "Other." Similarly, when files created by a Windows client are accessed by an AIX client, access is controlled using only the semantics and permissions of "Everyone."

Note: After a directory, folder or file is created from a particular client type (AIX or Windows), its security settings cannot be changed to another client type.

Assigning ownership and permissions to filesets:

When an administrator creates and attaches a new fileset to the SAN File System global namespace, the fileset is owned by user Anonymous. An AIX root user or a Windows administrator user must change the ownership/permissions of the fileset to anything they choose before the fileset will be usable. (Note that this activity must also be performed for the FlashCopy directory under the fileset root.) This only needs to be done once in the lifetime of a fileset. The changed permissions are persistent across Metadata server restarts and the detaching and attaching of the fileset.

Unlike the requirement for the global fileset (which is the root fileset for a cluster), an AIX or Windows user can own a generic root fileset exclusively. The fileset is not required to have write permissions for both AIX and Windows domains.

Related topics:

- "Administrative security" on page 39
- "Filesets" on page 24
- "Global namespace" on page 29
- "Granting root privileges to a client" on page 76
- "Metadata server" on page 33
- "Clients" on page 11

SNMP

Simple Network Management Protocol (SNMP) is typically used to monitor the health and performance of software, hardware, and networks. SNMP consists of two main components:

- SNMP agents, which are software components that reside on managed devices and collect management information (using Management Information Bases or MIBs). SNMP agents issue traps when SNMP events occur. These traps are sent through User Datagram Protocol (UDP) to an SNMP Manager.
- An SNMP manager, which is a management application (client) that monitors and controls devices using SNMP protocol.

In SAN File System, the Metadata server generates SNMP traps in response to certain events. Note that no SNMP traps are issued from the operating system, hardware, or the Administrative agent.

Note: The RSA II cards can be set up to generate hardware traps as well.

SAN File System administrators can choose to configure which severity levels of events (informational, warning, error, or severe) generate SNMP traps and can

specify the SNMP Managers that receive the traps. When an event occurs with a severity level that causes an SNMP trap, SAN File System sends the trap and logs the event in the cluster log.

Note: SAN File System supports asynchronous monitoring through traps but does not support SNMP GETs or PUTs for active management. The SNMP Manager cannot manage SAN File System.

Not all events in SAN File System generate traps. Examples of events that might generate SNMP trap messages include the following:

- When a server executes a change in state
- When a server detects that another server is not active
- When the size of a fileset reaches a specified percentage of its capacity

Related topics:

- “Adding SNMP managers” on page 67
- “Alerts” on page 4
- “Service Alert” on page 9
- “Setting up SNMP traps” on page 70

Storage management

SAN File System provides automatic file placement through the use of *policies* and *storage pools*. An administrator can create quality of service storage pools that are available to all users, and define *rules* in policies that cause newly created files to be placed in the appropriate storage pools automatically. For more information about policies, rules, and storage pools, see the related topics below.

Related topics:

- “Backup and restore” on page 5
- “File placement”
- “FlashCopy images” on page 26
- “Policies and rules” on page 44
- “Storage pools” on page 37

File placement

SAN File System provides automatic *file placement* through the use of policies and storage pools.

A *policy* is a list of rules that determines where the data for specific files is stored.

A *rule* is an SQL-like statement that tells a SAN File System Metadata server to place the data for a file in a specific storage pool if the file attribute that the rule specifies meets a particular condition. A rule can apply to any file being created or to only files being created within a specific list of filesets and any filesets nested within the specified filesets.

A *storage pool* is a named set of storage volumes that can be specified as the destination for files in rules. If a storage pool is the destination for user files, it is called a *user storage pool* (in contrast to a *system storage pool*, which is used to store SAN File System system metadata).

The rules in a policy are processed in order until the condition in one of the rules is met. The data for the file is then stored in the specified storage pool. If none of the conditions specified in the rules of the policy is met, the data for the file is stored in the default storage pool.

Notes:

1. Rules in a policy are evaluated only when a file is being created. If an administrator switches from one policy to another, the rules in the new policy apply only to newly created files. Activating a new policy does not change the storage pool assignments for existing files. And moving a file does not cause a policy to be applied. Note that while an administrator can create multiple policies, only one policy can be active at a time.
2. After a file has been created, you can check its storage pool assignment. However, the only way to do so is to list files on volumes in the target storage pool from the Administrative command-line interface using the **reportvolfiles** command.
3. If you base your policies on userids, be aware of how the UNIX **untar** command restores files from backup. During the restore, a file is first created from backup with the userid of the performer of the backup and is then changed to the userid of the original creator of the file. With SAN File System, since the policy is applied to the file at creation, the policy applies to the userid of the performer of the backup rather than the userid of the original file creator.

Related topics:

- “Activating a policy” on page 118
- “Creating a policy” on page 119
- “Creating a storage pool” on page 123
- “Changing the rules in a policy” on page 118
- “Policies and rules”
- “reportvolfiles” on page 232
- “Storage pools” on page 37

Policies and rules

Policies and the rules that they contain are used to assign files to specific storage pools. A storage pool typically contains a set of volumes that provide a specific quality of service for a specific use, such as to store all files for a particular application or a specific business division.

Policies:

A *policy* contains a list of rules that determines where specific files are placed. An administrator can define any number of policies, but only one policy can be active at a time. If an administrator switches from one policy to another (defines a different policy to be the active policy), that action has no effect on files that are already assigned to specific storage pools. Rules in the newly activated policy are evaluated only for files created after the policy is activated.

A policy can contain any number of rules; however, the entire policy cannot exceed a length of 32-KB (which includes any spaces used to delimit the rules).

SAN File System performs error checking for policies in the following phases:

- When an administrator creates a new policy, the master Metadata server checks the basic syntax of all the rules in the policy.

- When an administrator activates the policy, the master Metadata server checks all references to filesets and storage pools. If a rule in the policy refers to a fileset that does not exist or is not attached or to a storage pool that does not exist, an error is returned, and the policy is not activated.
- When a new file is created, the rules in the policy are evaluated in order. If an error is detected, the Metadata server responsible for creating the file logs an error, skips all subsequent rules, and assigns the file to the default storage pool.

Note: When SAN File System is first installed, a null policy is created and remains active until an administrator creates and activates a new one. The null policy assigns all files to the default storage pool. Note that a default storage pool is created when SAN File System is first started; however, an administrator must assign volumes to it before it can be used. If users attempt to create new files that would be assigned to the default storage pool, and there are no volumes assigned to it, users will receive No Space errors.

Rules:

A *rule* is an SQL-like statement that tells a SAN File System Metadata server to place the data for a file in a specific storage pool if the file meets a particular condition. A rule can apply to any file being created or only to files being created within a specific list of filesets and any filesets nested within them.

SAN File System evaluates rules in the order that they appear in the active policy. When a client creates a file, SAN File System scans the list of rules in the active policy to determine which rule applies to the file. When a rule applies to the file, SAN File System stops processing the rules and assigns the file to the appropriate storage pool. If no rule applies, the file is assigned to the default storage pool.

Notes:

1. Rules in a policy are evaluated only when a file is being created. If an administrator switches from one policy to another, the rules in the new policy apply only to newly created files. Activating a new policy does not change the storage pool assignments for existing files. And moving a file does not cause a policy to be applied. Note that while an administrator can create multiple policies, only one policy can be active at a time.
2. After a file has been created, you can check its storage pool assignment. However, the only way to do so is to list files on volumes in the target storage pool from the Administrative command-line interface using the **reportvolfiles** command.
3. If you base your policies on userids, be aware of how the UNIX **untar** command restores files from backup. During the restore, a file is first created from backup with the userid of the performer of the backup and is then changed to the userid of the original creator of the file. With SAN File System, since the policy is applied to the file at creation, the policy applies to the userid of the performer of the backup rather than the userid of the original file creator.

Note:

For detailed information about creating policies and rules, see the related topics below.

Related topics:

- “Creating a policy” on page 119

- “File placement” on page 43
- “Filesets” on page 24
- “Changing the rules in a policy” on page 118
- “reportvolfiles” on page 232
- “Storage pools” on page 37
- “Viewing policy details” on page 121
- “Viewing policy rules” on page 121

Storage devices

SAN File System supports various types of storage devices in your storage area network (SAN). For example, you can use JBOD, RAID with mirroring, and hierarchically-managed storage devices. An administrator can also attach tape devices for backups and long-term storage, although tape devices cannot be part of a SAN File System storage pool.

All storage devices attached to the SAN File System SAN can be accessed by all clients (unless an administrator uses zoning to allow only specific clients to access specific devices). This enables data sharing among heterogeneous clients.

SAN File System supports the following storage devices:

- IBM Enterprise Storage Server[®] (ESS)
- IBM SAN Volume Controller (initially, this device will support ESS and FASTT)

Note: Although several storage devices are supported in SAN File System, for this release, only one type of storage device is supported at a time on the SAN File System SAN (that is, storage devices must be homogeneous).

Related topics:

- “Backup and restore” on page 5
- “Storage pools” on page 37
- “Clients” on page 11

User roles

SAN File System provides different levels of user access that you can assign to administrative tasks in your environment. These access levels, or *user roles*, are one way to provide added security. The SAN File System user roles are described in the following table:

Table 4. SAN File System user roles

Role	Level	Description
Monitor	Basic level of access	Allows the user to obtain basic status information about the cluster, display the cluster message log, display the rules in a policy set, and list information regarding SAN File System elements such as storage pools, volumes, and filesets.

Table 4. SAN File System user roles (continued)

Backup	Monitor + backup access	Allows the user to use all commands available to the Monitor role. In addition, it provides the user with the ability to perform backup and recovery tasks for metadata.
Operator	Backup + additional access	Provides use of all commands available to the Backup and Monitor roles. Also allows the user to perform day-to-day operations and tasks requiring frequent modifications.
Administrator	Full access	Provides access to all administrative commands except for the IBM Support-only commands for initial cluster setup and configuration tasks.

Note: Each less-restrictive role includes the privileges and abilities of the more-restrictive roles above it.

Related topics:

- “Administrative security” on page 39
- “Granting SAN File System user access and roles” on page 126
- “Listing user roles” on page 127
- “Listing users” on page 127

User interfaces

SAN File System provides the following user interfaces:

- A Web-based administrative user interface called the SAN File System console
- An administrative command line interface called tanktool
- A client command line interface

SAN File System console:

The SAN File System console allows an administrator to control and monitor SAN File System from a Web-based graphical user interface. For ease of monitoring, it provides a system overview that illustrates the status of the various SAN File System components. In addition, the SAN File System console provides in-line messaging that assists with system configuration, performance tuning and troubleshooting tasks.

The SAN File System console also contains the Help Assistant, which provides panel-level help information as well as links to related topics in the SAN File System Information Center. The Information Center serves as an online, searchable repository for all of the product documentation.

The SAN File System console includes the following main features:

Task bar

The task bar has both a tool segment and a task segment. The tool segment provides buttons for global user actions, such as closing the banner, accessing user assistance, and signing off from the system. The task segment allows you to toggle the navigation frame, titled **My Work**, on and off. The task bar also keeps track of all opened primary (or “main”)

tasks, and allows you to quickly jump back and forth between them. Primary tasks are those tasks listed in the **My Work** frame, and serve as starting points for related sub tasks.

My Work frame

Contains primary task-based links that open panels in the work area.

Work area

Displays all user input and product content through panels. The panels contain static information and user interface elements, such as text fields and tables, that you can manipulate to affect SAN File System settings and behavior.

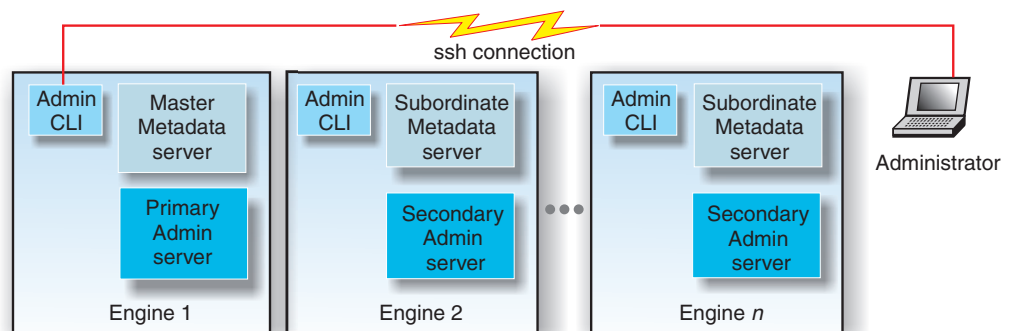
Tanktool:

Tanktool is an administrative command line interface (ACLI). An administrator can use this interface to administer all aspects of SAN File System, including setting up and managing storage pools, volumes, and filesets. For security reasons, tanktool runs only on the engines in your SAN File System server cluster. An administrator can use tanktool interactively and can embed tanktool commands in scripts.

To access tanktool, an administrator must log in to any engine that hosts a Metadata server.

Figure 4. Accessing tanktool

The following figure illustrates how the administrator accesses tanktool.



Client command line interface:

The client command line interface provides a set of commands used to set up a SAN File System client and to perform planning, migration, and verification tasks for data. This interface runs on all machines on which the SAN File System client code is installed.

To access client commands, a user logs in to the client machine.

Related topics:

- “Administrative commands” on page 142
- “Administrative server” on page 3
- Appendix B, “Commands”, on page 139
- “Client commands” on page 288
- “Using the SAN File System console” on page 333

Volumes

A *volume* is a logical unit number (LUN) labeled by SAN File System for its use. A LUN is the logical unit of storage that a SAN or other disk subsystem can assign to SAN File System Metadata servers and clients.

A LUN becomes a SAN File System volume when an administrator adds it to a storage pool. It is automatically assigned a system-generated label that identifies it as a SAN File System volume. An administrator must also give the volume a name that is unique within all the volumes used by a SAN File System cluster.

At every startup, a Metadata server scans all of the LUNs that it can access, searching for labels that indicate which LUNs are valid SAN File System volumes. Clients perform this same search whenever they are started.

Adding volumes to storage pools:

When you install SAN File System, there is a system storage pool, which is used by Metadata servers to store system and file metadata, and a default storage pool, which can be used to store file data. You can create additional user storage pools for file data. However, no data can be stored in a storage pool until an administrator assigns one or more volumes to it.

The volumes added to the system storage pool are called *system volumes*.

As the amount of metadata that is generated for the server cluster and client files grows, an administrator must ensure that the system storage pool always has enough volumes assigned to it so that it does not run out of space.

An administrator must also ensure that the default storage pool and any other user storage pools the administrator creates are assigned a sufficient number of volumes. Each storage pool must have at least one volume assigned to it before any files can be stored in it.

To assist an administrator in monitoring storage pool capacity, SAN File System provides a threshold option that an administrator can specify when adding a volume to a storage pool or changing settings for a storage pool. A threshold is a specified percentage of the estimated maximum capacity of the storage pool. When a storage pool reaches or exceeds the percentage specified as its threshold, SAN File System generates an alert. Note that this alert can also generate an SNMP trap message to notify an administrator of the condition asynchronously if the administrator sets the appropriate parameters for SNMP traps.

Activating volumes:

When an administrator adds a volume to a storage pool, the volume is activated by default. This means that a Metadata server can allocate data to the newly added volume. Note that an administrator can also choose to add a volume to a storage pool in a suspended state. However, no data can be allocated to the volume until the administrator activates it.

Suspending volumes:

A volume can be in a suspended state if an administrator chooses to add it to a storage pool without activating it. An administrator can also change the state of a

volume from activated to suspended. When a volume is in a suspended state, a Metadata server cannot allocate any data to it.

Removing volumes:

An administrator can also remove a volume from a storage pool. During this process, any files stored in the volume are automatically redistributed among the remaining volumes in the same storage pool. When an administrator removes a volume, data is moved and committed one logical partition at a time. If a failure occurs while moving the contents of a volume, the administrator can reissue the command, and the move process starts where it left off.

If I/O errors occur for some files or if the redistribution of the files to other volumes fails, an administrator can specify a force option that causes the volume to be removed anyway. If there is a bad file on the volume, specifying this option causes the system to delete the entire file, even if parts of the file reside on other volumes.

If the bad file is part of a FlashCopy image, SAN File System removes it from the FlashCopy image. This removal might render the FlashCopy image non-revertible. You can still backup and restore the remaining files on the FlashCopy image.

Note: There is no automatic recovery process when an administrator specifies the force option. All files are removed immediately without being copied. Before removing a volume with the force option, use the **reportvolfiles** command to display a list of files on the volume. The files for which failures occur are also listed in the cluster message log, and an administrator can restore those files manually.

When an administrator removes a volume from a storage pool, its label is removed, and the volume becomes a LUN again.

Related topics:

- “Adding volumes to a storage pool” on page 130
- “Alerts” on page 4
- “Listing files on a volume” on page 131
- “Listing volumes in a storage pool” on page 125
- “Storage pools” on page 37
- “Viewing volume details” on page 135
- “Viewing volume settings” on page 135

Chapter 2. Maintaining

Most hardware components in SAN File System are considered field-replaceable units; however there are a few components that you can replace.

Related topics:

- “Replacing master console components”
- “Replacing storage engine components”

Replacing master console components

Table 5. Master console parts listing

Part number	Units	Description
	1	Master console components
32P1032	1	• Display
28L3644	1	• Keyboard

Refer to the *IBM xSeries 345 Type 8670 Hardware Maintenance Manual and Troubleshooting Guide*, which was shipped with your master console, for replacement instructions.

Related topics:

- “Replacing storage engine components”

Replacing storage engine components

Table 6. Model 1RX parts listing

Part number	Units	Description
18P6148	2	SAN File System Model 1RX
49P2167	2	• Power supply, AC 514 watt, hot-swap
01R0587	8	• Fan assembly, 80-mm hot-swap

Related topics:

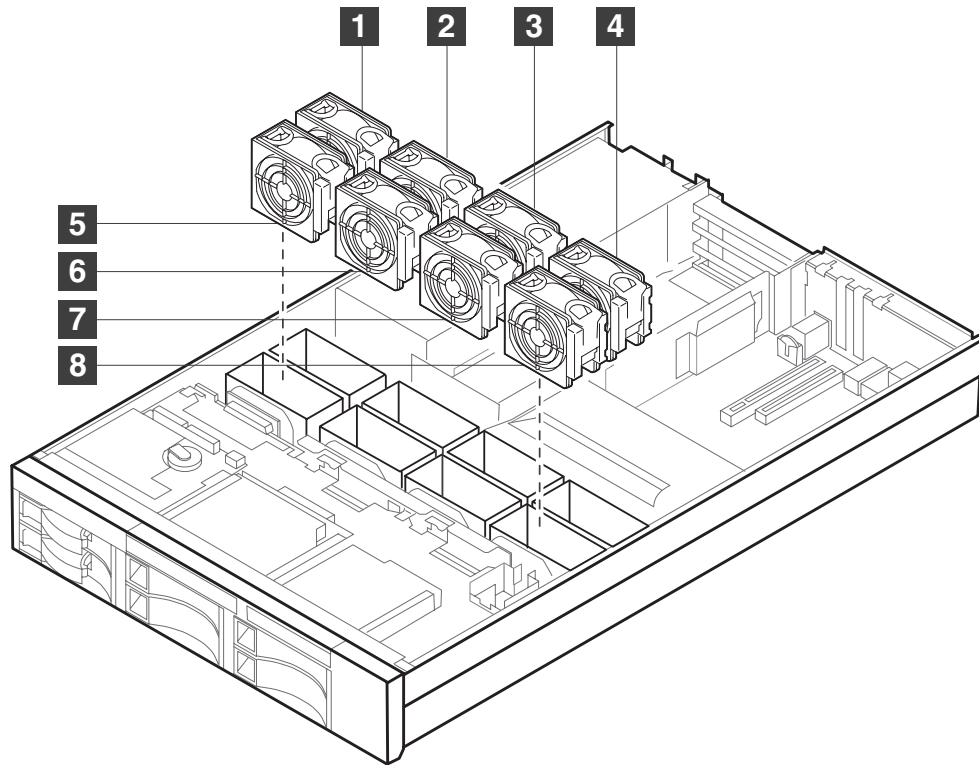
- “Replacing a hot-swap fan”
- “Replacing a hot-swap power supply” on page 52
- “Replacing master console components”

Replacing a hot-swap fan

The Model 1RX comes with eight hot-swap-fan assemblies making the fans fully redundant. You do not need to power off the Model 1RX to replace a hot-swap-fan assembly.

Attention: To ensure correct operation, if a fan fails, replace it.

The following illustration shows the location of the hot-swap fans.



- 1** Fan 1
- 2** Fan 2
- 3** Fan 3
- 4** Fan 4
- 5** Fan 5
- 6** Fan 6
- 7** Fan 7
- 8** Fan 8

Steps:

Perform the following steps to replace a hot-swap-fan assembly:

1. Read the safety information.
2. Remove the cover.
Attention: To ensure proper system cooling, do not remove the top cover for more than 30 minutes during this procedure.
3. Determine which fan assembly to replace by checking the LED on each fan. The LED on the failing fan assembly will be lit.
4. Place your fingers into the two handles on the top of the faulty fan assembly.
5. Squeeze the handles together and lift the fan out of the Model 1RX.
6. Orient the replacement fan to ensure that the LED on top of the fan is to the right of the Model 1RX (facing the front of the box).
7. Push the replacement fan assembly into the Model 1RX until it clicks into place.
8. Replace the cover.

Replacing a hot-swap power supply

The Model 1RX has two hot-swap power supplies for redundancy.

Attention: During normal operation, each power-supply bay must have either a power supply or power-supply blank installed for proper cooling.



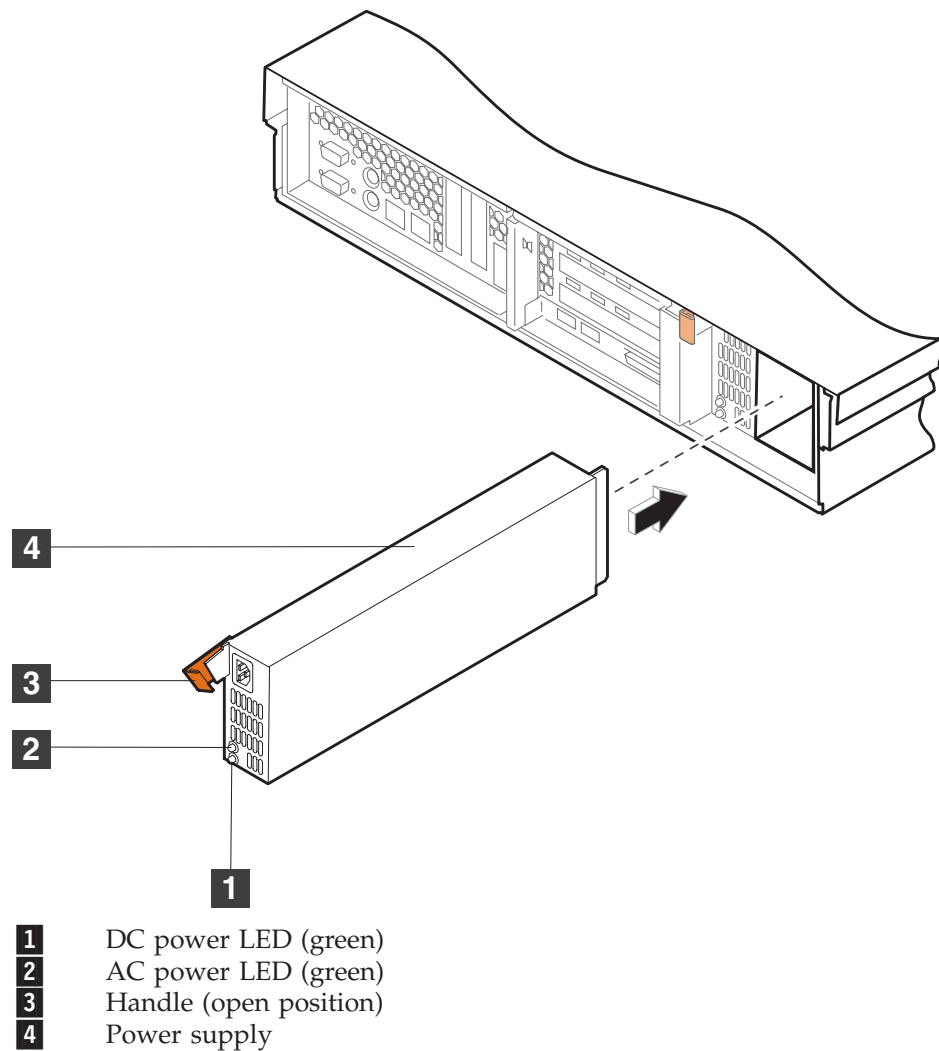
DANGER

Do not try to open the covers of the power supply module. (D02)



DANGER

Do not plug the power cable into the power supply module until the enclosure is completely installed, its retaining screws are tightened, and all signal cables are connected. (D03)



Notes:

1. The Model 1RX provides power redundancy and the ability to hot-swap certain components; therefore, you do not need to power off the engine to install hot-swap power supplies. However, if the load on your system requires the capacity of all installed power supplies, you do not have redundancy or the ability to hot-swap power supplies, and must power off the system before removing any of the power supplies.
2. When replacing a power supply, do not remove the power supply until you are ready to install its replacement.

Attention: If you remove a power supply while the engine is running and you do not have power redundancy, your appliance will abruptly cease to function.

Steps:

Perform the following steps to replace a hot-swap power supply:

1. Read the safety information.
2. Remove the cover.
3. Determine which power supply has failed by checking the Power Supply LEDs on the Light Path Diagnostics panel on the system planar.
 - If the Non Redundant LED is on, you do not have redundancy. Power OFF the engine and peripheral devices.
 - If the Non Redundant LED is not on, you have redundancy and do not need to power off the engine.
4. Remove the faulty power supply:
 - a. Disconnect the power cable from the back of the failing power supply. Do not disconnect the power cables from any other power supplies.
 - b. Press firmly on the tab behind the handle on the failing power supply and then pull out on the handle to release the power supply.
 - c. Gently slide out the power supply.
5. Install the replacement power supply in the bay:
 - a. Place the handle on the power supply in the open position (that is, perpendicular to the power supply) by pinching the side clip and pressing down on the handle.
 - b. Slide the power supply into the chassis and gently push the power supply into the bay until it connects to the power supply backplane connector.
 - c. Press on the handle to seat the power supply in the bay.
6. Plug the power cord for the new power supply into the power-cord connector on the rear of the new power supply.

The following illustration shows the rear of the server.



7. If you powered off the engine and any peripheral devices, power ON these devices.
8. Verify that the dc power LED and AC power LED on the power supply are lit, indicating that the power supply is operating correctly.

Related topics:

- “Powering off the engine” on page 95
- “Powering on the engine” on page 96

Chapter 3. Monitoring

Related topics:

The Administrative CLI allows you to monitor the SAN File System by displaying statistics, status, and logs. Additionally, you can use the one-button data collection function to collect diagnostic data on engines and clients.

- “Clearing logs” on page 68
- “Creating a report”
- “One-button data collection utility”
- “Viewing logs” on page 71
- “Viewing statistics” on page 61
- “Viewing system status” on page 62

Creating a report

Steps:

Perform the following steps to create a report of component statistics:

1. Click **Monitor System** → **Statistics** from the My Work frame.
2. Click the **Create Report** tab.
3. Select the components for which you want to include statistics in the report.
4. Click **Create Report**.

- Note:** You can print this report using the print function in your Web browser.
5. Click **Close** to close the report.

Related topics:

- “Statistics - Create Report” on page 415
- “Statistics - Report” on page 418

One-button data collection utility

You can use the one-button data collection utility to gather vital product data (VPD) about SAN File System hardware and software. This information can help you analyze a problem as well as collect the data to send to other support personnel.

You can invoke the one-button data collection utility in one of the following ways:

- For an engine in the cluster, perform one of these actions:
 - Start the SAN File System console. Then click **Maintain System** → **Collect Diagnostic Data**.
 - Access the engine and from a shell prompt, run `/usr/tank/server/bin/pmf.sh` to collect the default data or add additional parameters to customize the data collection.
- For a client running AIX, access the client and from a shell prompt, run `/usr/tank/client/bin/pmf.sh` to collect the default data or add additional parameters to customize the data collection.

- For a client running Windows, access the client and from a shell prompt, run `C:\Program Files\IBM\Storage Tank\client\bin\pmf.bat` to collect the default data. In addition, you must perform some additional steps to collect all of the data.

Related topics:

- “collectdiag” on page 166
- “Software Vital Product Data” on page 59
- “Hardware Vital Product Data”

Hardware Vital Product Data

This section describes the hardware vital product data that is collected by the one-button data collection utility.

Engine:

You can collect hardware information about an engine in the cluster, such as the following:

Table 7. Engine hardware VPD

Hardware area	Component	VPD collected
Processor/PCI devices	Machine	Type, model number, vendor, and serial number.
	Host bridge	Device, vendor, firmware version, and latency.
	ISA bridge	Device, vendor, and firmware version.
	Ethernet controller	For each device: adapter type, vendor, firmware version, latency, and memory usage.
	USB controller	
	VGA controller	
	IDE interface	
	Fibre-channel adapter	
	RSA II adapter	
Memory	Memory	Statistics for total memory available, used, free, shared, buffered, and cached. Additional usage statistics as well.
	Swap space	
LAN network	Ethernet interfaces	For each device: data received and transmitted, Internet address, network masks, packets, collisions, interrupts, errors, and base memory address.
	Loopback interface	Statistics about Internet address, network masks, packets, collisions, and errors.
	IP routing table	For each destination: gateway address, network masks, flags, and interfaces.
Local storage	SCSI devices	For each device: channel, ID, LUN, vendor, model, version, and type.
	File systems	Device, mount point type, and inodes (total, used, and free).
	Mount points	Device, file system, and read/write settings.

Related topics:

- “One-button data collection utility” on page 57

Software Vital Product Data

This section describes the software vital product data that is collected by the one-button data collection utility.

Engine:

You can collect information about the software running on an engine in the cluster, such as the following:

Table 8. Engine software VPD

Software area	Component	VPD collected
Operating system	Machine	Name and version.
	Operating system	Version, build information, and installation date.
	Processes	Owner, ID, binary file, status, runtime parameters, and environment variables. Additional details as well.
	System log files	Collect in their entirety.
	Core files	Operating system core files and corresponding binary file, if present and requested.
Network	Active connections	Protocol (TCP/UDP), local and remote addresses, state, and receive and send queues.
	Active sockets	Type, state, flags, reference count, and full pathname.
	ARP	IP address, hardware address, and device.

Table 8. Engine software VPD (continued)

Software area	Component	VPD collected
Metadata server	Configuration files	Version, Tank.Bootstrap, and Tank.Config files.
	Log files	log.std, log.audit, log.trace, log.cim, log.failover, and log.*.old.
	Core files	Server core file, if present.
	Server configuration	Dump information about: <ul style="list-style-type: none"> • Version of installed server code • Current server state • Current server role • Protocol (TCP or UDP) • IP address and network mask
	Server state	Dump information about: <ul style="list-style-type: none"> • Active threads, their state, and activity • Mutexes and current state of each one • Latches and current state of each one • Condition variables and information about each one • Write-ahead log writer thread information.
Administrative server	Cluster configuration	Dump information about: <ul style="list-style-type: none"> • Number of servers in cluster • Listing of all servers in cluster • Fileset (container) information • Global disk table and type of each disk (master, system, user) • List of registered clients and information about each one • Heartbeat interval between servers • Current state of High-Availability Manager
	Configuration files	tank.properties, cimom.properties, and tank_device_map files.
	Log files	cimom.log, console.log, security.log, and WebSphere Application Server-based trace.log files.
	Core files	Server core file, if present.

Client:

You can collect information about the software running on a client, such as the following:

Table 9. Client software VPD

Software area	Component	VPD collected
Operating system	Machine	Name and version.
	Operating system	Version, build information, and installation date.
	Processes	Owner, ID, binary file, status, runtime parameters, and environment variables. Additional details as well.
	System log files	Collect in their entirety.
	Core files	Operating system core files and corresponding binary file, if present and requested.
SAN File System client	Log files and trace files	All client log and trace files
Network	Active connections	Protocol (TCP/UDP), local and remote addresses, state, and receive and send queues.
	Active sockets	Type, state, flags, reference count, and full pathname.
	ARP	IP address, hardware address, and device.

Related topics:

- “One-button data collection utility” on page 57

Viewing statistics

Steps:

Perform the following steps to view statistics for specific SAN File System components:

1. Click **Monitor System** → **Statistics** from the My Work frame.
2. Click the tab for the component for which you want to view statistics.
3. Click **OK** to view the statistics.
4. Click **Close** to close the Statistics panel.

Related topics:

- “Creating a report” on page 57
- “statcluster” on page 267
- “Statistics - Client Sessions” on page 413
- “Statistics - Cluster” on page 414
- “Statistics - Engines” on page 416
- “Statistics - Filesets” on page 417
- “Statistics - LUNs” on page 417
- “Statistics - Servers” on page 419
- “Statistics - Storage Pools” on page 419
- “Statistics - Volumes” on page 421

Viewing system status

Steps:

To view status for the major SAN File System components (including cluster state, Metadata server activity and availability, and recent messages), click **Monitor System** → **System Overview** from the My Work frame.

Note: The SAN File System console provides limited filtering for recent messages. You can view either all messages or only the alerts that have been generated. To change the type of messages that is displayed, select the filter from the **Message Type** drop-down list, and then click **Submit**.

You can also change the interval at which the information on this page is refreshed. To change the refresh rate or to disable refreshing, select the appropriate interval from the **Refresh Interval** drop-down list, and then click **Submit**.

Related topics:

- “Alerts” on page 4
- “Cluster” on page 16
- “Filesets” on page 24
- “System Overview” on page 428

Chapter 4. Migrating data

Prerequisites:

Review the data-migration prerequisites before you begin migrating data.

Context:

Data is migrated using the **migratedata** command from the client machine.

Attention: When you migrate journaled file system (JFS) files to SAN File System, you will lose access control lists (ACLs) from those files.

Steps:

Perform the following steps to migrate your data in SAN File System:

1. Estimate the time that it will take to migrate the data.
2. Import (or migrate) the data to the SAN File System.
3. Verify the integrity of the migrated data.

Related topics:

- “Data migration” on page 22
- “Data-migration prerequisites” on page 437
- “Estimating the time to migrate”
- “Importing data into the SAN File System” on page 64
- “migratedata” on page 289

Estimating the time to migrate

For large migrations, it is recommended that you estimate the amount of time that it will take to migrate the data set before you begin. The data-migration utility estimates this time based on several factors:

- Data-transfer rate over the storage area network (SAN)
- Amount of data being migrated
- Amount of available disk space on the target file system
- Amount of available memory
- Number of CPUs

To determine the data-transfer rate, the data-migration utility copies a set of the actual files from the source to the target file system.

Note: The estimation process can take a while if the data set is comprised of a large number of small files.

Prerequisites:

Review the data-migration prerequisites before you use the data-migration utility.

Steps:

Perform these steps to estimate the time needed to migrate data:

1. On the client machine, change to the directory where the **migratedata** command is located. For AIX, this is the /usr/tank/migration/bin directory. For Windows, this is the c:\Program Files\IBM\Storage Tank\Migration directory.
2. Invoke the **migratedata -phase plan** command.

Related topics:

- “Data migration” on page 22
- “Data-migration prerequisites” on page 437
- Chapter 4, “Migrating data”, on page 63
- “migratedata” on page 289

Importing data into the SAN File System

Context:

Attention: When you migrate journaled file system (JFS) files to SAN File System, you will lose access control lists (ACLs) from those files.

You can migrate legacy data from your existing file system to the SAN File System using the data-migration utility on the client machine. This utility copies each file-system object from the source file system to the target SAN File System file system. The integrity of the migrated data and metadata (such as permissions and creation time) is checked automatically during the migration process.

The data-migration utility makes an entry in the log file before each file is migrated and marks that entry as “done” when the migration of that file is complete. When migrating large files, you can use the **-checkpoint** option to mark the entry in the log file after a specified number of blocks is migrated. The size of the block depends on the client platform.

Note: You can stop the data-migration process at any time and resume after the last completed file or block. The data-migration utility uses the log file to determine where the process was stopped; it knows where to resume the process.

Prerequisites:

Review the data-migration prerequisites before you begin migrating data.

Steps:

Perform the following steps to migrate your data:

1. On the client machine, change to the directory where the **migratedata** command is located. For AIX, this is the /usr/tank/migration/bin directory. For Windows, this is the c:\Program Files\IBM\Storage Tank\Migration directory.
2. Invoke the **migratedata -phase migrate** command.

Related topics:

- “Data migration” on page 22
- “Data-migration prerequisites” on page 437
- Chapter 4, “Migrating data”, on page 63

- “Stopping a data migration”
- “Resuming a data migration”
- “Backing out migrated data” on page 66
- “migratedata” on page 289

Stopping a data migration

You can stop the data-migration process at any time and resume from the last completed file or block.

Steps:

To stop the data-migration process, press **Ctrl+c**. You can then inspect the progress of the migration by viewing the log file.

Note: To clean up SAN File System after you stop the data migration, remove the migrated files using the standard operating system utilities, such as **rm** on UNIX or **del** on Windows, before you attempt to migrate data again.

Related topics:

- “Backing out migrated data” on page 66
- “migratedata” on page 289
- “Resuming a data migration”

Resuming a data migration

If you stop the data-migration process or if the process is terminated for some other reason, you can resume the migration after the last completed file or block without requiring a complete restart. If you migrated the same data set without specifying the **-restart** option again, the data-migration process starts from the beginning, and the data on the target file system will be overwritten.

Steps:

To resume a data migration, invoke the **migratedata -phase migrate -resume** command on the SAN File System client machine.

Attention: You must specify the same log file as that used by the data-migration process being resumed. If you specify a different log file and do not specify the **-f** option, you will receive an error and the migration will stop. If you specify a different log file and specify the **-f** option, you will receive a warning and the data on the target file system will be overwritten.

Related topics:

- “Data migration” on page 22
- “Data-migration prerequisites” on page 437
- Chapter 4, “Migrating data”, on page 63
- “Stopping a data migration”
- “migratedata” on page 289

Backing out migrated data

You can back out a set of migrated data after the migration process is complete by pointing to the source file system rather than the SAN File System file system. The data-migration utility does not modify or delete data in the source file system.

Related topics:

- “Data migration” on page 22
- Chapter 4, “Migrating data”, on page 63

Chapter 5. Managing

This section discusses tasks that will help you manage SAN File System on a daily basis.

Related topics:

- “Managing alerts and logs”
- “Managing backups” on page 72
- “Managing clients” on page 75
- “Managing the cluster” on page 82
- “Managing disaster recovery” on page 86
- “Managing engines” on page 93
- “Managing filesets (containers)” on page 102
- “Managing FlashCopy images” on page 107
- “Managing Metadata servers” on page 109
- “Managing policies” on page 117
- “Managing processes” on page 121
- “Managing storage pools” on page 122
- “Managing users” on page 126
- “Managing volumes and LUNs” on page 128

Managing alerts and logs

Through the setting of alerts, and SNMP traps, as well as the viewing of logs, you can monitor SAN File System components such as storage pool size, Metadata server performance, and so forth. The following topics will assist you with configuring SAN File System alerts and SNMP managers and in viewing and clearing logs.

Related topics:

- “Adding SNMP managers”
- “Clearing logs” on page 68
- “Deleting SNMP managers” on page 69
- “Modifying SNMP managers” on page 69
- “Modifying SNMP traps” on page 70
- “Setting up SNMP traps” on page 70
- “Viewing SNMP managers” on page 71
- “Viewing logs” on page 71
- “Viewing system status” on page 62

Adding SNMP managers

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

The SAN File System console allows you to specify up to two SNMP managers.

Note: If Service Alert is configured, then one of the managers will be set to the Service Alert SNMP client.

Steps:

Perform the following steps to add an SNMP manager for routing SAN File System SNMP traps:

1. Click **Monitor System** → **SNMP Properties** from the My Work frame.
2. Click the **SNMP Managers** tab, if it is not already selected.
3. Type a **Destination IP** for one or both managers.
4. Type a **Destination Port** for one or both managers.
5. Select the appropriate **SNMP version** for one or both managers.
6. Type an **SNMP community** for one or both managers.
7. Click **OK** to confirm the addition of the manager or managers specified.

Related topics:

- “addsnmpmgr” on page 149
- “Alerts” on page 4
- “Managing alerts and logs” on page 67
- “SNMP” on page 42
- “SNMP Properties - SNMP Managers” on page 412

Clearing logs

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

You can clear the entire audit log or cluster log.

Notes:

1. Clearing the cluster log also clears the event log because the event log is a subset of the cluster log.
2. You cannot clear the administrative log or security log through the SAN File System console or the Administrative command-line interface.

Steps:

Perform the following steps to clear the audit log or cluster log:

1. Click **Monitor System** and then click **Audit Log** or **Cluster Log** from the My Work frame.
2. Click the **Clear Log** button above the log content.
3. Click **Clear Log** to clear the log.

Related topics:

- “Audit Log” on page 349
- “clearlog” on page 165

- “Cluster Log” on page 358
- “Logs” on page 30
- “Viewing logs” on page 71

Deleting SNMP managers

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

The SAN File System console allows you to specify up to two SNMP managers. The SNMP manager is really an SNMP server, which then forwards traps to the appropriate location (such as an SNMP client if one is present in your environment).

Steps:

Perform the following steps to delete an SNMP manager:

Attention: If you do not have any SNMP managers specified, SNMP traps will not be forwarded.

1. Click **Monitor System** → **SNMP Properties** from the My Work frame.
2. Click the **SNMP Managers** tab, if not already selected.
3. Delete the text in the **Destination IP** field or fields.
4. You can also delete information in the other fields for the associated manager, but it is not required.
5. Click **OK** to confirm the deletion of the manager.

Related topics:

- “Alerts” on page 4
- “Managing alerts and logs” on page 67
- “rmsnmpmgr” on page 246
- “SNMP” on page 42
- “SNMP Properties - SNMP Managers” on page 412

Modifying SNMP managers

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

The SAN File System console allows you to specify up to two SNMP managers. The SNMP manager is really an SNMP server, which then forwards traps to the appropriate location (such as an SNMP client if one is present in your environment).

Steps:

Perform the following steps to modify an SNMP manager:

1. Click **Monitor System** → **SNMP Properties** from the My Work frame.
2. Click the **SNMP Managers** tab.
3. Modify the information in the fields for one or more managers as required.
4. Click **OK** to confirm the changes.

Related topics:

- “addsnmpmgr” on page 149
- “Alerts” on page 4
- “Managing alerts and logs” on page 67
- “rmsnmpmgr” on page 246
- “SNMP” on page 42
- “SNMP Properties - SNMP Managers” on page 412

Modifying SNMP traps

Prerequisites: You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to modify SNMP traps:

1. Click **Monitor System** → **SNMP Properties** from the My Work frame.
2. Click the **SNMP Events** tab.
3. Change the selected event severity levels for which you would like SNMP traps to be sent.
4. Click **OK** to confirm the changes.

Related topics:

- “Alerts” on page 4
- “Managing alerts and logs” on page 67
- “Istrapsetting” on page 212
- “settrap” on page 257
- “SNMP” on page 42
- “SNMP Properties - SNMP Managers” on page 412

Setting up SNMP traps

Prerequisites: You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to set up SNMP traps:

1. Click **Monitor System** → **SNMP Properties** from the My Work frame.
2. Click the **SNMP Events** tab.
3. Select the event severity levels for which you would like SNMP traps to be sent.
4. Click **OK** to confirm the selected levels.

Related topics:

- “Alerts” on page 4
- “Managing alerts and logs” on page 67
- “Istrapsetting” on page 212
- “settrap” on page 257
- “SNMP” on page 42
- “SNMP Properties - SNMP Managers” on page 412

Viewing logs

Context:

SAN File System console allows you view these logs to help you monitor the system and troubleshoot problems:

- Administrative log
- Audit log
- Cluster log
- Event log
- Security log

Note: The cluster log is a consolidated view of the server log on each engine in the cluster. The event log is a subset of the messages stored in the cluster log; it contains only those messages that have an message type of event.

Steps:

To to view the Administrative log, Audit log, Cluster log, or Security log, click **Monitor System** from the My Work frame, and then click on the name of the log that you want to view.

To view the event log, perform the following steps:

1. Click **Monitor System** from the My Work frame.
2. Click **Cluster log**.
3. Select **Events only** from the **Messages Type** drop-down list.
4. Click **Refresh**.

Related topics:

- “Administrative Log” on page 347
- “Audit Log” on page 349
- “catlog” on page 154
- “Cluster Log” on page 358
- “Logs” on page 30
- “Security Log” on page 406

Viewing SNMP managers

Context: The SAN File System console allows you to specify up to two SNMP managers. The SNMP manager is really an SNMP server, which then forwards traps to the appropriate location (such as an SNMP client if one is present in your environment).

Steps:

Perform the following steps to view the SNMP managers:

1. Click **Monitor System** → **SNMP Properties** from the My Work frame.
2. Click the **SNMP Managers** tab.
3. The panel displays the information about currently configured SNMP managers.
4. Click **Cancel** to close the window, or the close button to close the window.

Related topics:

- “Alerts” on page 4
- “Managing alerts and logs” on page 67
- “Issnmpmgr” on page 211
- “SNMP” on page 42
- “SNMP Properties - SNMP Managers” on page 412

Managing backups

SAN File System supports the use of backup tools that are already present in your environment. For example, if your enterprise currently uses a storage management product such as Tivoli® Storage Manager (TSM), SAN File System clients can use the functions and features of that product to back up and restore files that reside in the SAN File System global namespace.

For backing up in a normal, available environment, you can use the FlashCopy image feature of SAN File System.

To prepare for disaster recovery in situations where SAN File System becomes unavailable, you can perform LUN-based backups using the instant copy features that exist in the storage subsystems that SAN File System supports. If your SAN storage subsystems do not offer copy services, you must back up for disaster recovery using the API method.

Related topics:

- “Backup and restore” on page 5
- “Backing up using the LUN method”
- “Backing up using the API method” on page 73
- “Creating a FlashCopy image” on page 107
- “Managing disaster recovery” on page 86

Backing up using the LUN method

Prerequisites:

The LUN method of backup is only available to SANs comprised of storage subsystems with built-in copy services. SANs without such service must use the API method of backup.

Context:

Because the LUN method deals with data at the byte level, it is an all-or-nothing approach for backing up and restoring your entire SAN File System. In particular, it provides no ability to restore individual files (because it has no concept of files); you have to save and restore all the data — metadata and file data — or none of it.

Restoring a previously saved FlashCopy image is the best method for recovering some subset of SAN File System data. Therefore, the LUN method is best employed as part of a disaster recovery situation.

Steps:

To back up the system using the LUN method, both the metadata and user LUNs must be in a static, consistent state. Perform the following steps to do a SAN File System backup using the LUN method:

1. Stop or pause all SAN File System client applications. Because this task is application-specific, refer to the application documentation for details on performing this step.
2. The Metadata server and all clients must complete all active transactions and flush their data to disk.

Quiesce the SAN File System Metadata servers using the **quiescecluster -state full** command. This procedure will also lock out any subsequent new I/O from the clients or Metadata server.

3. Initiate the storage subsystem copy service using the procedure defined in its accompanying documentation.
4. After the storage subsystem copy is complete, re-enable the SAN File System Metadata servers using the **resumecluster** command.
5. Restart the client applications using the specific procedures for those applications.

A safeguard is to create backup copies of the Metadata server cluster configuration. These files: Tank.Bootstrap and Tank.Config, already have one automatic backup copy in the boot drive mirrors. This information may be regenerated from the metadata LUNs themselves, but with some difficulty. For information about creating a recovery file, refer to the *IBM TotalStorage SAN File System Administrator's Guide and Reference* on the publications CD that came with your Metadata servers.

For additional information about restore procedures, including commands, refer to the *IBM TotalStorage SAN File System Maintenance and Problem Determination Guide*, on the publications CD that came with your Metadata servers.

Related topics:

- "Backing up using the API method"
- "Backup and restore" on page 5
- "Clients" on page 11
- "Managing disaster recovery" on page 86
- "Engines" on page 23
- "FlashCopy images" on page 26
- "quiescecluster" on page 229
- "resumecluster" on page 236

Backing up using the API method

Prerequisites:

The API method of backup is used for SANs comprised of storage subsystems that do not offer built-in copy services. SANs that do offer copy services can use the LUN method of backup.

Steps:

You have two possible options when using the API method of backup. Which method you choose depends on the characteristics of the backup application in your existing environment.

If your existing backup application allows you to selectively choose subdirectory branches for backup, and allows you to restore files to the grandparent directory two levels above their original location, then follow this optimized procedure for SAN File System API backup:

During your regularly scheduled backup procedure, perform the following steps:

1. Stop or pause all SAN File System client applications. Because this task is application-specific, refer to the application documentation for details on performing this step.
2. Create FlashCopy images of each fileset with the `mkimage -<fileset name> -<directory name> <FlashCopy image name>` command.

This command must be executed for each fileset. Use the same <directory name> and <FlashCopy image name> for each fileset.

Note: The term “container” is being phased out of SAN File System in favor of the term “fileset.” “Container” still appears in command names, messages, and other places, although the publications use the newer term “fileset” wherever possible. The term “container” means the same as the term “fileset.”

3. Save the most recent metadata to accompany the FlashCopy images with the `mkdrfile <most recent metadata file name>` command on the master Metadata server engine. Copy that file onto the client machine from which backup applications will run. Now you have a valid backup of everything you need.
4. Restart the client applications using the specific procedures for those applications.
5. Use the backup application to back up all <fileset name>/<directory name> subdirectories and their contents, along with the <most recent metadata file name> file to your backup medium (usually tape).

Attention: Backup Windows filesets only from a Windows client; backup AIX filesets only from an AIX client.

If your existing backup application does not provide the features required for the enhanced method, then follow this procedure for SAN File System API backup:

During your regularly scheduled backup procedure, perform the following steps:

1. Stop or pause all SAN File System client applications. Because this task is application-specific, refer to the application documentation for details on performing this step.
2. Save the most recent metadata to accompany the FlashCopy images with the `mkdrfile <most recent metadata file name>` command on the master Metadata server engine. Copy that file onto the client machine from which backup applications will run.
3. Use the backup application to backup all <fileset name>/<directory name> subdirectories and their contents, to your backup medium (usually tape).

If possible, exclude all .flashcopy subdirectories and their contents since they will not be of any use during a subsequent restore operation.

Note: The term “container” is being phased out of SAN File System in favor of the term “fileset.” “Container” still appears in command names, messages, and other places, although the publications use the newer term “fileset” wherever possible. The term “container” means the same as the term “fileset.”

4. Restart the client applications using the specific procedures for those applications.

For additional information about restore procedures, including commands, refer to the *IBM TotalStorage SAN File System Maintenance and Problem Determination Guide*, provided on the publications CD that came with your Metadata servers.

Related topics:

- “Backing up using the LUN method” on page 72
- “Clients” on page 11
- “Managing disaster recovery” on page 86
- “Engines” on page 23
- “FlashCopy images” on page 26
- “mkdrfile” on page 220
- “mkimage” on page 221

Managing clients

A SAN File System client is a computer that accesses and creates data that is stored in the SAN File System global namespace. Some client-management tasks are performed from the SAN File System through the SAN File System console and Administrative command-line interface, and some tasks are performed from the client shell.

Related topics:

- “Displaying the client-driver version”
- “Granting root privileges to a client” on page 76
- “Listing client sessions” on page 76
- “Listing clients with root privileges” on page 77
- “Rediscovering volumes accessible to a client” on page 77
- “Revoking root privileges from a client” on page 77
- “Starting a client” on page 78
- “Stopping a client” on page 80
- “Viewing client-session details” on page 81
- “Viewing client-session statistics” on page 82

Displaying the client-driver version

Steps:

For AIX-based clients, to display the the version of a client driver, run the `stfsstatus` command from the client shell.

Related topics:

- “Clients” on page 11
- “stfsstatus” on page 304

Granting root privileges to a client

Prerequisites:

You must Operator or Administrator privileges to perform this task from the SAN File System console. You must have Administrator privileges to perform this task from the Administrative command-line interface.

Steps:

Perform the following steps to grant root privileges to a client:

1. Click **Manage Servers and Clients** → **Client Sessions** from the My Work frame.
2. Select one or more client sessions to which you want to grant privileges.
3. Click **Grant Clients Root Privileges** from the drop-down box in the table header.
4. Click **Go**.
5. Click **Grant Root Privileges** to grant privileges to the specified clients.

Note: From the SAN File System console, you can grant root privileges only to clients that are connected to SAN File System. From the Administrative command-line interface, you can grant root privileges to any client, even if it is not connected to SAN File System.

Related topics:

- “Clients” on page 11
- “Client Sessions” on page 355
- “chclusterconfig” on page 157
- “Listing client sessions”
- “Listing clients with root privileges” on page 77
- “Revoking root privileges from a client” on page 77
- “Viewing client-session details” on page 81

Listing client sessions

Steps:

To display a list of the current client sessions from the SAN File System console, click **Manage Servers and Clients** → **Client Sessions** from the My Work frame.

Related topics:

- “Clients” on page 11
- “Client Sessions” on page 355
- “Granting root privileges to a client”
- “Listing clients with root privileges” on page 77
- “lsclient” on page 179

- “Revoking root privileges from a client”
- “Viewing client-session details” on page 81

Listing clients with root privileges

Steps:

To display a list of clients with root privileges, click **Manage Servers and Clients** → **Client Privileges** from the My Work frame.

Related topics:

- “Clients” on page 11
- “Client Sessions” on page 355
- “Granting root privileges to a client” on page 76
- “Listing client sessions” on page 76
- “lsclient” on page 179
- “Revoking root privileges from a client”
- “Viewing client-session details” on page 81

Rediscovering volumes accessible to a client

Prerequisites:

You must have root privileges to use this task.

Steps:

For AIX-based clients, to scan the SAN File System for new and removed volumes and determine which volumes can be accessed by the local client, invoke the **stfsdisk** command from the client machine.

Related topics:

- “Clients” on page 11
- “stfsdisk” on page 297

Revoking root privileges from a client

Prerequisites:

You must Operator or Administrator privileges to perform this task from the SAN File System console. You must Administrator privileges to perform this task from the Administrative command-line interface.

Steps:

Perform the following steps to revoke root privileges from a client:

1. Click **Manage Servers and Clients** → **Client Sessions** from the My Work frame.
2. Select one or more client sessions from which you want to revoke root privileges.
3. Click **Revoke Clients Root Privileges** from the drop-down box in the table header.

4. Click **Go**.
5. Click **Revoke Root Privileges** to confirm the operation.

Note: From the SAN File System console, you can revoke root privileges only from clients that are connected to SAN File System. From the Administrative command-line interface, you can revoke root privileges from any client, even if it is not connected to SAN File System.

Related topics:

- “Clients” on page 11
- “Client Sessions” on page 355
- “chclusterconfig” on page 157
- “Granting root privileges to a client” on page 76
- “Listing client sessions” on page 76
- “Listing clients with root privileges” on page 77
- “Viewing client-session details” on page 81

Starting a client

The steps to start the client are specific to the operating system running on the client machine.

Related topics:

- “Clients” on page 11
- “AIX”
- “Windows” on page 79

AIX

Context:

You can use the **setupclient** command or a series of manual steps to start an AIX client. The client driver is automatically started by starting the operating system.

Steps:

Perform the following steps to start the client for AIX:

1. If you are using the **setupclient** command:
 - a. Type **setupclient** and press Enter to start the setup guide.
 - b. Enter the appropriate information, when prompted, to set up the virtual client.
2. To manually start the client:
 - a. Create a directory to use as the mount point (for example, `mkdir /mnt/sansfs`).
 - b. Change to the `base/client/bin` directory, where *base* is the base directory (for example, `chdir /usr/tank/client/bin`).
 - c. Type **stfsdriver -load <PATH to kernel extension> /stfs** and press Enter to load the kernel extension into AIX.
 - d. Type **./stfscient -create [client_name] server_IP_address[:port] -converter 8859-1 [-device=kmname | kmid kernel_extension]** and press Enter to create an instance of the SAN File System client, called *virtual client*, where

client_name The name of the SAN File System client. The default is the host name of the client machine.

server_IP_address The IP address of the Metadata server machine.

port The port of the Metadata server machine. The default port is 1700.

kernel_extension The path to kernel extension, as previously specified to the **stfsdriver** command, or kernel extension identifier that was returned by the **stfsdriver** command.

- e. Type **.stfsmount** *client_name* *mount_point* and press Enter to mount the specified file system, where

client_name The name of the client that you specified in step 2d on page 78.

mount_point The directory that you specified in step 2a on page 78.

Related topics:

- “Clients” on page 11
- “setupstclient” on page 293
- “Windows”
- “stfsclient” on page 294
- “stfsdriver” on page 300
- “stfsmount” on page 302
- “AIX” on page 80

Windows

Context:

In a typical setup, the Windows-based client starts automatically when the machine is booted. If your machine is not set up to start automatically, you can start the client manually.

Steps:

To manually start the client, type **net start stfs** from the command line and press Enter.

To change the automatic-restart setting for the client:

1. Edit the Windows registry by typing **regedt32** and pressing Enter.
2. Navigate to the start key:
\\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services \STFS
3. Double-click **Start** and change the value to **3** to enable the client from starting automatically when the machine is booted.
4. Click **OK** to close the window.
5. Reboot.

Related topics:

- “Clients” on page 11
- “AIX” on page 78

- “Windows” on page 81

Stopping a client

The steps to stop the client are specific to the operating system running on the client machine.

Related topics:

- “Clients” on page 11
- AIX
- Windows

AIX

Context:

You can use the **rmclient** command or a series of manual steps to stop the AIX client. The client driver is automatically stopped by stopping the operating system.

Steps:

Perform the following steps to stop the client for AIX:

1. If you are using the **rmclient** command:
 - a. Type **rmclient** and press Enter to unmount SAN File System, remove the SAN File System virtual client, and unload the SAN File System driver from the client system.
2. If you are manually stopping the client:
 - a. Type **stfsclient -query -kmname <PATH to kernel extension or kmit that was returned by stfsdriver command>** and press Enter to display a list of installed clients.
 - b. Type **stfsumount *client_name* *mount_point*** and press Enter to unmount the global namespace, where

<i>client_name</i>	The name of the SAN File System client. The default is the host name of the client machine.
<i>mount_point</i>	The directory that you created to mount to the global namespace.
 - c. Type **./stfsclient -destroy *client_name* -kmname | *kmid* <PATH to kernel extension or kmid that was returned from stfsdriver command>** and press Enter to remove the client instance, where

<i>client_name</i>	The name of the SAN File System client. The default is the host name of the client machine.
<i>client_IP_address</i>	The IP address of the client machine.
 - d. Type **stfsdriver -unload ./stfs** and press Enter to unload the kernel module.

Related topics:

- “Clients” on page 11
- “rmstclient” on page 292
- “stfsclient” on page 294
- “stfsdriver” on page 300
- “stfsumount” on page 305

- “AIX” on page 78
- “Windows”

Windows

Context:

You cannot stop the Windows clients without shutting down the entire system. If the automatic-restart setting is enabled, it will restart automatically when the system is restarted; however, you can disable the automatic-restart setting.

Steps:

To change the automatic-restart setting for the client:

1. Edit the Windows registry by typing **regedt32** and pressing Enter.
2. Navigate to the start key:
`\\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services \STFS`
3. Double-click **Start** and change the value to **2** to disable the client from starting automatically when the machine is booted.
4. Click **OK** to close the window.
5. Reboot.

Related topics:

- “Clients” on page 11
- “Windows” on page 79
- “AIX” on page 80

Viewing client-session details

Steps:

Perform the following steps to display the details for a specific client session:

1. Click **Manage Servers and Clients** → **Client Sessions** from the My Work frame.
2. Select a client session.
3. Click **Details** from the drop-down box in the table header.
4. Click **Go**.

Related topics:

- “Clients” on page 11
- “Client Sessions” on page 355
- “Granting root privileges to a client” on page 76
- “Listing client sessions” on page 76
- “Listing clients with root privileges” on page 77
- “Isclient” on page 179
- “Revoking root privileges from a client” on page 77

Viewing client-session statistics

Steps:

Perform the following steps to view statistics for client sessions:

1. Click **Monitor System** → **Statistics** from the My Work frame.
2. Click the **Client Sessions** tab.
3. Click **OK** to view the statistics.
4. Click **Close** to close the Statistics panel.

Related topics:

- “Creating a report” on page 57
- “Isclient” on page 179
- “Statistics - Client Sessions” on page 413
- “Viewing statistics” on page 61

Managing the cluster

A SAN File System cluster is a set of Metadata servers and engines with one Metadata server running on each engine. The servers in a cluster communicate with each other and with SAN File System clients over your existing IP network. A cluster provides a single point of control for administrative and service operations.

You can perform the following cluster tasks from the SAN File System console:

Related topics:

- “Changing active cluster states”
- “Configuring cluster tuning” on page 83
- “Listing cluster configuration” on page 83
- “Starting the cluster” on page 84
- “Stopping the cluster” on page 84
- “Viewing cluster details” on page 85
- “Viewing cluster statistics” on page 85
- “Viewing cluster tuning details” on page 85
- “Viewing the cluster software version” on page 86

Changing active cluster states

Prerequisites:

You must have Operator or Administrator privileges to perform this task.

Steps:

Perform the following steps to change the active cluster state:

1. Click **Manage Servers and Clients** → **Cluster** from the My Work frame.
2. Click **Change State** from the drop-down box in the table header.

Note: A cluster cannot be changed from an inactive (Not Running, Forming) or unknown state, to an active state (Online, Partly Quiescent, Fully Quiescent, and Offline); or from an active to an inactive state using the SAN File System console.

3. Click **Go**.
4. Select a new active state for the cluster.
5. Click **OK** to confirm the state change.

Related topics:

- “Change State of Cluster” on page 352
- “Cluster” on page 16
- “Cluster states” on page 19
- “Managing the cluster” on page 82
- “quiescecluster” on page 229
- “resumecluster” on page 236
- “startcluster” on page 261
- “stopcluster” on page 280

Configuring cluster tuning

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to modify the tuning for the SAN File System cluster:

1. Click **Manage Servers and Clients**→ **Cluster** from the My Work frame.
2. Click **Properties** from the drop-down box in the table header.
3. Click **Go**.
4. Click the **Tuning** tab.
5. Select the check box to enable **Pool Space Reclamation**, or clear the check box to disable this action.
6. If enabled, enter an interval from 1 to 1440 minutes.
7. Click **OK** to enable the new settings.

Related topics:

- “chclusterconfig” on page 157
- “Cluster” on page 16
- “Cluster Properties - Tuning” on page 361
- “Cluster states” on page 19
- “Managing the cluster” on page 82
- “statcluster” on page 267

Listing cluster configuration

Steps:

To view a listing of the SAN File System cluster configuration, click **Manage Servers and Clients** → **Cluster** from the My Work frame.

Related topics:

- “Cluster” on page 16
- “Cluster” on page 356
- “Cluster states” on page 19
- “Managing the cluster” on page 82
- “statcluster” on page 267

Starting the cluster

Prerequisites:

You must have Operator or Administrator privileges to perform this task.

Context:

If the cluster is not already running (that is, specifically in the “not running” state), you can start it from the SAN File System console.

Steps:

Perform the following steps to start the SAN File System cluster:

Note: Starting the cluster and waiting until all Metadata servers are online takes one to five minutes depending on how many Metadata servers are in the cluster.

1. Click **Manage Servers and Clients** → **Cluster** from the My Work frame.
2. Click **Start** from the drop-down box in the table header.
3. Click **Go**.

Related topics:

- “Cluster” on page 16
- “Cluster” on page 356
- “Managing the cluster” on page 82
- “startcluster” on page 261
- “Stopping the cluster”
- “User roles” on page 46

Stopping the cluster

Prerequisites: You must have Operator or Administrator privileges to perform this task.

Context:

If the cluster is not already in the “not running” state, you can stop it from the SAN File System console.

Steps:

Perform the following steps to stop the SAN File System cluster:

1. Click **Manage Servers and Clients** → **Cluster** from the My Work frame.
2. Click **Stop** from the drop-down box in the table header.
3. Click **Go**.
4. Verify that you have selected the servers that you want to stop, and click **OK** to confirm the cluster stop operation.

Related topics:

- “Cluster” on page 16
- “stopcluster” on page 280
- “Starting the cluster” on page 84
- “Stop Cluster” on page 422
- “User roles” on page 46

Viewing cluster details

Steps: Perform the following steps to display the details for the SAN File System cluster:

1. Click **Manage Servers and Clients** → **Cluster** from the My Work frame.
2. Click **Properties** from the drop-down box in the table header.
3. Click **Go**.
4. Click the **Details** tab.

Related topics:

- “Cluster” on page 16
- “Cluster” on page 356
- “Cluster Properties - Details” on page 360
- “statcluster” on page 267

Viewing cluster statistics

Steps:

Perform the following steps to view statistics for the cluster:

1. Click **Monitor System** → **Statistics** from the My Work frame.
2. Click the **Cluster** tab.
3. Click **OK** to view the statistics.
4. Click **Close** to close the Statistics panel.

Related topics:

- “Creating a report” on page 57
- “statcluster” on page 267
- “Statistics - Cluster” on page 414
- “Viewing statistics” on page 61

Viewing cluster tuning details

Steps: Perform the following steps to display the tuning details for the SAN File System cluster:

1. Click **Manage Servers and Clients** → **Cluster** from the My Work frame.
2. Click **Properties** from the drop-down box in the table header.
3. Click **Go**.
4. Click the **Tuning Details** tab.

Related topics:

- “Cluster” on page 16
- “Cluster Properties - Tuning Details” on page 361
- “statcluster” on page 267

Viewing the cluster software version

Steps: Perform the following steps to view the operational and pending cluster software versions:

1. Click **Manage Servers and Clients** → **Cluster** from the My Work frame.
2. View the current software version in the **Operational Software Version** field.
3. View the next version that the software will move to in the **Pending Software Version** field.

Note: Alternately, you can view the operational and pending software versions from the **Cluster Properties – Details** panel.

Related topics:

- “Cluster” on page 16
- “Cluster” on page 356
- “Cluster Properties - Details” on page 360
- “Managing the cluster” on page 82
- “statcluster” on page 267

Managing disaster recovery

The SAN File System console enables you to create and delete files to assist in disaster recovery. In addition, there are several disaster recovery tasks that can be performed from the Administrative command-line interface.

Several of the restoring tasks depend on having first executed certain backup tasks. The method of restoration will depend a great deal upon whether you chose the LUN or API method of backup.

Related topics:

- “Backup and restore” on page 5
- “Managing backups” on page 72
- “Creating a recovery file” on page 87
- “Deleting a recovery file” on page 87
- “Restoring the master console” on page 88
- “Restoring the engine hardware and operating system” on page 88
- “Restoring SAN connectivity” on page 89
- “Restoring SAN File System software” on page 90
- “Restoring SAN File System cluster configuration” on page 90
- “Restoring SAN File System metadata” on page 91

- “Restoring SAN File System clients” on page 92
- “Restoring SAN File System user data” on page 93

Creating a recovery file

Prerequisites:

You must have Operator or Administrator privileges to perform this task.

Steps:

Perform the following steps to create a file for disaster recovery:

Note: Recovery files are cluster-wide, not server-specific.

1. Click **Maintain System** → **Disaster Recovery** from the My Work frame.
2. Click **Create** from the drop-down box in the table header.
3. Click **Go**.
4. Select the check box to **Create a new recovery file** or select the check box for a **Forced Create**, which will overwrite an existing file.

Attention: When you overwrite an existing file, metadata recovery of items from that file may not be possible.

5. Type a name for the newly created file, or select the **Existing Recovery File** to overwrite from the drop-down menu.
6. Click **OK** to confirm the creation of the new file or to overwrite the existing one.
7. Click **Maintain System** → **Disaster Recovery** from the My Work frame to verify that the recovery file was created.

Related topics:

- “Checking metadata” on page 112
- “Cluster” on page 16
- “Create Recovery File” on page 372
- “Managing disaster recovery” on page 86
- “mkdrfile” on page 220

Deleting a recovery file

Prerequisites:

You must have Operator or Administrator privileges to perform this task.

Steps:

Perform the following steps to delete a disaster recovery file:

1. Click **Maintain System** → **Disaster Recovery** from the My Work frame.
2. Select a recovery file for deletion.
3. Click **Delete** from the drop-down box in the table header.
4. Click **Go**.

Attention: When you delete an existing recovery file, metadata recovery of items from that file may not be possible.

5. Click **Delete** to confirm the file deletion.

Related topics:

- “Checking metadata” on page 112
- “Disaster Recovery” on page 381
- “Managing disaster recovery” on page 86
- “rmdrfile” on page 241

Listing recovery files

Steps:

To display a list of all recovery files, click **Maintain System** → **Disaster Recovery** from the My Work frame.

Related topics:

- “Disaster Recovery” on page 381
- “Managing disaster recovery” on page 86
- “Isdrfile” on page 187

Restoring the master console

Steps:

Perform these steps to restore the hardware and operating system for the master console.

1. Determine if the hardware for the master console is working properly. If so, review information about recovering the hard drives (if necessary) as well as recovering the software.
2. If the hardware for the master console is not working properly,
 - Refer to the refer to the *IBM @server xSeries 305 Hardware Maintenance Manual and Troubleshooting Guide* to resolve problems with the hardware.
 - Refer to the *Planning, Installation and Configuration Guide* for information about installing the master console.

Restoring the engine hardware and operating system

Steps:

Perform these steps to restore the hardware and operating system for each engine in the SAN File System cluster.

1. Verify that there is no damage to the hardware and that the engine boots properly. If you suspect a problem with any of the hardware components, troubleshoot an engine to resolve the problem.
2. Verify that there is no damage to the master console and that it boots properly.
 - If you suspect a problem with any of the hardware components in the master console, refer to the to the *IBM @server xSeries 305 Hardware Maintenance Manual and Troubleshooting Guide* to resolve those problems.
 - If you suspect a problem with the software or the hard disk drive, troubleshoot the master console to resolve the problem.

3. From the master console, point the Web browser to the URL of the RSA II adapter on the engine and access the RSA II adapter to set up a remote console to the engine. This interface allows you to use the master console as your display and keyboard for the engine.

Note: Instead of using the RSA II Web interface from the master console, you can directly attach a keyboard and display to the engine. However, make sure that you attach the display to the VGA port of the RSA II card on the engine and not to the video port on the engine itself.

4. Determine if the boot drives for each engine hosting a Metadata server still have an intact SAN File System configuration and executable files (undamaged).
5. If there are corrupt or damaged configuration and executable files, attempt to recover the damaged files from the mirrored boot drive. If you cannot recover the damaged files from the mirrored boot drive:
 - a. Load the Disaster Recovery CD into the CD-ROM drive on the engine.
 - b. Reboot the engine using one of the following methods:
 - Open a bash shell prompt and enter **init 6**.
 - Press the Reset button on the front panel of the engine.
 - Power off the engine and then power it back on.
 - c. When you receive a warning prompt that the entire hard drive will be overwritten, respond by entering **y**.
 - d. After the operating system has been reloaded, the engine will eject the Disaster Recovery CD and automatically reboot.

Restoring SAN connectivity

Steps:

If the system was backed up using the LUN method, perform these steps on each engine in the SAN File System cluster to restore SAN connectivity:

1. Verify that the engines hosting the Metadata servers are connected to the SAN in the same configuration that existed at the point of the last backup operation (make sure the Metadata servers can see the same LUNs that existed prior to the unexpected outage).
2. If the LUN mapping has changed, use the device management tools for the storage subsystem or management tools for the SAN to re-create the old LUN map. After creating the old LUN map, reboot the Metadata server so that the changes to the LUN map are visible to the Metadata server.
3. If LUN contents were lost or corrupted, use the copy services facility of the storage subsystem to restore all LUN data (both metadata and user file data).

If the system was backed up using the API method, perform these steps on each engine in the SAN File System cluster to restore SAN connectivity.

1. Verify that the engines hosting the Metadata servers are connected to the SAN in the same configuration that existed at the point of the last backup operation (make sure the Metadata servers can see the same LUNs that existed prior to the unexpected outage).
2. If the LUN mapping has changed, use the device management tools for the storage subsystem or management tools for the SAN to re-create the old LUN

map. You can also choose to restore data onto a new LUN map. However, if you do so, you will have to manually run some of the steps used to restore metadata.

Restoring SAN File System software

Steps:

If you had to use the Disaster Recovery CD-ROM to recover the operating system for the engine, perform these steps on each engine in the SAN File System cluster to restore the Metadata server and Administrative server software.

1. Reinstall the software for the Metadata server.
 - a. Make sure that you are logged into the engine as root.
 - b. From a shell prompt on the engine, change to the directory where the Metadata server software package is installed.
cd /usr/tank/packages
 - c. Install the Metadata server software package using the following command:
bash# rpm -ivh Metadata_server_package_name.rpm
 - d. Install the Qlogic software using the following command:
bash# rpm -i snia_qlogic_hba*rpm
2. Reinstall the software package for the Administrative server.
 - a. Install the Administrative server software package using the following command:
bash# rpm -ivh Admin_server_package_name.rpm

Restoring SAN File System cluster configuration

Steps:

If the system was backed up using the LUN method, perform these steps to restore the cluster configuration information:

1. If you have previously saved the configuration files to another location, copy these files onto the boot drive for the engine.
 - a. Copy Tank.Bootstrap to /usr/tank/server/config.
 - b. Copy Tank.Config to /usr/tank/server/config.

Note: If you have saved any other administrative configuration files, you can reference them when restoring the SAN File System metadata configuration.
2. If the cluster bootstrap file, Tank.Bootstrap, is corrupted or missing, you can attempt to re-create the contents of that file using information from the metadata LUNs:
 - a. Use the **tank lsdisklabel -device** command to find the master volume. If you cannot remember which device is your master volume, this is an iterative process of searching all suspected master volume devices until the command indicates you have found a valid master volume.
 - b. Use the **tank extractbootrecord** command to regenerate Tank.Bootstrap from the master volume.
 - c. Use the **tank resetcluster** command to reinitialize the master volume for subsequent rebuilding of the cluster configuration.

- d. Use the **addserver** command for all subordinate MDS engines to re-create the cluster definition.

If the system was backed up using the API method, perform these steps to restore the cluster configuration information:

1. If you have previously saved the configuration files to another location, copy these files onto the boot drive for the engine.
 - a. Copy Tank.Bootstrap to /usr/tank/server/config.
 - b. Copy Tank.Config to /usr/tank/server/config.

Note: If you have saved any other administrative configuration files, you can reference them when restoring the SAN File System metadata configuration.

2. If you suspect that the metadata LUNs are corrupted, you can perform these steps to re-create the cluster definition:
 - a. Delete all Tank.Bootstrap and Tank.Config files from your MDS engines.
 - b. Start the **tank** binary on your master MDS with the **install** option rather than *normal* option. This will create new Tank.Bootstrap and Tank.Config files on your MDS master. Be sure to specify the same cluster name that was used prior to the disaster:
 - c. Now start the master MDS with **tank normal** command.
 - d. Use the **addserver** command to add all subordinate MDS engines. This will create new Tank.Bootstrap and Tank.Config files on the subordinates.

Restoring SAN File System metadata

Steps:

If the system was backed up using the API method, perform these steps to the SAN File System metadata:

Note: If the system was backed up using the LUN method, the metadata was restored when you restored SAN connectivity.

1. Verify that all Metadata servers in the cluster are online and that the cluster is running.

```
tanktool lsserver -state online
```

All of the online Metadata servers in the cluster are displayed.

2. Copy the system-metadata disaster-recovery file (and the scripts) that you had previously backed up to /usr/tank/server/DR on the master Metadata server.
3. Use the TankSysCLI.auto script:

- a. Edit the script TankSysCLI.auto for information about how the script is used and any changes that may need to be made to the script.

```
#####
# CLI Commands to create Storage Pools, Containers, Service Classes and
# Policy Sets.
# These commands need NO manual intervention.
#####
```

- b. Run the script TankSysCLI.auto.

```
tanktool -script /usr/tank/server/DR/TankSysCLI.auto
```

- c. If any errors occurred while running the script, make sure that you resolve those errors before continuing.

4. Use the TankSysCLI.volume script:
 - a. Edit /usr/tank/server/DR/TankSysCLI.volume and modify it to match your current SAN settings. It also contains usage information as well as information about any changes that may need to be made to the script.


```
#####
# CLI Commands to add Volumes to Storage pools.
# These commands need manual intervention.
# The device names were as they appeared during backup.
# Please make sure that the device names appearing here actually
# exist and have correct sizes and if not edit the device names to
# correct values.
# The System MASTER volume has to be specified in tank install command
# and therefore has no corresponding CLI.
# The other System Volumes can either be specified in tank install
# command, or, added using the CLI command, which appears inside comments
# forthis reason.
#####
```
 - b. Run the script TankSysCLI.volume.


```
tanktool -script /usr/tank/server/DR/TankSysCLI.volume
```
 - c. If any errors occurred while running the script, make sure that you resolve those errors before continuing.
5. Use the TankSysCLI.attachpoint script:
 - a. Edit /usr/tank/server/DR/TankSysCLI.attachpoint to verify the settings. It also contains usage information as well as information about any changes that may need to be made to the script.


```
#####
# CLI Commands to attach containers.
# These commands need manual intervention.
# All the "mkdir" and "attachcontainer" commands should be run in the
# order given.
# The "mkdir" command should be run on a client to recreate the directory
# path before running the following attachcontainer CLI commands.
#####
```
 - b. If all filesets are attached only to the root directories of other filesets, run the script TankSysCLI.attachpoint.


```
tanktool -script /usr/tank/server/DR/TankSysCLI.attachpoint
```

Note: If you have any filesets attached to directories, you have to reattach them manually.
 - c. If any errors occurred while running the script, make sure that you resolve those errors before continuing.
6. Grant privileges to those clients that require root or Administrator access to SAN File System using the **chclusterconfig -privclient** command.

Related topics:

- “Administrative commands” on page 142

Restoring SAN File System clients

SAN File System clients are access points to the SAN File System. Therefore, clients are not backed up. To restore a SAN File System clients, you can perform the normal client installation procedure, which is described in the *Planning, Installation, and Configuration Guide*.

Restoring SAN File System user data

Steps:

If the system was backed up using the API method, perform these steps to restore user data:

Note: If the system was backed up using the LUN method, user data was restored when you restored SAN connectivity.

1. From a client, mount the SAN File System at its usual mount point. The top of the subdirectory tree (the portion of the subdirectory tree that consists of the fileset names) should be visible from the client.
2. Restore the files onto that mount point. Follow the procedures for the backup and recovery application used by the customer to back up the files.
3. If the customer has followed the guidelines in the *Planning, Installation, and Configuration Guide* for backup and recovery, restore files to the Windows filesets from a Windows client and restore files to the AIX filesets from an AIX client.

Managing engines

Within SAN File System, an engine is the hardware (which is based on the IBM xSeries(R) platform) on which a Metadata server and an Administrative server run.

The administrative infrastructure on each engine allows an administrator to monitor and control SAN File System from a standard Web browser or an administrative command line interface. The two major components of the infrastructure are an Administrative agent, which provides access to administrative operations, and a Web server that is bundled with the console services and servlets that render HTML for the administrative browsers.

You can perform the following tasks on SAN File System engines:

Related topics:

- “Collecting diagnostic data on engines” on page 94
- “Listing engines” on page 95
- “Powering off the engine” on page 95
- “Powering on the engine” on page 96
- “Restarting the engine” on page 97
- “Starting the Configuration/Setup Utility” on page 98
- “Viewing engine fan status” on page 98
- “Viewing engine power status” on page 99
- “Viewing engine statistics” on page 99
- “Viewing the engine status summary” on page 100
- “Viewing engine temperatures” on page 100
- “Viewing engine timeouts” on page 101
- “Viewing engine voltage status” on page 101
- “Viewing vital engine data” on page 102

Accessing an engine through SSH

Prerequisites:

Before accessing an engine using SSH, the following requirements must be met:

- A SAN File System administrative user account must be set up for use in signing on to the SAN File System console.
- A Linux account must be set up on the engine to be accessed for use in signing on to the SSH session.
- If the service representative is accessing the master console remotely:
 - The customer must have previously initiated a VPN connection with the service representative.
 - The service representative must have established the VPN connection.

Steps:

Perform these steps to establish an SSH session to any of the engines in the SAN File System cluster.

1. From the master console, use one of these methods to access the engine:
 - Open a shell prompt and type **putty.exe -ssh engine_IP_address**, where *engine_IP_address* is the IP address of the engine to be accessed.

Note: If you used SSH to establish a remote session with the master console, type this command from that session to establish an SSH session between the master console and the engine.

- Click **Start→Programs→PuTTY→PuTTY**.
 - a. Type the IP address of the engine to be accessed.
 - b. Select SSH as the protocol.
 - c. Click **Open**.
2. After the session is established, log in using a Linux user ID and password.

Result:

After connecting to the engine, you can perform these activities:

- Access the SAN File System administrative command-line interface (CLI) to run SAN File System commands. These commands provide the ability to manage engines, Metadata servers, and Administrative servers.
- Access operating-system commands to enable or disable tracing, obtain dumps, and stop or start applications.

Related topics:

- “Administrative commands” on page 142

Collecting diagnostic data on engines

Prerequisites:

You must have Administrator or Operator privileges to perform this task.

Steps:

You can collect diagnostic data for engines using the one-button data collection function from the SAN File System console.

Perform the following steps to collect engine diagnostics:

1. Click **Maintain System**→**Collect Diagnostic Data** from the My Work frame.
2. Select the engine or engines on which you would like to collect diagnostics.
3. Click **OK**.

Result:

After the data is collected, a progress panel will appear that displays a listing of all the engines on which diagnostics were created, and the file path where the data is stored.

Related topics:

- “collectdiag” on page 166
- “Collect Diagnostic Data” on page 363
- “Engines” on page 23
- Chapter 3, “Monitoring”, on page 57
- “Viewing engine statistics” on page 99
-

Listing engines

Steps:

To display a list of all engines, click **Maintain System** → **Engines** from the My Work frame.

Related topics:

- “Engines” on page 23
- “Engines” on page 382
- “lsengine” on page 189
- “Viewing the engine status summary” on page 100
- “Viewing engine temperatures” on page 100
- “Viewing engine fan status” on page 98
- “Viewing engine voltage status” on page 101
- “Viewing engine timeouts” on page 101
- “Viewing vital engine data” on page 102
- “Viewing engine power status” on page 99

Powering off the engine

Prerequisites:

You must have Administrator or Operator privileges to perform this task.

Context:

After you power off the engine, wait at least five seconds before you power on the engine again.

Steps:

Perform the following steps to manually power off the engine:

1. Click **Maintain System** → **Engines** from the My Work frame.
2. Select an engine.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Power** tab.
6. Click **Change Power State**.
7. Select the **Power Off**, **Forced Power Off**, or **Scheduled Power On or Off** radio button.
8. Fill in any appropriate fields.
9. Click **OK** to power off the engine.

Note: You can also manually power off the engine by shutting down the operating system (refer to your operating-system documentation for procedures), and then pressing the Power-control button on the front of the engine. This will put the engine in Standby mode. To cause an immediate shutdown of the engine and force the power off, press and hold the Power-control button for more than four seconds (for example, when the operating system stops functioning).



CAUTION:

<2-19> The power control button on the device and the power switch on the power supply do not turn off the electrical current supplied to the device. The device also might have more than one power cord. To remove all electrical current from the device, ensure that all power cords are disconnected from the power source.

Standby mode refers to the condition in which the engine operating system is not running and all core logic is shut down except the service processor.

Related topics:

- “Change Power State of Engine” on page 351
- “Engines” on page 23
- “Engines Properties - Power” on page 385
- “Listing engines” on page 95
- “Powering on the engine”
- “stopengine” on page 281
- “Viewing engine power status” on page 99

Powering on the engine

Prerequisites:

You must have Administrator or Operator privileges to perform this task.

Verify that all external devices are powered on and that the engine power cords are plugged into the power source.

Context:

If the engine is powered on and a power failure occurs, the engine restarts automatically when power is restored.

While the engine is in the process of powering on, the power-on LED on the front of the engine is lit.

Steps:

Perform the following steps to power on the engine:

1. Click **Maintain System** → **Engines** from the My Work frame.
2. Select an engine.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Power** tab.
6. Click **Change Power State**.
7. Select the **Power On** or **Scheduled Power On** or **Off** radio button
8. Fill in any appropriate fields.
9. Click **OK** to change the state.

Note: You can also manually power on the engine by pressing the Power-control button on the front of the engine. You can also install a circular disk over the Power-control button to prevent accidental manual power-off. This disk, known as the power-control-button shield, is included with your engine.

Related topics:

- “Change Power State of Engine” on page 351
- “Engines” on page 23
- “Engines Properties - Power” on page 385
- “Listing engines” on page 95
- “Powering off the engine” on page 95
- “startengine” on page 262
- “Viewing engine power status” on page 99

Restarting the engine

Prerequisites:

You must have Administrator or Operator privileges to perform this task.

Steps:

Perform the following steps to reboot a specific engine:

1. Click **Maintain System** → **Engines** from the My Work frame.
2. Select an engine.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.

5. Click the **Power** tab.
6. Click **Change Power State**.
7. Select the **Power Cycle** radio button.
8. Click **OK** to reboot the engine

Related topics:

- “Change Power State of Engine” on page 351
- “Engines” on page 23
- “Engines Properties - Power” on page 385
- “Listing engines” on page 95
- “Powering off the engine” on page 95
- “Powering on the engine” on page 96
- “restartengine” on page 235
- “Viewing engine power status” on page 99

Starting the Configuration/Setup Utility

Steps:

The Configuration/Setup Utility program is a menu-driven utility that is part of the BIOS code that comes with the Model 1RX. You can use it to perform these functions:

- Configure serial port assignments.
- Change the drive startup sequence.
- Enable USB keyboard and mouse support.
- Resolve configuration conflicts.
- Set the date and time.
- Set passwords.

Perform the following steps to start the Configuration/Setup Utility program:

1. Power ON the engine and watch the monitor screen.
2. When the message Press F1 for Configuration/Setup appears, press F1.

Note: To run the Configuration/Setup Utility, you must use the highest-level password available for the engine when you power it on. For example, if an administrative password is set for the engine, you must use it when you power on the engine instead of using the power-on password.

If you do not use the administrative password, you will have access to only a limited set of Configuration/Setup Utility functions.

3. Follow the instructions that appear on the screen.

Related topics:

- “Powering on the engine” on page 96

Viewing engine fan status

Steps:

Perform the following steps to display a summary of the health of a specific engine:

1. Click **Maintain System** → **Engines** from the My Work frame.
2. Select an engine.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Fans** tab.

Related topics:

- “Engines” on page 23
- “Engines Properties - Fans” on page 384
- “Listing engines” on page 95
- “statengine” on page 272
- “Viewing engine power status”
- “Viewing the engine status summary” on page 100
- “Viewing engine temperatures” on page 100
- “Viewing engine timeouts” on page 101
- “Viewing engine voltage status” on page 101
- “Viewing vital engine data” on page 102

Viewing engine power status

Steps:

Perform the following steps to display the power status for a specific for a engine:

1. Click **Maintain System** → **Engines** from the My Work frame.
2. Select an engine.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Power** tab.

Related topics:

- “Engines” on page 23
- “Engines Properties - Power” on page 385
- “Listing engines” on page 95
- “statengine” on page 272
- “Viewing engine fan status” on page 98
- “Viewing the engine status summary” on page 100
- “Viewing engine temperatures” on page 100
- “Viewing engine timeouts” on page 101
- “Viewing engine voltage status” on page 101
- “Viewing vital engine data” on page 102

Viewing engine statistics

Steps:

Perform the following steps to view statistics for engines:

1. Click **Monitor System** → **Statistics** from the My Work frame.
2. Select the **Engine** tab.
3. Click **OK** to view the statistics.
4. Click **Close** to close the Statistics panel.

Related topics:

- “Creating a report” on page 57
- “Engines” on page 23
- “statengine” on page 272
- “Statistics - Engines” on page 416
- “Viewing statistics” on page 61

Viewing the engine status summary

Steps:

Perform the following steps to display a summary of the health of a specific engine:

1. Click **Maintain System** → **Engines** from the My Work frame.
2. Select an engine.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Summary** tab.

Related topics:

- “Engines” on page 23
- “Listing engines” on page 95
- “Engines Properties - Summary” on page 385
- “statengine” on page 272
- “Viewing engine temperatures”
- “Viewing engine fan status” on page 98
- “Viewing engine voltage status” on page 101
- “Viewing engine timeouts” on page 101
- “Viewing vital engine data” on page 102
- “Viewing engine power status” on page 99

Viewing engine temperatures

Steps:

Perform the following steps to display the temperatures for a specific engine:

1. Click **Maintain System** → **Engines** from the My Work frame.
2. Select an engine.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Temperatures** tab.

Related topics:

- “Engines” on page 23
- “Engines Properties - Temperatures” on page 387
- “Listing engines” on page 95
- “statengine” on page 272
- “Viewing engine fan status” on page 98
- “Viewing engine power status” on page 99
- “Viewing the engine status summary” on page 100
- “Viewing engine timeouts”
- “Viewing engine voltage status”
- “Viewing vital engine data” on page 102

Viewing engine timeouts

Steps:

Perform the following steps to display the timeout settings for a specific engine:

1. Click **Maintain System** → **Engines** from the My Work frame.
2. Select an engine.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Timeouts** tab.

Related topics:

- “Engines” on page 23
- “Engines Properties - Timeouts” on page 389
- “Listing engines” on page 95
- “statengine” on page 272
- “Viewing engine fan status” on page 98
- “Viewing engine power status” on page 99
- “Viewing the engine status summary” on page 100
- “Viewing engine temperatures” on page 100
- “Viewing engine voltage status”
- “Viewing vital engine data” on page 102

Viewing engine voltage status

Steps:

Perform the following steps to display the voltage status for a specific engine:

1. Click **Maintain System** → **Engines** from the My Work frame.
2. Select an engine.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Voltages** tab.

Related topics:

- “Engines” on page 23
- “Engines Properties - Voltages” on page 390

- “Listing engines” on page 95
- “statengine” on page 272
- “Viewing engine fan status” on page 98
- “Viewing engine power status” on page 99
- “Viewing the engine status summary” on page 100
- “Viewing engine temperatures” on page 100
- “Viewing engine timeouts” on page 101
- “Viewing vital engine data”

Viewing vital engine data

Steps:

Perform the following steps to display the vital data for a specific engine:

1. Click **Maintain System** → **Engines** from the My Work frame.
2. Select an engine.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Vital Engine Data** tab.

Related topics:

- “Engines” on page 23
- “Engines Properties - Vital Engine Data” on page 390
- “Listing engines” on page 95
- “statengine” on page 272
- “Viewing the engine status summary” on page 100
- “Viewing engine fan status” on page 98
- “Viewing engine voltage status” on page 101
- “Viewing engine timeouts” on page 101
- “Viewing engine temperatures” on page 100
- “Viewing engine power status” on page 99

Managing filesets (containers)

A fileset (except for the global fileset) is a subset of the entire SAN File System global namespace. It serves as the unit of workload for Metadata servers and is also the unit that an administrator specifies to create FlashCopy images that are used in backup procedures.

From a client perspective, a fileset appears to be a regular directory. Users and applications on SAN File System clients can create objects, such as directories and files, within the fileset.

You can perform the following actions involving filesets:

Related topics:

- “Attaching a fileset” on page 103
- “Creating a fileset” on page 104
- “Deleting a fileset” on page 105

- “Detaching a fileset” on page 105
- “Listing filesets” on page 106
- “Changing fileset settings”
- “Viewing fileset details” on page 106
- “Viewing fileset settings” on page 106
- “Viewing fileset statistics” on page 106

Attaching a fileset

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

The attach point (*existing directory/new directory*) appears as a drive and directory on Windows clients or as a file system on UNIX-based clients. It is recommended that you map the name of the directory to its associated fileset by using the same name as the fileset plus an additional suffix to distinguish it as the directory and not the fileset. For example, if you want to attach the fileset named *work* to the root of the global namespace, you would specify “*sanfs*” for the *existing directory* and specify “*work_ap*” for the *new directory*. On a Windows client, the “*work_ap*” directory would appear under the “*sanfs*” drive.

Steps:

Perform the following steps to attach a fileset to the global namespace:

1. Click **Manage Filing** → **Filesets** from the My Work frame.
2. Select the fileset that you want to attach.
3. Click **Attach** from the drop-down box in the table header.
4. Click **Go**.
5. Fill in the appropriate fields.
6. Click **OK** to attach the fileset.

Related topics:

- “Detaching a fileset” on page 105
- “Attach Fileset” on page 348
- “Filesets” on page 24
- “attachcontainer” on page 151

Changing fileset settings

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to change the general settings of a specific fileset:

1. Click **Manage Filing** → **Filesets** from the My Work frame.
2. Select a fileset.

3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click on the **General Settings** tab.
6. Edit the appropriate fields.
7. Click **OK** to save the new settings.

Related topics:

- “Filesets” on page 24
- “Fileset Properties - General Settings” on page 392
- “chcontainer” on page 159

Creating a fileset

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

Although SAN File System allows you to create nested filesets, it is not recommended for several reasons:

- You cannot access a child fileset if the parent fileset is unavailable.
- You cannot revert a FlashCopy image of a fileset that has nested filesets attached to it (for example, if /fileset1/fileset2, you must detach fileset2, revert to a FlashCopy image of fileset1, and then reattach fileset2).
- You cannot make FlashCopy images of multiple filesets (whether the filesets are siblings or nested) in one operation.
- Backup and restore of filesets that are used primarily by one platform should be performed on the primary platform. If the parent and nested filesets are each used primarily by different platforms, you cannot effectively perform backup and recovery from one platform.

Steps:

Perform the following steps to create a fileset:

1. Click **Manage Filing** → **Filesets** from the My Work frame.
2. Click **Create** from the drop-down box in the table header.
3. Click **Go**.
4. Fill in the appropriate fields.
5. Click **OK** to create the fileset.
6. Click **Manage Filing** → **Filesets** from the My Work frame to verify that the fileset was created.

Related topics:

- “Attaching a fileset” on page 103
- “Filesets” on page 24
- “Create a Fileset” on page 365
- “Deleting a fileset” on page 105
- “Changing fileset settings” on page 103
- “mkcontainer” on page 217

Deleting a fileset

Prerequisites: You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to delete a fileset:

1. Click **Manage Filing** → **Filesets** from the My Work frame.
2. Select the fileset that you want to delete.
3. Click **Delete** from the drop-down box in the table header.
4. Click **Go**.
5. Select the deletion method.
6. Click **OK**.
7. Click **OK** to confirm the deletion.

Related topics:

- “Delete Filesets” on page 375
- “Filesets” on page 24
- “rmcontainer” on page 239

Detaching a fileset

Prerequisites:

You must have Administrator privileges to perform this task.

The fileset that you want to detach must not have subfilesets. You must detach any subfilesets before detaching the desired fileset.

Context:

The directory to which the fileset was attached is deleted after the fileset is detached.

Steps:

Perform the following steps to detach a fileset from the global namespace:

1. Click **Manage Filing** → **Filesets** from the My Work frame.
2. Select the fileset that you want to detach.
3. Click **Detach** from the drop-down box in the table header.
4. Click **Go**.
5. Select the appropriate detachment method.
6. Click **OK** to confirm the detachment.

Related topics:

- “Attaching a fileset” on page 103
- “Filesets” on page 24
- “Detach Filesets” on page 377
- “detachcontainer” on page 168

Listing filesets

Steps:

To display a list of all filesets, click **Manage Filing** → **Filesets** from the My Work frame.

Related topics:

- “Filesets” on page 24
- “Filesets” on page 394
- “Iscontainer” on page 183
- “Viewing fileset details”
- “Viewing fileset settings”

Viewing fileset details

Steps: Perform the following steps to display the details of a specific fileset:

1. Click **Manage Filing** → **Filesets** from the My Work frame.
2. Select a fileset.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Details** tab.

Related topics:

- “Filesets” on page 24
- “Fileset Properties - Details” on page 391
- “Iscontainer” on page 183
- “Viewing fileset settings”

Viewing fileset settings

Steps: Perform the following steps to display the settings of a specific fileset:

1. Click **Manage Filing** → **Filesets** from the My Work frame.
2. Select a policy.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **General Settings** tab.

Related topics:

- “Filesets” on page 24
- “Fileset Properties - Details” on page 391
- “Iscontainer” on page 183
- “Viewing fileset details”

Viewing fileset statistics

Steps:

Perform the following steps to view statistics for a fileset:

1. Click **Monitor System** → **Statistics** from the My Work frame.
2. Select the **Fileset** tab.
3. Click **OK** to view the statistics.
4. Click **Close** to close the Statistics panel.

Related topics:

- “Creating a report” on page 57
- “Filesets” on page 24
- “Iscontainer” on page 183
- “statcontainer” on page 271
- “Statistics - Filesets” on page 417
- “Viewing statistics” on page 61

Managing FlashCopy images

A FlashCopy image is a space-efficient, read-only copy of the contents of a fileset in a SAN File System global namespace at a particular point in time. A FlashCopy image can be used with standard backup tools available in your environment to create backup copies of files on tape.

You can perform the following FlashCopy image tasks:

Related topics:

- “Creating a FlashCopy image”
- “Deleting a FlashCopy image” on page 108
- “Listing FlashCopy images” on page 108
- “Reverting to a previous FlashCopy image” on page 108
- “Viewing FlashCopy image details” on page 109

Creating a FlashCopy image

SAN File System provides a wizard to step you through the process of creating a FlashCopy image.

Prerequisites:

You must have Administrator, Operator, or Backup privileges to perform this task.

Steps:

Perform the following steps to start the wizard to create a FlashCopy image of one or more filesets:

1. Click **Maintain System** → **Create FlashCopy Images** from the My Work frame.
2. Follow the steps in the wizard to create a FlashCopy image.
3. Click **Maintain System** → **FlashCopy Images** from the My Work frame and verify that the FlashCopy image was created.

Related topics:

- “Create FlashCopy Images of Filesets- Introduction” on page 366
- “FlashCopy images” on page 26

- “Listing FlashCopy images”
- “mkimage” on page 221
- “Reverting to a previous FlashCopy image”
- “Viewing FlashCopy image details” on page 109

Deleting a FlashCopy image

Prerequisites:

You must have Administrator, Operator, or Backup privileges to perform this task.

Steps:

Perform the following steps to delete a FlashCopy image:

1. Click **Maintain System** → **FlashCopy Images** from the My Work frame.
2. Select one or more FlashCopy images to delete.
3. Click **Delete** from the drop-down box in the table header.
4. Click **Go**.
5. Click **OK** to confirm the deletion.

Related topics:

- “FlashCopy images” on page 26
- “Creating a FlashCopy image” on page 107
- “FlashCopy Images” on page 395
- “Listing FlashCopy images”
- “Reverting to a previous FlashCopy image”
- “rmimage” on page 242

Listing FlashCopy images

Steps:

To display a list of all FlashCopy[®] images for a specific fileset, click **Maintain System** → **FlashCopy Images** from the My Work frame.

Related topics:

- “Creating a FlashCopy image” on page 107
- “Deleting a FlashCopy image”
- “FlashCopy Images” on page 395
- “FlashCopy images” on page 26
- “lsimage” on page 192
- “Reverting to a previous FlashCopy image”
- “Viewing FlashCopy image details” on page 109

Reverting to a previous FlashCopy image

Attention: When you revert to a FlashCopy image, all FlashCopy images created after the specified FlashCopy image are deleted. The specified FlashCopy image becomes the primary image for the fileset and no longer appears as an image listed in the .flashcopy directory.

Prerequisites:

You must have Administrator, Operator, or Backup privileges to perform this task.

Steps:

Perform the following steps to revert to a previous FlashCopy image of a fileset:

1. Click **Maintain System** → **FlashCopy Images** from the My Work frame.
2. Select the FlashCopy image that you want the fileset to revert to
3. Click **Revert to** from the drop-down box in the table header.
4. Click **Go**.
5. Click **OK** to revert to specified the FlashCopy image.

Related topics:

- “Creating a FlashCopy image” on page 107
- “Create FlashCopy Images of Filesets- Introduction” on page 366
- “Deleting a FlashCopy image” on page 108
- “FlashCopy images” on page 26
- “Listing FlashCopy images” on page 108
- “reverttoimage” on page 237
- “Viewing FlashCopy image details”

Viewing FlashCopy image details

Prerequisites:

You must have Administrator or Operator privileges to perform this task.

Steps:

Perform the following steps to display the details of a specific FlashCopy image:

1. Click **Maintain System** → **FlashCopy Images** from the My Work frame.
2. Select a FlashCopy image to view.
3. Click **Details** from the drop-down box in the table header.
4. Click **Go**.

Related topics:

- “FlashCopy Images” on page 395
- “FlashCopy images” on page 26
- “Listing FlashCopy images” on page 108
- “lsimage” on page 192

Managing Metadata servers

In addition to providing metadata to clients and managing locks, Metadata servers perform a wide variety of other tasks. They process requests issued by administrators to create and manage filesets, storage pools, volumes, and policy sets, and they enforce the policies defined by administrators to place files in appropriate storage pools and ensure that capacity quotas established for filesets and storage pools are not exceeded.

You can perform the following Metadata server tasks:

Related topics:

- “Changing the master Metadata server”
- “Checking metadata” on page 112
- “Listing Metadata servers” on page 112
- “Starting a Metadata server” on page 112
- “Starting the Metadata server restart service” on page 113
- “Stopping a Metadata server” on page 114
- “Stopping the Metadata server restart service” on page 115
- “Viewing Metadata server details” on page 115
- “Viewing Metadata server networking details” on page 116
- “Viewing Metadata server restart service statistics” on page 116
- “Viewing Metadata server statistics” on page 116

Changing the master Metadata server

Prerequisites:

You must have Operator or Administrator privileges to perform this task.

Before this task can be performed, the current master server must be down, but the cluster must be up. Additionally, the engine hosting the master server must be powered off. The subordinate server that will become the master must be in the “joining” or “forming” state only.

Context:

This task can be performed only at the Administrative command-line interface. You may need to convert a subordinate server to the master server in the event of an irrecoverable loss of the original cluster master server. Such a loss might be due to hardware failures, software failures, or partitioned networks.

Steps:

Perform the following steps to change the master server:

1. Go to the Administrative command-line interface.
2. Make sure the cluster does not have a master Metadata server: In the SAN File System console, if the master engine is running, you can check the **Servers** panel to make sure the master Metadata server is down.

Note: If the master engine is not up, the SAN File System console will not work.

If the master Metadata server is up, then shut it down. If you are not sure if the master engine is up and the SAN File System console does not respond, from a subordinate Metadata server, you can use the command **statcluster -netconfig**, which will tell you the name of the master, and **lsengine**.

3. In certain failure scenarios it is possible, but unlikely, that the SAN File System console is not working, but the master Metadata server is still up. You can always try to log onto the engine hosting the master Metadata server and use the **lsserver** and a **stopserver** commands if necessary. In this scenario it is likely

that the Administrative command-line interface on the master engine will also not work, in which case, you should just shut down the engine as described in a following step.

4. On the subordinate engine that you want to make the new master, a subordinate Metadata server must be running. Log into this engine and use the **lsserver** command to see if a local subordinate Metadata server is running. If not, using the **startserver** command on this engine will start the subordinate Metadata server.
5. Now you must make sure the engine that was hosting the master Metadata server is shut down:
 - a. Use the **lengine master_Metadata_server** command to determine if the engine hosting the master Metadata server is running.
 - b. Use the **stopengine master_Metadata_server** command to shut it down.

Note: You may need to use **-f** flag if the engine is hung. Use **lengine** again to ensure it is down.

6. In the unlikely case that the failure is severe enough that these commands are not working, you will have to manually power off the engine that was hosting the master, and manually reboot the engine you want to become master. Then return to step 4.
7. Enter the **setmaster** command and the name of the server that you want to assign master status. You can test the success of this action by using the **lsserver** command. Once this step is successful, the SAN File System console will work.

Attention: If you receive note of an internal error after performing this operation, please contact your IBM technical support representative.

Post-processing requirements:

After issuing the **setmaster** command, you must manually edit the `tank.property` file on the subordinate server that will become the master and set the **IsMaster** property to `true`.

Related topics:

- “Cluster” on page 16
- “lengine” on page 189
- “lsserver” on page 207
- “Metadata server” on page 33
- “Managing Metadata servers” on page 109
- “Metadata server states” on page 35
- “Powering off the engine” on page 95
- “Powering on the engine” on page 96
- “Servers” on page 410
- “setmaster” on page 252
- “startserver” on page 265
- “Starting a Metadata server” on page 112
- “statcluster” on page 267
- “stopengine” on page 281
- “stopserver” on page 283
- “Stopping a Metadata server” on page 114

- “User roles” on page 46

Checking metadata

Prerequisites:

You must have Administrator privileges to check or repair the metadata.

Steps:

Perform the following steps to check the integrity of the global namespace metadata:

1. Click **Maintain System** → **Check Metadata** from the My Work frame.
2. Fill in the appropriate fields.
3. Optionally, select the **Repair as necessary** check box to repair any corrupt metadata.
4. Click **OK** to start the process.
5. If you selected the **Repair as necessary** check box, click **OK** to confirm that you want to repair any corrupt metadata.

Note: This is a long-running process. After you click **OK**, the Check Metadata Progress panel indicates that the metadata check is in progress. If you close this panel before the metadata check is complete, you can list the long-running processes to determine if it has completed.

Related topics:

- “Check Metadata” on page 352
- “Listing processes” on page 121
- “startmetadatabasecheck” on page 263

Listing Metadata servers

Steps:

To display a list of SAN File System Metadata servers, click **Manage Servers and Clients** → **Servers** from the My Work frame.

Related topics:

- “Isserver” on page 207
- “Metadata server” on page 33
- “Servers” on page 410

Starting a Metadata server

Prerequisites:

You must have Administrator or Operator privileges to perform this task.

Context:

Starting a Metadata server will attempt to put the server or servers selected in an available clustered state, if they are not already in such a state. You should not use the start action to recycle a server.

Notes:

1. If the master server is either “not running” or “unknown,” then starting subordinate servers, without also starting the master, is not allowed.
2. The system will try to match the servers to the cluster’s current target state (online, offline, fully or partly quiescent).

Steps:

Perform the following steps to start a Metadata server:

1. Click **Manage Servers and Clients** → **Servers** from the My Work frame.
2. Select one or more Metadata servers to start.

Note: You can only start those Metadata servers with a state of “not running” or “unknown.”

3. Click **Start** from the drop-down box in the table header.
4. Click **Go**.

Related topics:

- “Changing the master Metadata server” on page 110
- “Isautorestart” on page 175
- “Managing Metadata servers” on page 109
- “Metadata server” on page 33
- “Metadata server states” on page 35
- “Servers” on page 410
- “startautorestart” on page 259
- “startserver” on page 265
- “Stopping a Metadata server” on page 114

Starting the Metadata server restart service

Prerequisites: You must have Operator or Administrator privileges to perform this task.

Context:

The Metadata server restart service is enabled by default.

This service enables probes to periodically check Metadata server condition. They will automatically restart the Metadata servers as appropriate, as they continually monitor the server state.

Note: If a Metadata server has the restart service enabled, and the Metadata server is later manually shut down, the restart service is disabled and you will need to start it again as described in this task.

Steps:

Perform the following steps to enable the Metadata server restart service:

1. Click **Maintain System** → **Server Restart Service** from the My Work frame.
2. Select one or more Metadata servers on which to enable the restart service.

Note: The **Service State** field indicates if the service is currently enabled or disabled for a particular server.

3. Click **Start Service** from the drop-down box in the table header.
4. Click **Go** to start the service on the selected server or servers.

Related topics:

- “lsautorestart” on page 175
- “Metadata server” on page 33
- “Managing Metadata servers” on page 109
- “Metadata server states” on page 35
- “startautorestart” on page 259
- “stopautorestart” on page 279
- “Server Restart Service” on page 408

Stopping a Metadata server

Prerequisites: You must have Administrator or Operator privileges to perform this task.

Context:

This task allows you to gracefully stop the one or more selected Metadata servers. The Metadata servers and their associated states, roles, and filesets are displayed for review and confirmation before the stop is executed.

Steps:

Perform the following steps to stop a Metadata server:

1. Click **Manage Servers and Clients** → **Servers** from the My Work frame.
2. Select one or more Metadata servers to stop.

Note: Only active Metadata servers can be stopped. Active Metadata servers are those in one of the following states:

- offline
 - fully quiescent
 - partly quiescent
 - online
3. Click **Stop** from the drop-down box in the table header.
 4. Click **Go**.
 5. Verify that you selected the Metadata servers that you want to be stopped.
 6. Click **OK** to confirm the stop action.

Related topics:

- “Metadata server” on page 33
- “stopserver” on page 283
- “Managing Metadata servers” on page 109
- “Metadata server states” on page 35

- “Servers” on page 410
- “Starting a Metadata server” on page 112
- “Stop Servers” on page 423

Stopping the Metadata server restart service

Prerequisites: You must have Operator or Administrator privileges to perform this task.

Context:

This service enables probes to periodically check Metadata server condition. The probes will automatically restart the Metadata servers as appropriate, as they continually monitor the server state. This task describes how to disable the service.

Steps:

Perform the following steps to disable the Metadata server restart service:

1. Click **Maintain System** → **Server Restart Service** from the My Work frame.
2. Select one or more Metadata servers on which to disable the restart service.

Note: The **Service State** field indicates if the service is currently enabled or disabled for a particular server.

3. Click **Stop Service** from the drop-down box in the table header.
4. Click **Go** to stop the service on the selected server or servers.

Related topics:

- “Metadata server” on page 33
- “Managing Metadata servers” on page 109
- “Metadata server states” on page 35
- “startautorestart” on page 259
- “stopautorestart” on page 279
- “Server Restart Service” on page 408

Viewing Metadata server details

Steps: Perform the following steps to display Metadata server details:

1. Click **Manage Servers and Clients** → **Servers** from the My Work frame.
2. Select the server for which you want to view details.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Details** tab.

Related topics:

- “Metadata server” on page 33
- “Managing Metadata servers” on page 109
- “Metadata server states” on page 35
- “Isserver” on page 207
- “Server Properties - Details” on page 407
- “Servers” on page 410

Viewing Metadata server networking details

Steps:

Perform the following steps to display Metadata server networking details:

1. Click **Manage Servers and Clients** → **Servers** from the My Work frame.
2. Select the server for which you want to view networking information.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Networking Details** tab.

Related topics:

- “Metadata server” on page 33
- “statserver” on page 276
- “Managing Metadata servers” on page 109
- “Metadata server states” on page 35
- “Server Properties - Networking Details” on page 408
- “Servers” on page 410

Viewing Metadata server statistics

Steps:

Perform the following steps to view statistics for Metadata server:

1. Click **Monitor System** → **Statistics** from the My Work frame.
2. Select the **Server** tab.
3. Click **OK** to view the statistics.
4. Click **Close** to close the Statistics panel.

Related topics:

- “Creating a report” on page 57
- “statserver” on page 276
- “Statistics - Servers” on page 419
- “Viewing statistics” on page 61

Viewing Metadata server restart service statistics

Context:

This service enables probes to periodically check Metadata server condition. They will automatically restart the Metadata servers as appropriate, as they continually monitor the server state. This task describes how to view statistics related to this service.

Steps:

Perform the following steps to view Metadata server restart service statistics:

1. Click **Maintain System** → **Server Restart Service** from the My Work frame.
2. Select a Metadata server for which you want to view restart service statistics.

Note: The **Service State** field indicates if the service is currently enabled or disabled for a particular server.

3. Click **Statistics** from the drop-down box in the table header.
4. Click **Go** to view the statistics for the selected server.
5. Click the **Probe Overview**, **Test Details**, or **Tuning Details** tabs, if not already selected, depending on the type of information that you want to view.

Related topics:

- “Isautorestart” on page 175
- “Isserver” on page 207
- “Metadata server” on page 33
- “Managing Metadata servers” on page 109
- “Metadata server states” on page 35
- “Server Restart Service” on page 408
- “Restart Service Statistics- Probe Overview” on page 401
- “Restart Service Statistics- Test Details” on page 402
- “Restart Service Statistics- Tuning Details” on page 403

Removing down-level Metadata server software

Steps:

Perform the following steps to remove down-level Metadata server software:

1. Stop the engine.
2. Type **stopserver** *server_name*, and press Enter, where *server_name* is the name or IP address of the engine to stop.
3. Type **rpm -e** *package_name*, and press Enter to remove the Metadata server component software package, where *package_name* is the file name of the Metadata server package (for example, `rpm -e storagetank-server-RHLAS-1.0.0.rpm`).

Managing policies

Policies and the rules that they contain are used to assign files to specific storage pools. A storage pool typically contains a set of volumes that provide a specific quality of service for a specific use, such as to store all files for a particular application or a specific business division.

Keep in mind that policies can reference file or path names in ASCII only.

Related topics:

- “Activating a policy” on page 118
- “Creating a policy” on page 119
- “Deleting a policy” on page 120
- “Listing policies” on page 120
- “Changing the rules in a policy” on page 118
- “Viewing policy details” on page 121
- “Viewing policy rules” on page 121

Activating a policy

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to activate a policy:

1. Click **Manage Filing** → **Policies** from the My Work frame.
2. Select the policy that you want to activate.
3. Click **Activate** from the drop-down box in the table header.
4. Click **Go**.
5. Click **OK** to confirm the change.

Related topics:

- “Policies” on page 397
- “Policies and rules” on page 44
- “usepolicy” on page 287

Changing the rules in a policy

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

Policy properties, including any associated rules, are stored in metadata. They are not stored in a file. Therefore, you cannot change the rules in the policy by updating the rules file that you used to initially create the policy.

Each policy has a length limit of 32 KB.

Steps:

Perform the following steps to change the rules in a policy:

1. Click **Manage Filing** → **Policies** from the My Work frame.
2. Select the policy that you want to change.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Rules** tab.
6. Edit the existing rules or add new ones.
7. Click **Apply** to save the changes.

To change the rules in a policy from the ACLI, delete the policy (using the **rmpolicy** command), and recreate the policy (using the **mkipolicy** command) with the updated policy rules.

Related topics:

- “mkipolicy” on page 223

- “Policies” on page 397
- “Policies and rules” on page 44
- “rmpolicy” on page 244

Copying a policy

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

SAN File System provides a wizard to step you through the process of creating a copy of an existing policy.

Steps:

To start the create-policy wizard, click **Manage Filing** → **Create a Policy** from the My Work frame. Select the **Clone Policy** radio button, and then select the policy that you want to copy from the **Existing Policy** drop-down list.

Related topics:

- “Create a Policy - Introduction” on page 372
- “Create a Policy - Add Rules” on page 369
- “Create a Policy - Edit Rules” on page 371
- “Create a Policy - High-Level Settings” on page 371
- “mkpolicy” on page 223
- “Policies and rules” on page 44
- “Policies” on page 397

Creating a policy

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

SAN File System provides a wizard to step you through the process of creating a policy.

Policy properties, including any associated rules, are stored in metadata. They are not stored in a file.

Steps:

Log onto the SAN File System console. Start the Create-policy wizard by clicking **Manage Filing** → **Create a Policy** in the My Work frame. Then, click **Manage Filing** → **Policies** in the My Work frame to verify that the policy was created.

Related topics:

- “Create a Policy - Introduction” on page 372
- “Create a Policy - Add Rules” on page 369

- “Create a Policy - Edit Rules” on page 371
- “Create a Policy - High-Level Settings” on page 371
- “mkpolicy” on page 223
- “Policies and rules” on page 44
- “Policies” on page 397
- “Activating a policy” on page 118

Deleting a policy

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to delete a policy:

1. Click **Manage Filing** → **Policies** from the My Work frame.
2. Select the policy that you want to delete.
3. Click **Delete** from the drop-down box in the table header.
4. Click **Go**.
5. Click **OK** to confirm the deletion.

Related topics:

- “Policies” on page 397
- “Policies and rules” on page 44
- “rmpolicy” on page 244

Listing policies

Steps:

To display a list of all policies, click **Manage Filing** → **Policies** from the My Work frame.

Related topics:

- “Ispolicy” on page 199
- “Policies and rules” on page 44
- “Policies” on page 397

Changing the name of a policy

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

You cannot change the name of a policy. Instead, clone the existing policy, and then delete the original one.

Related topics:

- “Copying a policy” on page 119
- “Deleting a policy” on page 120
- “Changing the rules in a policy” on page 118
- “Policies” on page 397
- “Policies and rules” on page 44

Viewing policy details

Steps: Perform the following steps to display the details of a specific policy:

1. Click **Manage Filing** → **Policies** from the My Work frame.
2. Select a policy.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Details** tab.

Related topics:

- “Ispolicy” on page 199
- “Policies and rules” on page 44
- “Policy Properties - Details” on page 398

Viewing policy rules

Steps:

Perform the following steps to display the rules in a specific policy:

1. Click **Manage Filing** → **Policies** from the My Work frame.
2. Select a policy.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Rules** tab.

Related topics:

- “catpolicy” on page 156
- “Policies and rules” on page 44
- “Policy Properties - Rules” on page 399

Managing processes

You can configure and view SAN File System processes (also known as “threads.”)

Related topics:

- “Listing processes”
- “Viewing process details” on page 122
- “Viewing process limits” on page 122

Listing processes

Steps:

To display a list of all long-running processes, click **Monitor System** → **Processes** from the My Work frame.

Related topics:

- “Isproc” on page 205
- “Processes” on page 399
- “Viewing process details”
- “Viewing process limits”

Viewing process details

Steps:

Perform the following steps to display the details for a specific long-running process:

1. Click **Monitor System** → **Processes** from the My Work frame.
2. Select a process.
3. Click **General Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Details** tab.

Related topics:

- “Listing processes” on page 121
- “Isproc” on page 205
- “Processes Properties - Cluster-Level Details” on page 400
- “Processes Properties - Server-Level Details” on page 400
- “Viewing process limits”

Viewing process limits

Steps: Perform the following steps to display the limits for a specific long-running process:

1. Click **Monitor System** → **Processes** from the My Work frame.
2. Select a process.
3. Click **General Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Process Limits** tab.

Related topics:

- “Listing processes” on page 121
- “Isproc” on page 205
- “Viewing process details”

Managing storage pools

A storage pool is a collection of SAN File System volumes that can be used to store either metadata or file data. A storage pool typically contains a set of volumes that provide a desired quality of service for a specific use, such as to store all files for a particular application or a specific business division. An administrator must assign one or more volumes to a storage pool before it can be used.

You can perform the following storage pool tasks:

Related topics:

- “Creating a storage pool”
- “Deleting a storage pool” on page 124
- “Listing storage pools” on page 124
- “Listing volumes in a storage pool” on page 125
- “Changing storage pool settings”
- “Setting the default storage pool” on page 125
- “Viewing storage pool details” on page 125
- “Viewing storage pool statistics” on page 126

Changing storage pool settings

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to modify the general settings for a specific storage pool:

1. Click **Manage Storage** → **Pools** from the My Work frame.
2. Select a storage pool.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **General Settings** tab.
6. Edit the appropriate fields.
7. Click **OK** to save the new settings.

Related topics:

- “chpool” on page 161
- “Listing storage pools” on page 124
- “Storage pools” on page 37
- “Storage Pool Properties - General Settings” on page 427
- “Viewing storage pool details” on page 125

Creating a storage pool

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

SAN File System console provides a wizard to step you through the process of creating a storage pool. This wizard creates volumes and assigns the volumes to the storage pool.

To create a storage pool from the Administrative command-line interface, you must first create the volumes using the **mkvol** command, and then create the storage pool using the **mkpool** command.

Steps:

To start the wizard to create a storage pool, click **Manage Storage** → **Create a Pool** from the My Work frame. Then, click **Manage Storage** → **Pools** from the My Work frame to verify that the storage pool was created.

Related topics:

- “Create a Storage Pool - Add Volumes” on page 373
- “Create a Storage Pool - Introduction” on page 374
- “Create a Storage Pool - Set Properties” on page 374
- “Create a Storage Pool - Verify Settings” on page 374
- “mkvol” on page 227
- “mkpool” on page 225
- “Storage pools” on page 37

Deleting a storage pool

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to delete a storage pool:

1. Click **Manage Storage** → **Pools** from the My Work frame.
2. Select the storage pool that you want to delete.
3. Click **Delete** from the drop-down box in the table header.
4. Click **Go**.
5. Click **OK** to confirm the deletion.

Related topics:

- “Delete Storage Pools” on page 376
- “Storage pools” on page 37
- “rmpool” on page 245

Listing storage pools

Steps:

To display a list of all filesets, click **Manage Storage** → **Pools** from the My Work frame.

Related topics:

- “Storage Pools” on page 423
- “Storage pools” on page 37
- “lspool” on page 202
- “Listing available LUNs” on page 131

- “Listing volumes in a storage pool”
- “Changing storage pool settings” on page 123
- “Viewing storage pool details”

Listing volumes in a storage pool

Steps:

Perform the following steps to display a list of volumes that are assigned to a specific storage pool:

1. Click **Manage Storage** → **Pools** from the My Work frame.
2. Select a storage pool.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Current Volumes** tab.

Related topics:

- “Storage pools” on page 37
- “Storage Pool Properties - Current Volumes” on page 426
- “Listing available LUNs” on page 131
- “Listing storage pools” on page 124
- “lsvol” on page 213
- “Removing volumes from a storage pool” on page 133

Setting the default storage pool

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to set the default storage pool:

1. Click **Manage Storage** → **Pools** from the My Work frame.
2. Select the storage pool that you want to be the default storage pool.
3. Click **Set as Default** from the drop-down box in the table header.
4. Click **Go**.
5. Click **OK** to confirm the change.

Related topics:

- “Storage pools” on page 37
- “Storage Pools” on page 423
- “setdefaultpool” on page 251

Viewing storage pool details

Steps: Perform the following steps to display the details of a specific storage pool:

1. Click **Manage Storage** → **Pools** from the My Work frame.
2. Select a storage pool.

3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Details** tab.

Related topics:

- “Storage pools” on page 37
- “Storage Pool Properties - Details” on page 426
- “Listing storage pools” on page 124
- “lspool” on page 202
- “Listing available LUNs” on page 131
- “Listing volumes in a storage pool” on page 125
- “Changing storage pool settings” on page 123

Viewing storage pool statistics

Steps:

Perform the following steps to view statistics for storage pools:

1. Click **Monitor System** → **Statistics** from the My Work frame.
2. Select the **Storage Pools** tab.
3. Click **OK** to view the statistics.
4. Click **Close** to close the Statistics panel.

Related topics:

- “Creating a report” on page 57
- “lspool” on page 202
- “Statistics - Storage Pools” on page 419
- “Viewing statistics” on page 61

Managing users

SAN File System provides different levels of user access that you can assign to administrative tasks in your environment. User roles must be configured at an LDAP server.

Related topics:

- “Granting SAN File System user access and roles”
- “Listing user roles” on page 127
- “Listing users” on page 127
- “Timing out all user authorizations” on page 128

Granting SAN File System user access and roles

Prerequisites:

You must have access to your LDAP server to perform this task.

Context:

SAN File System authenticates users by taking their names (with role mappings) and associated passwords from an LDAP server. Therefore, you cannot grant anyone administrative access to SAN File System without first defining them on your LDAP server.

User name, user-to-role mapping and passwords must be entered at the LDAP server, employing the schemas of the LDAP server in your environment. Refer to your LDAP server documentation for more information.

Note: User roles and associated level of read, write, and execute access are defined as a function of your business requirements during the planning phase. Please refer to the *Planning, Installation and Configuration Guide* for additional information about LDAP configuration.

Related topics:

- “lsadmuser” on page 172
- “resetadmuser” on page 234
- “Managing users” on page 126
- “User roles” on page 46

Listing user roles

Steps:

To view a listing of user roles currently defined within SAN File System, click **Administer Access** → **Roles** from the My Work frame.

Related topics:

- “Listing users”
- “lsadmuser” on page 172
- “Managing users” on page 126
- “Users” on page 433
- “Roles” on page 405
- “User roles” on page 46

Listing users

Steps:

To view a listing of users with administrative access to SAN File System, along with their associated user roles, click **Administer Access** → **Users** from the My Work frame.

Related topics:

- “Listing user roles”
- “lsadmuser” on page 172
- “Roles” on page 405
- “Managing users” on page 126
- “Users” on page 433
- “User roles” on page 46

Timing out all user authorizations

Prerequisites: You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to time-out administrative user authorizations, thereby requiring all users to re-authenticate with the LDAP server:

1. Click **Administer Access** → **Users** from the My Work frame.
2. Click **Timeout All Authorizations** from the drop-down box in the table header.
3. Click **Go** to confirm the timeout action.

Related topics:

- “Administrative security” on page 39
- “resetadmuser” on page 234
- “Managing users” on page 126
- “Users” on page 433
- “User roles” on page 46

Managing volumes and LUNs

A LUN becomes a SAN File System volume when an administrator adds it to a storage pool. It is automatically assigned a system-generated label that identifies it as a SAN File System volume. An administrator must also give the volume a name that is unique within all the volumes used by a SAN File System cluster.

You can perform the following tasks for volumes and LUNs:

Related topics:

- “Activating a volume” on page 129
- “Adding volumes to a storage pool” on page 130
- “Listing available LUNs” on page 131
- “Listing files on a volume” on page 131
- “Listing LUNs” on page 131
- “Listing volumes” on page 131
- “Changing volume settings” on page 130
- “Removing volumes from a storage pool” on page 133
- “Suspending a volume” on page 133
- “Viewing available LUN details” on page 134
- “Viewing LUN details” on page 134
- “Viewing LUN statistics” on page 134
- “Viewing volume details” on page 135
- “Viewing volume settings” on page 135
- “Viewing volume statistics” on page 135

Activating a volume

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to activate a volume that has been suspended:

1. Click **Manage Storage** → **Volumes** from the My Work frame.
2. Select a volume.
3. Click **Activate** from the drop-down box in the table header.
4. Click **Go**.

Related topics:

- “activatevol” on page 148
- “Suspending a volume” on page 133
- “Volumes” on page 49
- “Volumes” on page 435

Adding LUNs to SAN File System

Prerequisites:

Context:

Refer to the *Planning, Installation and Configuration Guide* for LUN restrictions and limitations.

Steps:

Perform the following steps to add new LUNs to the SAN File System:

1. Assign the new LUNs to the appropriate zone so that it can be accessed by the Metadata servers and clients.
2. Stop and restart each Metadata server in the cluster.
3. For each client for AIX, type **stfsdisk** and press Enter to force the client discover the new volume.

Related topics:

- “Adding volumes to a storage pool” on page 130
- “Listing LUNs” on page 131
- “Starting a Metadata server” on page 112
- “stfsdisk” on page 297
- “Stopping a Metadata server” on page 114
- “Viewing LUN details” on page 134
- “Viewing LUN statistics” on page 134
- “Volumes” on page 49

Adding volumes to a storage pool

Prerequisites:

You must have Administrator privileges to perform this task.

Context:

SAN File System provides a wizard to step you through the process of adding volumes to a storage pool.

Steps:

To start the wizard to add a volume, click **Manage Storage** → **Add Volumes** from the My Work frame.

Related topics:

- “Adding LUNs to SAN File System” on page 129
- “Create a Storage Pool - Introduction” on page 374
- “Create a Storage Pool - Add Volumes” on page 373
- “Create a Storage Pool - Verify Settings” on page 374
- “Listing volumes in a storage pool” on page 125
- “mkvol” on page 227
- “Storage pools” on page 37
- “Volumes” on page 49

Changing volume settings

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to change the general settings for a specific volume:

1. Click **Manage Storage** → **Volumes** from the My Work frame.
2. Select a volume.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **General Settings** tab.
6. Edit the appropriate fields.
7. Click **OK** to save the new settings.

Related topics:

- “chvol” on page 163
- “Listing volumes” on page 131
- “Volumes” on page 49
- “Volume Properties - General Settings” on page 435
- “Viewing storage pool details” on page 125

Listing available LUNs

Steps:

To display a list of LUNs in the SAN that are available to add to storage pools, click **Manage Storage** → **Available LUNs** from the My Work frame.

Related topics:

- “Available LUNs” on page 350
- “Islun” on page 195
- “Listing volumes in a storage pool” on page 125
- “Volumes” on page 49

Listing files on a volume

Prerequisites:

You must have Administrator, Operator, or Backup privileges to perform this task.

Steps:

To display a list of files on a specific volume, run the **reportvolfiles** command from the Administrative CLI.

Related topics:

- “Volumes” on page 49
- “reportvolfiles” on page 232

Listing LUNs

Steps:

To display a list of all LUNs in the SAN, click **Manage Storage** → **LUNs** from the My Work frame.

Related topics:

- “LUNs” on page 396
- “Islun” on page 195
- “Listing volumes in a storage pool” on page 125
- “Volumes” on page 49

Listing volumes

Steps:

To display a list of all volumes, click **Manage Storage** → **Volumes** from the My Work frame.

Related topics:

- “Listing LUNs”
- “lsvol” on page 213

- “Removing volumes from a storage pool” on page 133
- “Volumes” on page 49
- “Volumes” on page 435

Managing free and allocated storage space

Prerequisites:

Context:

You can use the storage pool usage threshold and the fileset quota and usage threshold to manage the storage space in SAN File System.

The storage pool usage threshold is a percentage of the storage pool’s estimated storage capacity. When the amount of used space reaches or exceeds the specified threshold, the Metadata server generates an alert.

The fileset quota is the size limit, in megabytes for the fileset. When the size of the fileset reaches or exceeds the specified limit, the Metadata server generates an alert. You can specify either a hard or soft quota. A hard quota denies requests for additional space, and a soft quota grants requests space allocation.

Note: The quota must be set to a whole number.

The fileset usage threshold is a percentage of the fileset’s quota. When the amount of used space reaches or exceeds the specified threshold, the Metadata server generates an alert.

Steps:

Perform the following steps to configure the storage pool threshold:

1. Click **Manage Storage** → **Pools** from the My Work frame.
2. Select a storage pool.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **General Settings** tab.
6. Verify that the **Enable usage alerts** check box is selected.
7. Type a percentage in the **Usage Threshold** field.
8. Click **OK** to save the new settings.

Perform the following steps to configure the fileset quota:

1. Click **Manage Filing** → **Filesets** from the My Work frame.
2. Select a fileset.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click on the **General Settings** tab.
6. Specify the type of quote by selecting either a hard or soft quota in the **Quota Type** drop-down list.
7. Type a size limit in the **Quota Size** field.

Note: If a hard quota already exists, you must not set the new size to a value less than the current quota size for this fileset.

8. Verify that the **Enable usage alerts** check box is selected.
9. Type a percentage in the **Usage Threshold** field.
10. Click **OK** to save the new settings.

Related topics:

- “Changing fileset settings” on page 103
- “Changing storage pool settings” on page 123
- “chpool” on page 161
- “chcontainer” on page 159
- “Fileset Properties - General Settings” on page 392
- “Filesets” on page 24
- “Storage pools” on page 37
- “Storage Pool Properties - General Settings” on page 427
- “Viewing storage pool details” on page 125

Removing volumes from a storage pool

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to remove one or more volumes from a specific storage pool:

1. Click **Manage Storage** → **Volumes** from the My Work frame.
2. Select one or more volumes that you want to remove.
3. Click **Remove** from the drop-down box in the table header.
4. Click **Go**.
5. Select the removal method.
6. Click **OK**.

Related topics:

- “Adding volumes to a storage pool” on page 130
- “Storage pools” on page 37
- “Remove Volumes” on page 401
- “Listing volumes” on page 131
- “rmvol” on page 247
- “Volumes” on page 49

Suspending a volume

Prerequisites:

You must have Administrator privileges to perform this task.

Steps:

Perform the following steps to suspend an active volume so that the Metadata server cannot allocate new data on it:

1. Click **Manage Storage** → **Volumes** from the My Work frame.
2. Select a volume.
3. Click **Suspend** from the drop-down box in the table header.
4. Click **Go**.

Related topics:

- “Activating a volume” on page 129
- “suspendvol” on page 285
- “Volumes” on page 49
- “Volumes” on page 435

Viewing available LUN details

Steps:

Perform the following steps to display the details of a specific available LUN:

1. Click **Manage Storage** → **Available LUNs** from the My Work frame.
2. Select an available LUN.
3. Click **Details** from the drop-down box in the table header.
4. Click **Go**.

Related topics:

- “Islun” on page 195
- “Volumes” on page 49
- “Details of LUN” on page 379

Viewing LUN details

Steps: Perform the following steps to display the details of a specific LUN:

1. Click **Manage Storage** → **LUNs** from the My Work frame.
2. Select a LUN.
3. Click **Details** from the drop-down box in the table header.
4. Click **Go**.

Related topics:

- “Islun” on page 195
- “Volumes” on page 49
- “LUNs” on page 396

Viewing LUN statistics

Steps:

Perform the following steps to view statistics for LUNs:

1. Click **Monitor System** → **Statistics** from the My Work frame.
2. Select the **LUNs** tab.
3. Click **OK** to view the statistics.

4. Click **Close** to close the Statistics panel.

Related topics:

- “Creating a report” on page 57
- “Islun” on page 195
- “Statistics - LUNs” on page 417
- “Viewing statistics” on page 61

Viewing volume details

Steps: Perform the following steps to display the details for a specific volume:

1. Click **Manage Storage** → **Volumes** from the My Work frame.
2. Select a volume.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **Details** tab.

Related topics:

- “Isvol” on page 213
- “Listing volumes” on page 131
- “Viewing volume settings”
- “Volumes” on page 49
- “Volume Properties - Details” on page 435

Viewing volume settings

Steps: Perform the following steps to display the general settings for a specific volume:

1. Click **Manage Storage** → **Volumes** from the My Work frame.
2. Select a volume.
3. Click **Properties** from the drop-down box in the table header.
4. Click **Go**.
5. Click the **General Settings** tab.

Related topics:

- “Isvol” on page 213
- “Listing volumes” on page 131
- “Changing volume settings” on page 130
- “Viewing volume details”
- “Volumes” on page 49
- “Volume Properties - General Settings” on page 435

Viewing volume statistics

Steps:

Perform the following steps to view statistics for volumes:

1. Click **Monitor System** → **Statistics** from the My Work frame.

2. Click the **Volume** tab.
3. Click **OK** to view the statistics.
4. Click **Close** to close the Statistics panel.

Related topics:

- “Creating a report” on page 57
- “Statistics - Volumes” on page 421
- “Viewing statistics” on page 61

Appendix A. Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully.

Features:

These are the major accessibility features in SAN File System:

- You can use screen-reader software and a digital speech synthesizer to hear what is displayed on the screen.

Note: The SAN File System Information Center and its related publications are accessibility-enabled for the IBM Home Page Reader.

- You can operate all features using the keyboard instead of the mouse.

Navigating by keyboard:

You can use keys or key combinations to perform operations and initiate many menu actions that can also be done through mouse actions. You can navigate the SAN File System console and help system from the keyboard by using the following key combinations:

- To traverse to the next link, button or topic, press Tab inside a frame (page).
- To expand or collapse a tree node, press Right or Left arrows, respectively.
- To move to the next topic node, press Down arrow or Tab.
- To move to the previous topic node, press Up arrow or Shift+Tab.
- To scroll all the way up or down, press Home or End, respectively.
- To go back, press Alt+Left arrow
- To go forward, press Alt+Right arrow.
- To go to the next frame, press Ctrl+Tab. There are quite a number of frames in the help system.
- To move to previous frame, press Shift+Ctrl+Tab.
- To print the current page or active frame, press Ctrl+P.

Appendix B. Commands

SAN File System has two sets of commands: administrative and client commands.

Administrative commands:

The administrative commands run on the storage engines that host the Metadata server. Most commands must be run from the master Metadata server. There are a few commands that must be run from subordinate Metadata server for specific situations.

You run a majority of the administrative commands from the tanktool session to manage SAN File System. There are a few commands that must be run from the operating-system shell prompt.

To use the administrative commands, you must log in directly to the engine, or from another workstation through SSH, using the local operating system authentication mechanism. You must then log in to the Administrative server on the engine using the same administrative user ID and password that you would use to log into the SAN File System console. You can specify the password in one of two ways:

- Set the password in the sclif.properties file, located in your home directory on the engine (for example, joe/sclif.properties), to your valid LDAP password using the **tankpasswd** utility.
- Set the SFS_CLI_PASSWDFILE environment variable to the location of the password file.

Note: The Administrative commands are case sensitive. If you enter a command in uppercase, you will receive an error.

Client commands:

The client commands runs on any client machine on which the client file-system driver has been installed. It provides a set of commands that you can use to manage your clients.

To use the client commands, you must log in directly to the client machine or from another workstation using SSH. You log in using the user ID and password for the client machine. You must have administrative (Windows) or root (UNIX-based) privileges to use the client commands.

Related topics:

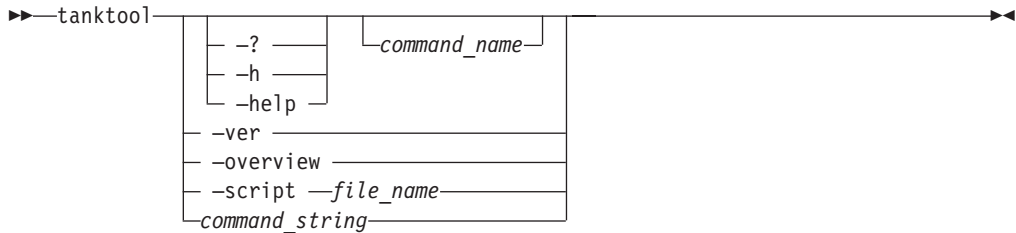
- “Accessing an engine through SSH” on page 94
- “Administrative commands” on page 142
- “Client commands” on page 288
- “Command modes” on page 311
- Appendix C, “Environment variables”, on page 319
- “Naming guidelines” on page 311
- “Standard format parameters” on page 312
- “Syntax diagram conventions” on page 315

- “tanktool”
- “User assistance for commands” on page 142

tanktool

Starts the Administrative command-line interface (CLI), or tanktool session, to run commands in interactive mode. This utility also runs a single command or runs a set of commands from a script without starting a tanktool session.

Syntax:



Parameters:

-? | -h | -help *command_name*

Displays help for the specified command (for example, **-h mkpool** displays help for the **mkpool** command). If a command name is not specified, this parameter displays a list of available commands in the Administrative CLI.

Note: Although the **tanktool** utility is listed as a possible command in the help facility, you will receive an error if you attempt to run the **tanktool** utility from within tanktool (for example, `tanktool> tanktool help`). To get help for the **tanktool** utility, enter **tanktool -help tanktool**.

-ver

Displays the current version and licensing information for this utility.

-overview

Displays the overview information about the tanktool utility, including command modes, standard command and listing parameters, syntax diagram conventions, and user assistance.

-script *file_name*

Runs the set of command strings in the specified file outside of a tanktool session. If you specify this parameter, you must specify a file name.

The format options specified using the **setoutput** command apply to all commands in the script.

Output from successful commands routes to stdout. Output from unsuccessful commands route to stderr. If an error occurs while one of the commands in the script is running, the script will exit at the point of failure and return to the system prompt.

command_string

Runs the specified command string outside of a tanktool session.

Description:

If you run this utility with any of the valid parameters, the Administrative CLI is not started in interactive mode. If you run this command with no parameters, this command starts the Administrative CLI. When you are in the Administrative CLI, the `tanktool>` prompt is displayed.

Examples:

Display the rules in a policy using single-shot mode The following example displays the rules for the DEFAULT policy and the policy named `activePolicy` from the `tanktool` utility using single-shot mode:

```
shell> tanktool catpolicy DEFAULT activePolicy
DEFAULT:
  VERSION 1
  rule 'stgRule1' set stgpool 'pool1' for fileset ('cnt_A')
  rule 'stgRule2' set stgpool 'pool2' where NAME like '%.doc'
  rule 'stgRule3' set stgpool 'pool3' where DAYOFWEEK(CREATION_DATE) == 1
  rule 'stgRule4' set stgpool 'pool4' where USER_ID <= 100
```

Related topics:

- “Command modes” on page 311
- Appendix C, “Environment variables”, on page 319
- “exit” on page 170
- “help” on page 171
- “quit” on page 231
- “setoutput” on page 255
- “Standard format parameters” on page 312
- “User assistance for commands” on page 142

tankpasswd

Generates a password file that enables you to log in to the Administrative command-line interface, also known as `tanktool`.

Syntax:

```
►►—tankpasswd— -u —user_name— -p —password—►►
```

Parameters:

- u** *user_name*
Specifies the user name.
- p** *password*
Specifies the password associated with the given user name.

Description:

This command must be run before you attempt to log on to the Administrative CLI for the first time and when your password changes.

This command creates a password file, named `tank.passwd`, in the current directory. By default, the `tanktool` utility expects this file to be in your home directory. You can change this location by modifying the `SFS_CLI_PASSWDFILE` environment variable.

Note: You must move the password file to the location expected by the **tanktool** utility.

The user ID and password that you specify in this command must be the same user ID and password that is specified in the LDAP server by your system administrator. The LDAP server is used to connect to the Administrative agent, which authorizes or denies access to SAN File System each time you attempt to use the **tanktool** utility.

Examples:

Generate the password file The following example generates a password file for user saki:

```
$ pwd
/home/saki
$ tankpasswd -u saki -p mypassword
The password file was successfully written to: /home/saki/.tank.passwd
```

Related topics:

- “tanktool” on page 140
- Appendix C, “Environment variables”, on page 319

User assistance for commands

You can get user assistance for the commands that are available in the Administrative CLI using the **help** command or the command-help parameters.

The **help** command allows you to display the following types of user assistance:

- A list of available commands
- A list of available commands with the syntax diagram for each
- A list of available commands with the a brief description for each
- A detailed description of a specific command

You can use the command-help parameters supported by each command to display a detailed description of the specified command. The following help parameters are supported:

- -?
- -h
- -help

Note: When you use a help parameter, all other parameters are ignored.

Related topics:

- “help” on page 171

Administrative commands

The following table provides a brief description and administrative role for each command in the Administrative command-line interface.

Command	Description	Roles
General		

Command	Description	Roles
exit	Ends a tanktool session.	Monitor, Backup, Operator, Administrator
help	Displays a list of commands available in the Administrative command-line interface and optionally displays the syntax or brief description of each command.	Monitor, Backup, Operator, Administrator
quit	Ends the tanktool session.	Monitor, Backup, Operator, Administrator
setoutput	Sets the output format for the Administrative command-line interface.	Monitor, Backup, Operator, Administrator
Alerts and logs		
addsnmpmgr	Adds an SNMP manager to receive SNMP traps. To generate traps, you must add an SNMP manager and set the SNMP traps to be generated (using the settrap command).	Administrator
catlog	Displays the contents of the various log files maintained by the Administrative server and the cluster.	Monitor, Backup, Operator, Administrator
clearlog	Clears the audit log and cluster log files. Because the event log is a subset of the cluster log, the event log is also cleared when you clear the cluster log.	Administrator
lssnmpmgr	Displays a list of SNMP managers and their attributes.	Monitor, Backup, Operator, Administrator
lstrapsetting	Displays a list of event types that currently generate an SNMP trap.	Monitor, Backup, Operator, Administrator
rmsnmpmgr	Removes an SNMP manager (recipient).	Administrator
settrap	Specifies whether an SNMP trap is generated and sent to all SNMP managers when a specific type of event occurs on the Metadata server.	Administrator
Clients		
lsclient	Displays a list of clients that are currently being served by one or more Metadata servers in the cluster. You must be logged on to the master Metadata server to use this command.	Monitor, Backup, Operator, Administrator
Cluster		
chclusterconfig	Modifies the cluster settings that do not require a restart when changed.	Administrator
quiescecluster	Changes the state of all Metadata servers in the cluster to one of three quiescent states.	Operator, Administrator
resumecluster	Brings all Metadata servers in the cluster to the online state.	Operator, Administrator
startcluster	Starts all Metadata servers in the cluster and brings them to the full online state.	Operator, Administrator

Command	Description	Roles
statcluster	Displays status, network, workload, and configuration information about the cluster.	Monitor, Backup, Operator, Administrator
stopcluster	Stops all Metadata servers in the cluster gracefully.	Operator, Administrator
upgradeclasser	Verifies that all the engines in the cluster have been upgraded to the new version of the software and then initiates the cluster upgrade process.	Administrator
Engines		
collectdiag	Collects data on all SAN File System components running in the system and saves the diagnostic results to various files on one or more specified engines.	Operator, Administrator
lsengine	Displays a list of storage engines and their attributes.	Monitor, Backup, Operator, Administrator
restartengine	Reboots one or more storage engines.	Operator, Administrator
startengine	Starts the CPU and operating system on one or more storage engines.	Operator, Administrator
statengine	Displays status information about a specific storage engine.	Monitor, Backup, Operator, Administrator
stopengine	Shuts down the operating system and powers off one or more storage engines.	Operator, Administrator
Filesets (containers)		
attachcontainer	Attaches an existing detached fileset (container) to a specific point in the global namespace or moves an already attached fileset to a new attach location.	Administrator
chcontainer	Modifies the settings of one or more filesets (containers).	Administrator
detachcontainer	Detaches one or more filesets (containers) from the global namespace.	Administrator
lscontainer	Displays a list of filesets (containers) and their attributes.	Monitor, Backup, Operator, Administrator
mkcontainer	Creates a new fileset (container).	Administrator
rmcontainer	Removes one or more empty, detached filesets (containers) and optionally the files in the filesets, including any FlashCopy images.	Administrator
setcontainerserver	Reassigns an existing fileset (container) to be hosted by a different Metadata server.	Administrator
statcontainer	Displays the number of started and completed transactions for the filesets (containers) being served by the local Metadata server.	—
FlashCopy images		

Command	Description	Roles
lsimage	Displays a list of FlashCopy images in all filesets or a specified fileset.	Monitor, Backup, Operator, Administrator
mkimage	Creates a near-instantaneous FlashCopy image of the file layout and contents of the specified fileset and stores it in that fileset.	Backup, Operator, Administrator
reverttoimage	Reverts the current fileset to a specified FlashCopy image of the file layout and contents.	Administrator
rmimage	Deletes one or more FlashCopy images for a specific fileset.	Backup, Operator, Administrator
Metadata		
builddrscript	Converts the system-metadata disaster-recovery file, created by the mkdrfile command, into a set of recovery scripts. You would then run these scripts to recreate the system metadata in the event of a disaster.	Backup, Operator, Administrator
lsdrfile	Displays a list of system-metadata disaster-recovery files.	Monitor, Backup, Operator, Administrator
mkdrfile	Creates a new system-metadata disaster-recovery dump file	Backup, Operator, Administrator
rmdrfile	Deletes an existing system-metadata disaster-recovery dump file.	Backup, Operator, Administrator
startmetadatabackup	Starts the utility that performs a consistency check on the metadata for the entire system or a set of filesets (containers), generates reports in the cluster log, and optionally repairs inconsistencies in the metadata.	Administrator
stopmetadatabackup	Stops the metadata check utility that is currently in progress.	Administrator
Metadata servers		
lsautorestart	Displays a list of Metadata servers and the automatic-restart settings for each.	Monitor, Backup, Operator, Administrator
lsserver	Displays a lists of all Metadata servers in the cluster and their attributes if issued from the master Metadata server, or displays attributes about the local Metadata server if issued from a subordinate Metadata server.	Monitor, Backup, Operator, Administrator
setmaster	Sets a subordinate Metadata server as the new master Metadata server in the event of an irrecoverable loss of the current master Metadata server.	Administrator
startautorestart	Enables the Metadata server to restart automatically if it is down.	Administrator
startserver	Starts the specified Metadata server.	Operator, Administrator

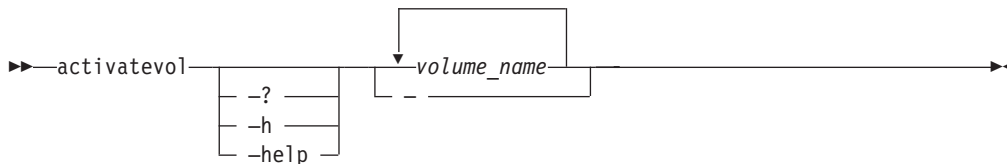
Command	Description	Roles
statserver	Displays status, configuration, and workload information for a specific Metadata server in the cluster, if issued from the master Metadata server. Lists status, configuration, and workload information for the local Metadata server, if issued from a subordinate Metadata server.	Monitor, Backup, Operator, Administrator
stopautorestart	Disables the Metadata server from restart automatically if it is down.	Administrator
stopsserver	Shuts down a subordinate Metadata server gracefully.	Operator, Administrator
Policies		
catpolicy	Displays the rules of the specified policies.	Monitor, Backup, Operator, Administrator
lspolicy	Lists the active and inactive policies.	Monitor, Backup, Operator, Administrator
mkpolicy	Creates a policy.	Administrator
rmpolicy	Deletes one or more inactive policies.	Administrator
usepolicy	Directs the Metadata server to make an existing policy the active policy and applies its rules to all subsequent file creations.	Administrator
Processes		
lsproc	Displays a list of long-running processes that are not yet complete and their attributes.	Monitor, Backup, Operator, Administrator
Storage pools		
chpool	Modifies the settings of one or more storage pools.	Administrator
lspool	Displays a list of the existing storage pools and their attributes.	Monitor, Backup, Operator, Administrator
mkpool	Defines one or more new user storage pool.	Administrator
rmpool	Deletes one or more empty, unreferenced storage pools.	Administrator
setdefaultpool	Changes a user storage pool to the default storage pool, and changes the previous default storage pool to a regular, nondefault user storage pool.	Administrator
Users		
lsadmuser	Displays a list of administrative users and their attributes.	Monitor, Backup, Operator, Administrator
resetadmuser	Forces all administrative users to log in again.	Administrator
Volumes and LUNs		
activatevol	Activates a suspended volume so that the Metadata server can allocate new data on the volume.	Administrator

Command	Description	Roles
chvol	Modifies the settings of one or more volumes.	Administrator
lslun	Lists the LUNs that are accessible from SAN File System.	Monitor, Backup, Operator, Administrator
lsvol	Displays a list of available volumes.	Monitor, Backup, Operator, Administrator
mkvol	Adds one or more volumes to a storage pool.	Administrator
reportvolfiles	Displays a list of files (and their attributes) in the specified volume.	Backup, Operator, Administrator
rmvol	Removes one or more volumes from a storage pool and redistributes the contents to other volumes in the same storage pool.	Administrator
suspendvol	Suspends one or more volumes so that the Metadata server cannot allocate new data on the volumes.	Administrator

activatevol

Activates a suspended volume so that the Metadata server can allocate new data on the volume.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

volume_name

Specifies the names of the one or more volumes to activate.

- Specifies that you want to read the names of one or more volumes to activate from stdin (for example, - << /work/vol_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine that hosts the master Metadata server to run this command.

Examples

Activate two volumes The following example activates two volumes: vol1 and vol2.

```
tanktool> activatevol vol1 vol2
Volume vol1 activated.
Volume vol2 activated.
```

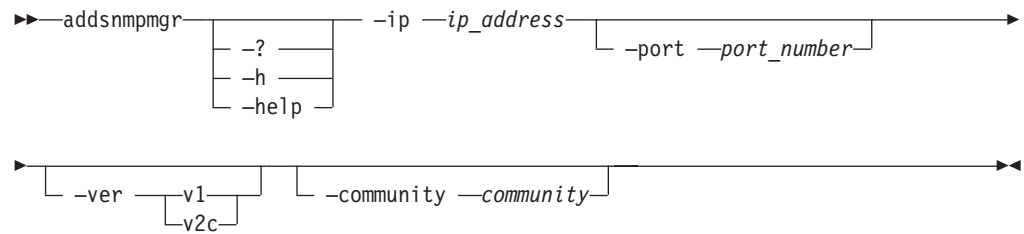
Related topics

- “Activating a volume” on page 129
- “chvol” on page 163
- “lsvol” on page 213
- “mkvol” on page 227
- “reportvolfiles” on page 232
- “Standard format parameters” on page 312
- “suspendvol” on page 285
- “Volumes” on page 49

addsnmpmgr

Adds an SNMP manager to receive SNMP traps.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-ip *ip_address*

Specifies the IP address of the SNMP trap receiver.

-port *port_number*

Specifies the SNMP-trap port number. The default port number is 162.

-ver *v1 | v2c*

Specifies the SNMP version. You can specify one of the following values:

v1 Sends SNMP version 1 traps. This is the default value.

v2c Sends SNMP version 2 notifications using community-based security.

-community *community*

Specifies the community as any alphanumeric string used to authenticate v1 or v2c traps. The default community is "public."

Commas and colons are not valid characters in the community name. This name can be no longer than 128 characters in length.

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Note: To generate traps, you must add an SNMP manager and set the SNMP traps to be generated (using the **settrap** command).

If an SNMP manager is not added, then traps are not generated.

You can add up to two SNMP managers to SAN File System. One of these SNMP managers may be used by the optional Call Home feature.

When you change a disruptive cluster setting, dynamic cluster settings (such as SNMP settings) cannot be modified until you reboot the cluster.

Examples

Adds an SNMP manager The following example adds an SNMP manager using SNMP v2c.

```
tanktool> addsnmpmgr -ip 192.168.0.1 -port 8192 -ver v2c -community SNMPMgr1  
SNMP manager 192.168.0.1 successfully added.
```

Related topics

- “Adding SNMP managers” on page 67
- “Service Alert” on page 9
- “catlog” on page 154
- “clearlog” on page 165
- “lssnmpmgr” on page 211
- “lstrapsetting” on page 212
- “rmsnmpmgr” on page 246
- “settrap” on page 257
- “SNMP” on page 42
- “Standard format parameters” on page 312

attachcontainer

Attaches an existing detached fileset (container) to a specific point in the global namespace or moves an already attached fileset to a new attach location.

Syntax

```
▶▶ attachcontainer -attach location -dir directory_name ▶▶
    -?
    -h
    -help
▶ container_name ▶▶
```

Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-attach *location*

Specifies the directory path at which to attach the fileset. The root fileset in the global namespace must be included in the directory path. For example, to attach the fileset to the /home location, specify `-attach sanfs/home`. To attach the fileset directly to the root of the global namespace, use `-attach sanfs`.

Use only forward slashes (/) in the directory path for delimiters.

This directory *must* already exist.

Note: The name of the root fileset is the same as the name of the cluster. This name is "sanfs."

-dir *directory_name*

Specifies the directory name for the root of the fileset directory tree. The directory name *must not* already exist. This directory can be up to 256 characters in length.

The directory must not contain backslash (\) or forward slash (/) characters as delimiters.

The attach point (*location/directory_name*) appears as a drive and directory on clients for Windows or as a file system for client based on UNIX. It is recommended that you map the name of the directory to its associated fileset by using the same name as the fileset plus an additional suffix to distinguish it as the directory and not the fileset. For example, if you want to attach the fileset named work to the root of the global namespace, you would specify "sanfs" for the *location* and specify "work_ap" for the *directory_name*. On a client for Windows, the "work_ap" directory would appear under the "sanfs" drive.

container_name

Specifies the name of the fileset (or container) to attach.

- Specifies that you want to read the name of the fileset to attach from stdin (for example, `- << /work/cnt_list.txt`).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

You make a fileset part of the global namespace by attaching it to a specific location. To a client, this attach point looks like a regular directory.

Examples

Attaching a fileset The following example attaches fileset cntA to the sanfs/homes location in the global namespace using cntA_ap as its directory name.

```
tanktool> attachcontainer -attach sanfs/homes -dir cntA_ap cntA  
Container cntA attached to directory sanfs/homes/cntA_ap.
```

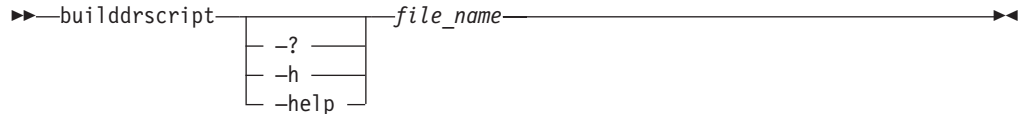
Related topics

- “Attaching a fileset” on page 103
- “chcontainer” on page 159
- “Filesets” on page 24
- “detachcontainer” on page 168
- “lscontainer” on page 183
- “mkcontainer” on page 217
- “rmcontainer” on page 239
- “Standard format parameters” on page 312

builddrscript

Converts the system-metadata disaster-recovery file into a set of recovery scripts used to recreate the system metadata in the event of a disaster.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

file_name

Specifies the name of an existing system-metadata disaster-recovery file.

Description

Prerequisites:: You must have Backup, Operator, or Administrator privileges to use the command.

Note: This command operates only against the local engine.

This command is used in conjunction with the **mkdrfile** command to create a set of recovery scripts that are used to recreate the system metadata in the event of a disaster. The **builddrscript** command is run against the system-metadata disaster-recovery file created by the **mkdrfile** command.

Note: The output for this command is written to the `/usr/tank/server/DR` directory. This command will overwrite any files that were created by a previous run of this command. If you want to preserve the existing files, copy them to another directory.

You would run this command against the system-metadata disaster-recovery file only if you are intending to recreate the system metadata.

After this command generates the recovery scripts, you must edit the scripts before running them.

Examples

Builds the recovery scripts The following example builds the disaster recovery scripts from the system-metadata disaster-recovery file named `dr1`.

```
tanktool> builddrscript dr1
```

Disaster recovery script files for "dr1" were built successfully.

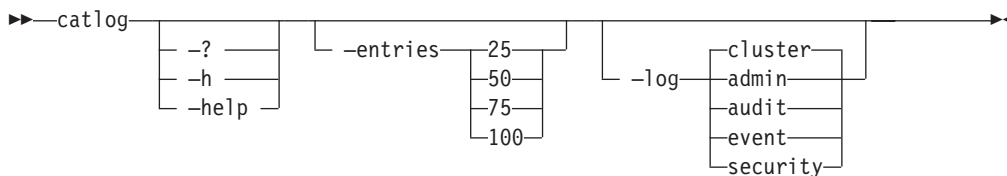
Related topics

- “Backup and restore” on page 5
- “mkdrfile” on page 220
- “lsdrfile” on page 187
- “rmdrfile” on page 241
- “Standard format parameters” on page 312

catlog

Displays the contents of the various log files maintained by the Administrative server and the cluster.

Syntax



Parameters

`-?` | `-h` | `-help`

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

`-entries`

Specifies the number of log entries to show at a time, from oldest to newest. Valid values are 25, 50, 75, or 100. If not specified, this command shows the entire log.

`-log`

Displays entries in the specified log, ordered by timestamp starting with the most recent entry. The default is cluster.

admin Displays entries in the administrative log, which maintains a history of messages created by the Administrative server.

audit Displays entries in the audit log, which maintains a history of all commands issued by any administrator for all Metadata servers in the cluster.

cluster

Displays entries in the cluster log, which maintains a history of messages created by all Metadata servers in the cluster.

event Displays entries in the event log, which maintains a history of event messages issued by all Metadata servers in the cluster.

security

Displays entries in the security log, which maintains a history of administrative-user login activity.

Description

Note: If you run this command from an engine hosting a subordinate Metadata server, logs for only the local engine are displayed. If you run this command from the engine hosting the master Metadata server, logs for the entire cluster are displayed.

If there are log entries that have not been displayed, you are prompted to press Enter to display the next set of entries or to type `exit` and press Enter to stop.

This command displays the following information for the specified log:

- Message code.

- Severity level (Severe, Error, Warning, Info).
- Message type (Normal or Event).
- Name of the Metadata server.
- Date and time the message was generated.
- Message.

Examples

Display the cluster log The following example displays the contents of the cluster log.

```
tanktool> catlog -log event
```

ID	Level	Server	Date and Time
TANCM0383I	Info	ST1	Feb 20, 2003 8:39:15 PM
TANCM0384I	Info	ST2	Feb 20, 2003 8:40:46 PM
TANCM0384I	Info	ST2	Feb 21, 2003 1:05:27 AM
TANCM0383I	Info	ST1	Feb 21, 2003 1:05:18 AM

Message

```
=====
ALERT: The cluster's state has changed to Online(10)
ALERT: The server's state has changed to NotRunning(0)
ALERT: The server's state has changed to Online(10)
ALERT: The cluster's state has changed to Online(10)
```

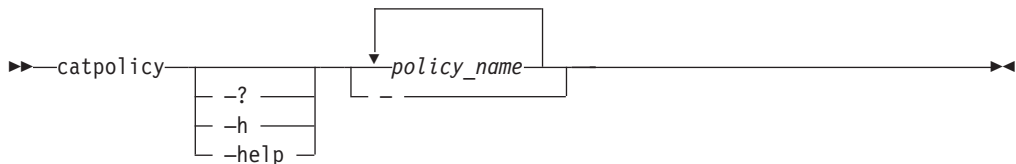
Related topics

- “Administrative log” on page 327
- “Audit log” on page 328
- “clearlog” on page 165
- “Server log” on page 330
- “Event log” on page 329
- “Logs” on page 30
- “Security log” on page 330
- “Standard format parameters” on page 312
- “Viewing logs” on page 71

catpolicy

Displays the rules of the specified policies.

Syntax



Description

Prerequisite: You must be logged in to the engine hosting the master Metadata server to run this command.

Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

policy_name

Specifies the names of one or more policies to display.

- Specifies that you want to read the names of one or more policies to display from stdin (for example, - << /work/policies_list.txt).

Examples

Display the rules in a policy The following example displays the rules for the DEFAULT and the ActivePolicy policies:

```
tanktool> catpolicy DEFAULT activePolicy
DEFAULT:
  VERSION 1
  rule 'stgRule1' set stgpool 'pool1' for CONTAINER ('cnt_A')
  rule 'stgRule2' set stgpool 'pool2' where NAME like '%.doc'
  rule 'stgRule3' set stgpool 'pool3' where DAYOFWEEK(CREATION_DATE) == 1
  rule 'stgRule4' set stgpool 'pool4' where USER_ID <= 100
```

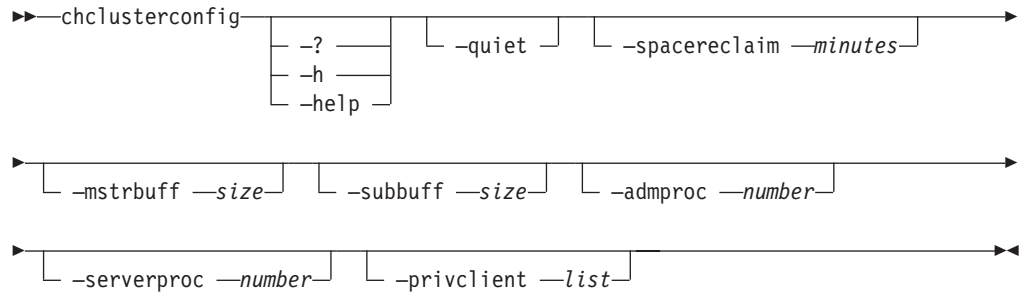
Related topics

- “lspolicy” on page 199
- “Policies and rules” on page 44
- “mkpolicy” on page 223
- “rmpolicy” on page 244
- “usepolicy” on page 287
- “Viewing policy rules” on page 121
- “Standard format parameters” on page 312

chclusterconfig

Modifies the cluster settings that do not require a restart when changed.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the change operation.

-spacereclaim *minutes*

Specifies the storage pool-space-reclamation interval, in minutes. You can specify a value from 0 to 1 440. The default is 60. A value of 0 disables storage pool-space reclamation.

-mstrbuff *size*

Specifies the size, in 4-KB pages, of the master Metadata server buffer. You can specify a value from 2 048 and 8 192. The default is 2 048.

Note: The buffer size can only be increased; it cannot be decreased.

Note: This parameter is intended for use only by trained service personnel.

-subbuff *size*

Specifies the size, in 4-KB pages, of the subordinate Metadata server buffer. You can specify a value from 30 000 and 250 000. The default is 200 000.

Note: The buffer size can only be increased; it cannot be decreased.

Note: This parameter is intended for use only by trained service personnel.

-admproc *number*

Specifies the maximum limit for administrative processes, including the number of Metadata server commands (from the administrative CLI or SAN File System console), that can run simultaneously. This limit is independent of the Server Workload Process Limit. A few commands, such as those that change the state of the cluster, can run on a temporary process even if the limit is currently exceeded. You can specify a value from 4 to 10. If not specified, the default value is 4.

Note: This limit can only be increased; it cannot be decreased.

Note: This parameter is intended for use only by trained service personnel.

-serverproc *number*

Specifies the maximum limit for the Metadata server workload process, which involves only Metadata server workload operations. You can specify a value from 10 to 50. The default value is 20.

Note: This limit can only be increased; it cannot be decreased.

Note: This parameter is intended for use only by trained service personnel.

-privclient *list*

Specifies a list of clients, separated by commas, that are to have root-level privileges (for example, `-privclient saki,mefi,leki`). The current list is replaced with the new list of clients. You can specify from zero to 30 clients; each client name can be no more than 256 characters in length.

Attention: This new list replaces the entire list of current privileged clients. If there are clients that have root or administrative privileges and you want to add one more client, you must specify both the current and new clients in the new list.

To remove all clients with root or administrative privileges, specify `-privclient ""`; however, at least one privileged client is required to create files and directories under the root directory of filesets.

This command does not verify that the specified client names are valid because the Metadata server does not keep static information about the clients. It does not keep a list of allowed or possible clients. It has only a list of clients that it allows privileged access if they present themselves. This command lists the clients that are currently accessing data.

Examples

Changes the cluster settings The following example changes space-reclamation setting for the cluster:

```
tanktool> chclusterconfig -spacereclaim 100  
Are you sure you want to change cluster configuration settings? [y/n] y  
Cluster successfully modified.
```

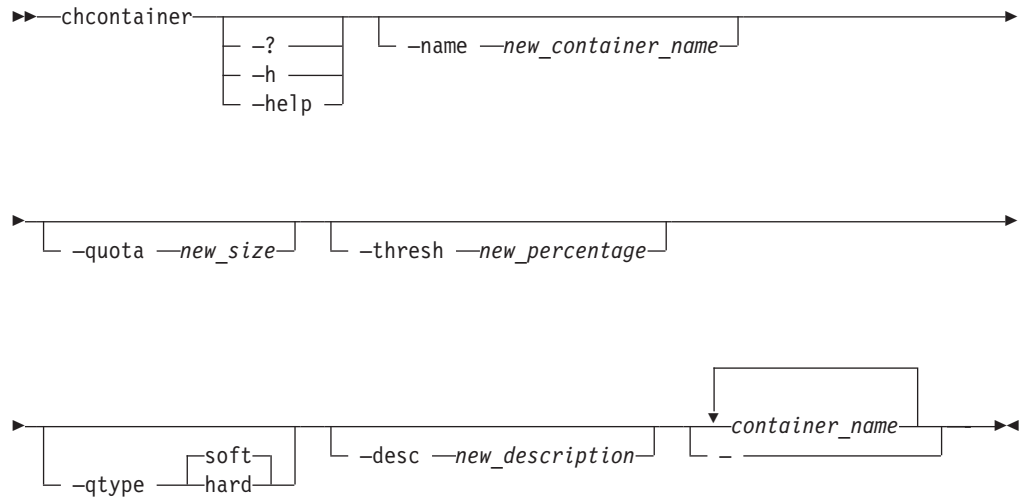
Related topics

- “Cluster” on page 16
- “Changing active cluster states” on page 82
- “Standard format parameters” on page 312
- “startcluster” on page 261
- “statcluster” on page 267
- “stopcluster” on page 280
- “upgradeclass” on page 286

chcontainer

Modifies the settings of one or more filesets (containers).

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-name *new_container_name*

Specifies the new name to assign to the fileset (or container). This name can be up to 256 characters in length.

-quota *new_size*

Specifies the new maximum size limit, in megabytes, for the specified fileset that, when exceeded, causes the Metadata server to generate an alert. You can specify a number from 0 to 1 073 741 824 MB (1 PB). If set to 0, there is no quota size limit for this fileset, and alerts are not sent.

Note: If a hard quota already exists, you must not set the new size to a value less than the current quota size for this fileset.

-thresh *new_percentage*

Specifies the new maximum percentage (alert threshold) of the specified quota size for this fileset that, when exceeded, causes the Metadata server to generate an alert. You can specify a value between 0 and 100. If set to 0, no alerts are generated.

-qtype **hard** | **soft**

Specifies the quota type for the fileset. You can specify one of the following values:

hard A hard quota produces a log message and potential alert when the quota is met, and denies requests for additional space.

soft A soft quota produces a log message and potential alert when the quota size is exceeded, but grants requests space allocation.

-desc *description*

Specifies a new description for the fileset. The description must be enclosed in matching single (') or double (") quotation marks if it contains any blank characters. This description can be up to 256 characters in length.

The default is an empty string.

container_name

Specifies the names of one or more filesets (or containers) to modify.

Note: If you specify a name using the **-name** option, you may specify one only fileset.

- Specifies that you want to read the names of one or more filesets to modify from stdin (for example, - << /work/cnt_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

You cannot modify the name for multiple filesets at one time.

Examples

Modify parameters of two filesets The following example changes the quota and threshold values for two filesets (*cnt_A* and *cnt_B*). It sets the quota to 1 000 MB and specifies to send an alert when the quota reaches 70%.

```
tanktool> chcontainer -quota 1000 -thresh 70 -qtype soft cnt_A cnt_B
Container cnt_A modified.
Container cnt_B modified.
```

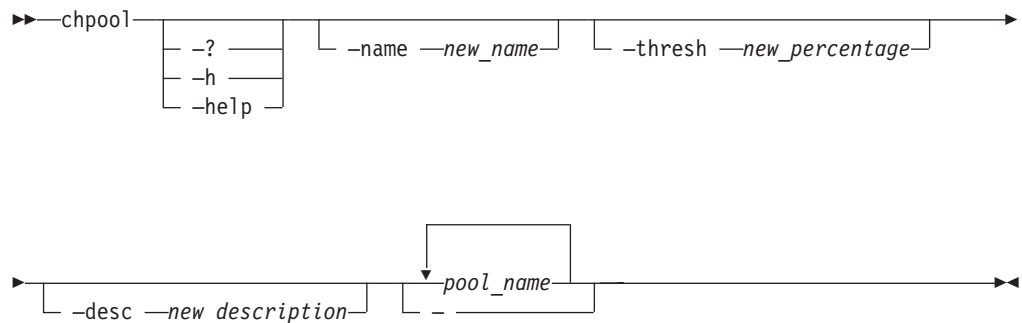
Related topics

- “attachcontainer” on page 151
- “Filesets” on page 24
- “detachcontainer” on page 168
- “lscontainer” on page 183
- “Changing fileset settings” on page 103
- “mkcontainer” on page 217
- “rmcontainer” on page 239
- “Standard format parameters” on page 312

chpool

Modifies the settings of one or more storage pools.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-name *new_name*

Specifies the new name to assign to the storage pool. This name can be up to 256 characters in length.

Note: You cannot change the name of the SYSTEM storage pool.

-thresh *new_percentage*

Specifies a percentage of the storage pool's estimated capacity that, when reached or exceeded, causes the Metadata server to generate an alert. If the capacity drops and then reaches the percentage again, additional alerts are sent. You can specify a value between 0 and 100. If set to 0, no alert is generated.

-desc *new_description*

Specifies a description for the storage pool. The description must be enclosed in matching single (') or double (") quotation marks if it contains any blank characters. This description can be up to 256 characters in length.

The default is an empty string.

Note: You cannot change the description of the SYSTEM storage pool.

pool_name

Specifies the names of one or more storage pools to modify.

Note: If you specify a name using the **-name** option, you may specify only one storage pool.

- Specifies that you want to read the names of one or more storage pools to modify from stdin (for example, - << /work/stgpools.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.

2. You must be logged in to the engine hosting the master Metadata server to run this command.

You cannot modify the names of multiple storage pools at one time.

Examples

Modify settings for two storage pools The following example modifies the threshold setting for two storage pools: stgpool1 and stgpool2.

```
tanktool> chpool -thresh 70 stgpool1 stgpool2  
Storage pool stgpool1 modified.  
Storage pool stgpool2 modified.
```

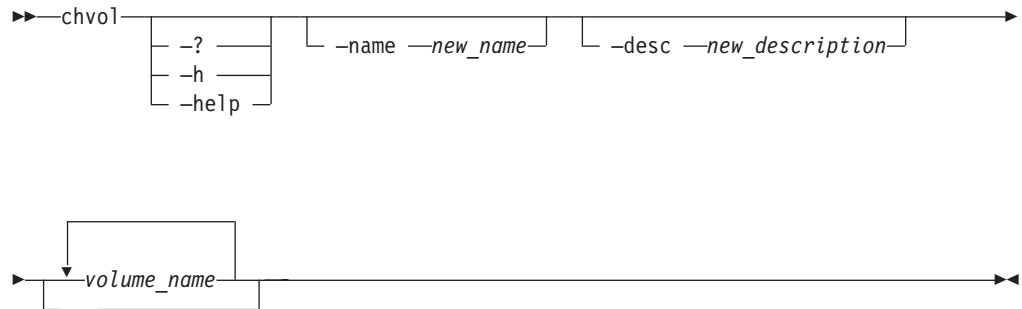
Related topics

- “lspool” on page 202
- “Changing storage pool settings” on page 123
- “mkpool” on page 225
- “Storage pools” on page 37
- “rmpool” on page 245
- “setdefaultpool” on page 251
- “Standard format parameters” on page 312

chvol

Modifies the settings of one or more volumes.

Syntax



Parameters

-? | **-h** | **-help**

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-name *new_name*

Specifies the new name to assign to the volume. The volume name must be unique within a cluster and can be up to 256 characters in length.

-desc *new_description*

Provides a new description of the volume. The description must be enclosed in matching single (') or double (") quotation marks if it contains any blank characters and can be up to 256 characters in length.

The default is an empty string.

volume_name

Specifies the name of one or more volumes to modify.

Note: If you specify a name using the **-name** option, you may specify one only volume.

- Specifies that you want to read the names of one or more volumes to modify from stdin (for example, - << /work/vol_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

You cannot modify the name of multiple volumes at one time.

Examples

Modify settings for multiple volumes The following example modifies the description of all the volumes listed in the file */tmp/vol_list.txt* to be "Engineering Volumes":

```
tanktool> chvol -desc "My SAN File System volumes" - << /tmp/vol_list.txt
```

```
Volume vol1 modified.  
Volume vol2 modified.  
Volume vol3 modified.
```

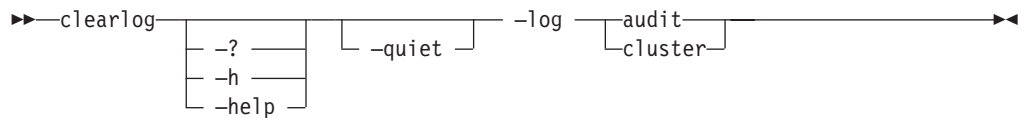
Related topics

- “activatevol” on page 148
- “lsvol” on page 213
- “Changing volume settings” on page 130
- “mkvol” on page 227
- “reportvolfiles” on page 232
- “rmvol” on page 247
- “suspendvol” on page 285
- “Volumes” on page 49
- “Standard format parameters” on page 312

clearlog

Clears the audit log and cluster log files.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the clear operation

-log

Specifies the log file to clear.

audit Clears all entries in the audit log, which maintains a history of messages created by the Administrative server.

cluster

Clears all entries in the cluster log, which maintains a history of all messages issued by any administrator for all Metadata servers in the cluster.

Description

Prerequisites: You must have Administrator privileges to use the command.

Notes:

1. If you run this command from an engine hosting a subordinate Metadata server, the audit log and cluster log on only the local engine are cleared. If you run this command from the engine hosting the master Metadata server, the audit logs and cluster logs on all engines are cleared.
2. Because the event log is a subset of the cluster log, the event log is also cleared when you clear the cluster log.

Examples

Clear the cluster log The following example clears the cluster log entries.

```
tanktool> clearlog -log cluster
```

```
Are you sure you want to remove all entries in the cluster log? [y/n] y  
Cluster log cleared.
```

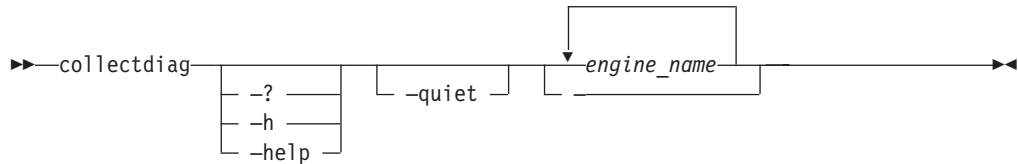
Related topics

- “Clearing logs” on page 68
- “Audit log” on page 328
- “catlog” on page 154
- “Server log” on page 330
- “Logs” on page 30
- “Standard format parameters” on page 312

collectdiag

Collects data on all SAN File System components running in the system and saves the diagnostic results to various files on the local file system of each specified engine. This command is used to assist in problem determination.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm this operation

engine_name

Specifies the names of one or more engines for which to collect diagnostic data.

Note: You must be logged on to the master Metadata server to issue this command against one or more non-local engine. If you are not logged on to the master Metadata server, this command collects diagnostics data only for the local engine.

- Specifies that you want to read the name of the engine for which to gather diagnostic data from stdin (for example, - << /work/vol_list.txt).

Description

Prerequisites: You must have Operator or Administrator privileges to use the command.

Diagnostic data is saved on each specified engine in a subdirectory of the directory specified by the TANKDIR environment variable. The default directory is /usr/tank/pmf. The subdirectory name is the timestamp when the data is collected (for example, ./usr/tank/pmf/20030507_081010).

Stdout and stderr output that is created by this command is saved in the /tmp/odbcoutfile on the local disk of each specified engine.

You must be logged on to the master Metadata server to issue this command against a non-local engine. Otherwise, this command works only on the local engine.

Examples

Collect diagnostic data The following example collects diagnostic data for engine ST1:

```
tanktool> collectdiag ST1  
Are you sure you want to collect diagnostic data for ST1 since this  
procedure may take a few minutes to complete? [y/n] y  
Diagnostic file successfully created for Engine ST1. Tip: See  
usr/tank/pmf to read the diagnostic files.
```

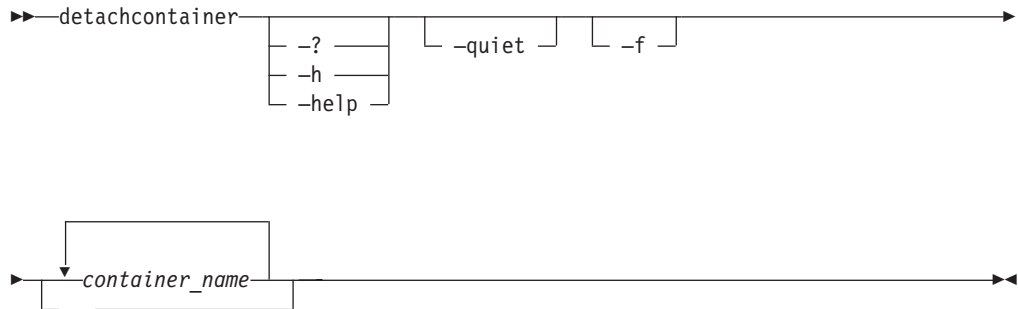
Related topics

- “Engines” on page 23

detachcontainer

Detaches one or more filesets (containers) from the global namespace.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the detach operation.

-f Forces the Metadata server to continue the detach operation even if there are open files in the fileset and suppresses any warning messages.

container_name

Specifies the names of one or more filesets (or containers) to detach.

- Specifies that you want to read the names of one or more filesets to detach from stdin (for example, **- << /work/cnt_list.txt**).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

The specified fileset **must not** have nested filesets. You must detach any nested filesets before detaching the fileset.

Unless you specify the **-f** parameter, there must not be any open files or any locks on files in the fileset.

The directory to which the fileset was attached is deleted after this operation is complete.

Examples

Detaching a fileset The following example detaches the fileset named *cnt_A*.

```
tanhktool> detachcontainer cnt_A
Are you sure you want to detach container cnt_A? [y/n] y
Container cnt_A detached from directory sanfs/homes.
```

Related topics

- “attachcontainer” on page 151
- “chcontainer” on page 159
- “Detaching a fileset” on page 105
- “Filesets” on page 24
- “lscontainer” on page 183
- “mkcontainer” on page 217
- “rmcontainer” on page 239
- “Standard format parameters” on page 312

exit

Ends a tanktool session.

Syntax



Parameters

`-?` | `-h` | `-help`

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

`-l` Displays a list of available commands with the syntax diagrams for each.

Examples

Ends a tanktool session The following example ends the tanktool session:

```
tanktool> exit
shell>
```

Related topics

- “Standard format parameters” on page 312
- “tanktool” on page 140

help

Displays a list of commands available in the Administrative CLI and optionally displays the syntax or brief description of each command.

Syntax

```
→ help [-? | -h | -help] [-l | -s] [-p [off | on]] [-r number] [command_name]
```

Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-l Displays a list of available commands with the syntax diagrams for each. If you specify a command name with this parameter, this command displays the syntax for only the specified command.

-s Displays a list of available commands with a brief description of each. If you specify a command name with this parameter, this command displays a brief description for only the specified command.

-p Specifies whether to display one page of text at a time or all text at once.

off Displays all text at one time. This is the default value.

on Displays one page of text at time. Pressing any key displays the next page.

-r number

Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.

command_name

Displays help information for the specified command, including the syntax diagram, parameter descriptions, return codes and errors, descriptions, examples, and miscellaneous remarks.

Description

If you specify this command with no parameters, this command displays only a list of available commands.

Examples

Display a description of a command The following example displays the description of the **mkimage** command:

```
tanktool>help -s lspool
```

```
lspool Displays a list of existing storage pools and their attributes.
```

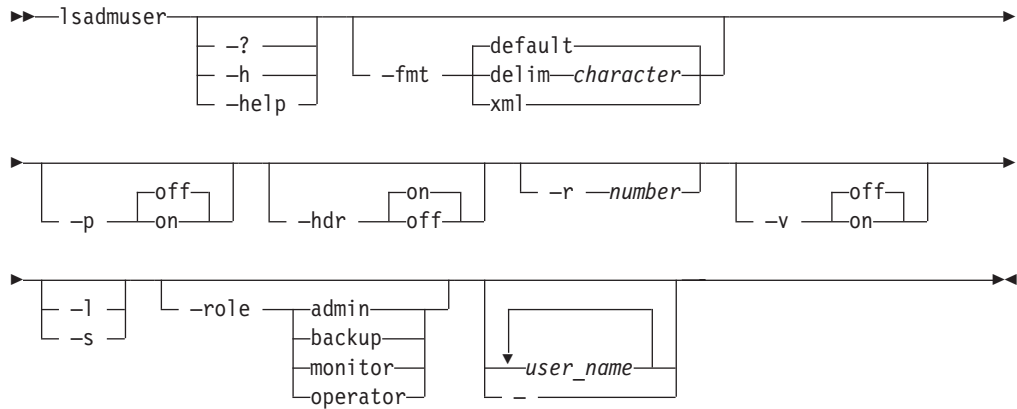
Related topics

- “Command modes” on page 311
- “Naming guidelines” on page 311
- “Standard format parameters” on page 312
- “Syntax diagram conventions” on page 315
- “User assistance for commands” on page 142

lsadmuser

Displays a list of administrative users and their attributes.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Name  User Role  Authorization  Authorization Timeout (secs)
=====
maki  Admin  Current  300
```

delim *character*

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
maki,Admin,Current,300 secs
```

xml

Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_AdminUser">
<PROPERTY NAME="Name" TYPE="string"><VALUE TYPE='string'>maki</VALUE>
</PROPERTY>
<PROPERTY NAME="EffectiveRole" TYPE="unit16"><VALUE="unit16">3</VALUE>
</PROPERTY>
<PROPERTY NAME="IsAuthorizationCurrent" TYPE="boolean"><VALUE
TYPE="boolean">>true</VALUE></PROPERTY>
<PROPERTY NAME="AuthCurrentRemainingTime" TYPE="uint32"><VALUE
TYPE="uint32">300</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>
```

-p Specifies whether to display one page of text at a time or all text at once.

- off** Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.
 - on** Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.
- hdr**
Specifies whether to display the table header.
- on** Displays the table header. This is the default value.
 - off** Does not display the table header.
- r *number***
Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.
- v** Specifies whether to enable verbose mode.
- off** Disables verbose mode. This is the default value.
 - on** Enables verbose mode.
- l** Displays additional information about the administrative users.
- s** Displays only the names of the administrative users.
- role**
Displays information for only those users with the specified user role . Possible roles are:
- admin** Shows only those users assigned to the Administrator role.
 - backup**
Shows only those users assigned to the Backup role.
 - monitor**
Shows only those users assigned to the Monitor role.
 - operator**
Shows only those users assigned to the Operator role.
- user_name***
Specifies the names of one or more administrative users to list. If not specified, this command lists all of the administrative users assigned to all user roles.
- << *file_name***
Specifies that you want to read the names of the one or more administrative users to list from stdin (for example, - << /work/users_list.txt).

Description

Note: This command operates only against the local engine.

Administrative users are defined in the LDAP server as authorized to perform administrative actions on the SAN File System. A user's role is determined by its membership in an LDAP group that is associated with the various SAN File System roles.

If you do not specify a listing parameter, this command displays the following default information for each administrative user:

- Name of the administrative user.
- Role assigned to the administrative user.

- Authorization (Current or Not Current).

If you specify the `-l` listing parameter, this command displays the following information in addition to the default information for each administrative user:

- Authorization timeout, in seconds.

Examples

Lists all administrative users The following example lists all SAN File System administrative users and additional information about their assigned roles.

```
tanktool> lsadmuser -l
```

Name	User Role	Authorization	Authorization Timeout (secs)
maki	Admin	Current	300
saki	Backup	Not Current	0
mizi	Operator	Current	465
jeko	Monitor	Not Current	0

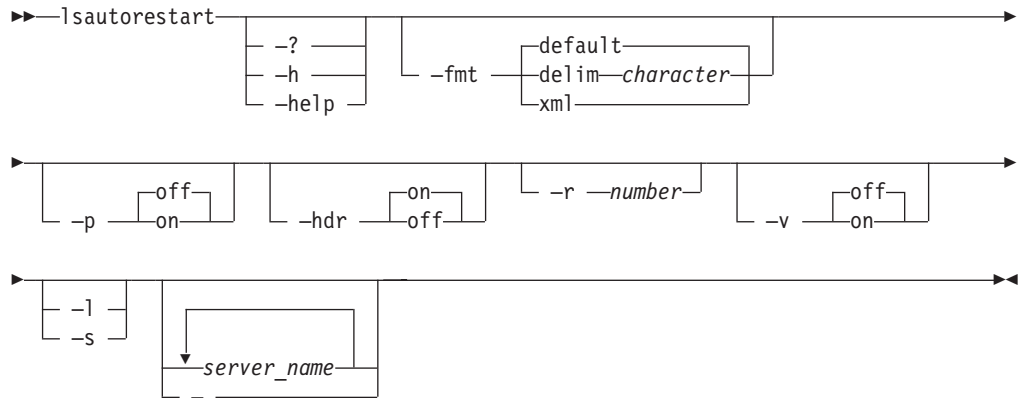
Related topics

- “Listing users” on page 127
- “resetadmuser” on page 234
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314
- “User roles” on page 46

Isautorestart

Displays a list of Metadata servers and the automatic-restart settings for each.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Name    Service State    Probe State    Last Probe                Probes
-----
ST1     Running           Live Server    Feb 11, 2003 9:35:47 AM    218690

Highest Retries
=====
                2
```

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
ST1,Running,Live Server,Feb 11, 2003 9:35:47 AM,218690,2
```

xml

Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_TankWatchdog">
<PROPERTY NAME="ServiceName" TYPE="string"><VALUE TYPE="string">ST1
</VALUE></PROPERTY>
<PROPERTY NAME="State" TYPE="uint32"><VALUE TYPE="uint32">1</VALUE>
</PROPERTY>
<PROPERTY NAME="ProbeState" TYPE="uint32"><VALUE TYPE="uint32">2</VALUE>
</PROPERTY>
<PROPERTY NAME="LastProbeTimeStamp" TYPE="datetime"><VALUE
TYPE="datetime">20030211093547&#46;000009&#45;420</VALUE></PROPERTY>
<PROPERTY NAME="TotalProbes" TYPE="uint64"><VALUE TYPE="uint64">218690
```

```

</VALUE></PROPERTY>
<PROPERTY NAME="RetriesHWM" TYPE="uint32"><VALUE TYPE="uint32">2</VALUE>
</PROPERTY>
</INSTANCE>
</IRETURNVALUE>

```

-p Specifies whether to display one page of text at a time or all text at once.

off Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.

on Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.

-hdr

Specifies whether to display the table header.

on Displays the table header. This is the default value.

off Does not display the table header.

-r *number*

Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.

-v Specifies whether to enable verbose mode.

off Disables verbose mode. This is the default value.

on Enables verbose mode.

-l Displays additional information about the Metadata servers.

-s Displays default information about the Metadata servers.

server_name

Specifies the names of one or more Metadata servers to display. If not specified, settings for all Metadata servers are displayed.

- Specifies that you want to read the names of one or more Metadata servers to display from stdin (for example, - << /work/server_list.txt).

Description

Note: If you run this command from an engine hosting a subordinate Metadata server, you can display information about only the local Metadata server. If you run this command from the engine hosting the master Metadata server, you can display information about any Metadata server.

If you do not specify a listing parameter, this command displays the following default information for each Metadata server:

- Name of the Metadata server.
- State of the Metadata server restart service. Possible states are:
 - Off** The Metadata server restart service is not running.
 - Failed** The Metadata server restart service is running but has failed to restart the Metadata server after several attempts.

Running

The Metadata server restart service is running and ready to restart the Metadata server if it fails.

Standby

The Metadata server restart service is running, but the Metadata server has been manually shut down. You must manually restart Metadata server to return the Metadata server restart service to the running state.

Unknown

The Metadata server restart service is in an unknown state because the Metadata server could not be reached.

- Probed Metadata server status. Possible status values are:

Absent Server

The Metadata server restart service could not find the Metadata server and will attempt to restart the Metadata server.

Live Server

The Metadata server restart service found the Metadata server and it is up and running. There is no need to restart the Metadata server.

Not Probed

The Metadata server restart service has not started probing the Metadata server because it is either in the off or standby state.

Probing

The Metadata server restart service has started probing the Metadata servers for status.

Unknown

The Metadata server restart service cannot determine whether the Metadata server is live or absent.

It is possible for the probe state to be unknown when the Metadata server is live. For example, if the Metadata server fails, the Metadata server restart service determines that the Metadata server is absent, and restarts it. If the Metadata server is slow to start, on the next probe cycle, the Metadata server is neither live nor absent. When it is probed again, the Metadata server is live.

Another example is if the local disk is full, and the Metadata server restart service tries to restart the Metadata server but cannot write to its log file. The Metadata server crashes, and the Metadata server restart service tries to restart it again. On the next probe cycle, the state of the Metadata server is absent or unknown.

- Timestamp when the Metadata server was last probed for status.
- Total number of probes that the Metadata server restart service performed.
- Highest number of times that the Metadata server restart service has attempted to restart the Metadata server.

If you specify the `-l` listing parameter, this command displays the following information in addition to the default information for each Metadata server:

- Timestamp when the automatic-restart service started on the Metadata server.
- Amount of time, in milliseconds, between each probe.
- Current number of times that the automatic-restart service has attempted to restart the Metadata server.
- Total number of times that the automatic-restart service has attempted to restart the Metadata server.
- Maximum number of times that the automatic-restart service will attempt to restart the Metadata server.
- Lowest number of times that the automatic-restart service has attempted to restart the Metadata server.
- Total number of times that a probe for a live Metadata server has taken longer than the test timeout interval.
- Maximum amount of time, in milliseconds to wait for a Metadata server to respond before declaring it down.
- Amount of time, in milliseconds, used the last time the Metadata server was probed to see if it is running.
- Highest amount of time, in milliseconds, taken to determine if the Metadata server is running.

- Lowest amount of time, in milliseconds, taken to determine if the Metadata server is running.
- Number of times the watchdog probed the Metadata server to determine if it is absent.
- Amount of time, in milliseconds, used the last time the Metadata server was probed to determine if it is absent.
- Highest amount of time, in milliseconds, taken to determine if the Metadata server is absent.
- Lowest amount of time, in milliseconds, taken to determine if the Metadata server is absent.

Examples

Displays the automatic-restart settings The following example displays the automatic-restart settings and additional information for all Metadata servers in the cluster.

```
tanktool> lsautorestart ST1
```

Name	Service State	Probe State	Last Probe	Probes
ST1	Running	Live Server	Feb 11, 2003 9:35:47 AM	5

```
Highest Retries
```

```
=====
```

```
2
```

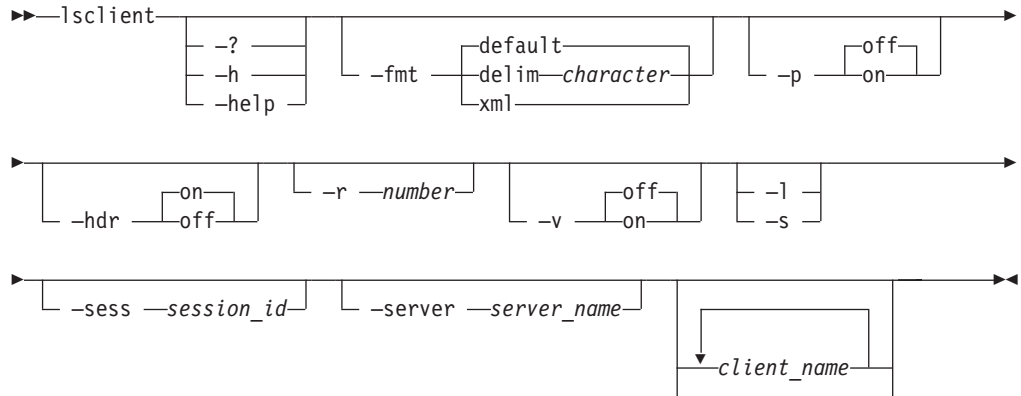
Related topics

- “Metadata server” on page 33
- “Standard format parameters” on page 312
- “startautorestart” on page 259
- “Starting the Metadata server restart service” on page 113
- “stopautorestart” on page 279

Isclient

Displays a list of clients that are currently being served by one or more Metadata servers in the cluster.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Client    Session ID  State    Server  Renewals  Privilege
-----
jeko      1           Current  ST1     45        Standard
```

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
jeko,1,Current,ST1,45,Standard
```

xml Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_RegisteredFSClient">
<PROPERTY NAME="Name" TYPE="string"><VALUE TYPE="string">jeko</VALUE>
</PROPERTY>
<PROPERTY NAME="Id" TYPE="uint64"><VALUE TYPE="uint64">1</VALUE>
</PROPERTY>
<PROPERTY NAME="State" TYPE="uint16"><VALUE TYPE="uint16">1</VALUE>
</PROPERTY>
<PROPERTY NAME="SystemName" TYPE="string"><VALUE TYPE="string">ST1
</VALUE></PROPERTY>
<PROPERTY NAME="LeaseRenewals" TYPE="uint64"><VALUE TYPE="uint64">45
```

```

</VALUE></PROPERTY>
<PROPERTY NAME="IsPrivilege" TYPE="boolean"><VALUE TYPE="boolean">>true
</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>

```

- p** Specifies whether to display one page of text at a time or all text at once.
 - off** Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.
 - on** Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.
- hdr** Specifies whether to display the table header.
 - on** Displays the table header. This is the default value.
 - off** Does not display the table header.
- r** *number* Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.
- v** Specifies whether to enable verbose mode.
 - off** Disables verbose mode. This is the default value.
 - on** Enables verbose mode.
- l** Displays additional information about each client.
- s** Displays limited information about each client.
- sess** *session_id* Specifies the client-session ID to list. If not specified, this command displays a list of all clients.
- server** *server_name* Specifies the name of a Metadata server to query for clients. If specified, this command displays only those clients that are served by the specified Metadata server. If not specified, this command displays clients that are served by all Metadata servers in the cluster.
- client_name* Specifies the names of one or more clients to list. If not specified, this command displays all clients that are being served by the specified Metadata server.
- Specifies that you want to read the names of one or more clients to list from stdin (for example, - << /work/client_list.txt).

Description

Prerequisite: You must be logged on to the master Metadata server to use the command.

When you specify multiple parameters in this command, these behaviors apply:

- Flags work as AND coordinators.
- Objects work as OR coordinators.

- When flags and objects are combined, they are listed in this manner: WHERE flag_1 AND flag_2 AND ... AND flag_n AND (object_1 OR object_2 OR ... OR object_m).
- When flags or objects are not required and not specified, this command searches all possible values unless a default overrides this rule.

For example, if you specify `-sess`, `-server`, and two client names, the search would be performed in the following manner: WHERE session_ID AND server_name AND (client_name_1 OR client_name_2).

If you do not specify a listing parameter, this command displays the following default information for each client:

- Client name. This name is configured by the administrator on the client node.
- Client-session ID. Each client can have one session running on each Metadata server simultaneously.
- State of the client. Possible states are:

Current

The client is currently being served by one of the Metadata servers.

Expired

The client is not currently being served by one of the Metadata servers.

- Metadata server that is serving the client.
- Number of renewals since the client-session was started.
- Client privilege or access level. Possible privileges are:

Root Indicates that the user has root privileges, and is considered a privileged user.

Standard

Indicates that the user has default access level.

If you specify the `-s` listing parameter, this command displays only the following information for each client:

- Client name.
- Client-session ID.
- Metadata server that is serving the client.

If you specify the `-l` listing parameter, this command displays the following information in addition to the default information for each client:

- Date and time of the last client-session renewal.
- Time until the next client-session renewal is to occur, in seconds.
- Client IP address.
- Client port number.
- Client operating-system platform.
- Client file-system driver version.
- Number of transactions started, per client session.
- Number of transactions that have completed, per client session.
- Number of session locks, per client session.
- Number of data locks, per client session.
- Number of byte-range locks, per client session.

Note: The Metadata server does not know which clients are connected to SAN File System. If you specify a client that is not connected, this command will return a warning that no rows were found.

Examples

List clients The following example displays information about all clients currently being served by Metadata server ST1.

```
tanktool> lsclient -server ST0,ST1
```

Client	Session ID	State	Server	Renewals	Privilege
jeko	1	Current	ST1	45	Standard
mefi	2	Current	ST1	1546	Root
saki	3	Expired	ST0	587	Standard

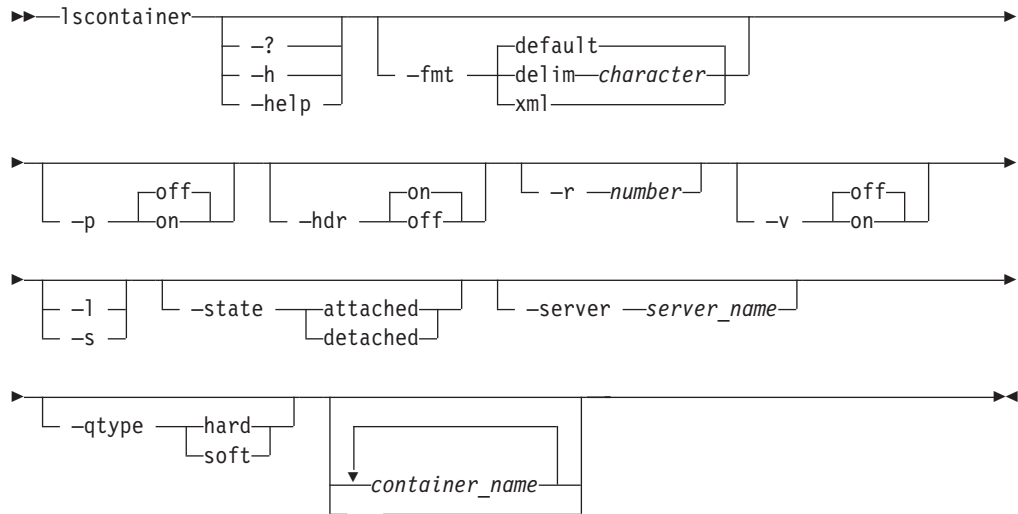
Related topics

- “Listing client sessions” on page 76
- “Listing clients with root privileges” on page 77
- “Clients” on page 11
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314
- “Viewing client-session details” on page 81
- “Viewing client-session statistics” on page 82

lscontainer

Displays a list of filesets (containers) and their attributes.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Name      Container State  Quota  Quota (MB)  Used (MB)  Used (%)
          Type
=====
cnt_B     Attached      Soft   20         10         50

Threshold (%)  Most Recent Image
=====
19    Feb 6, 2003 14:54:15 PM
```

delim *character*

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
cnt_B,Attached,Soft,20 MB,10 MB,50%,19%,Feb 6, 2003 14:54:15 PM
```

xml Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_Container">
<PROPERTY NAME="Name" TYPE="string"><VALUE TYPE="string">cnt_B</VALUE>
</PROPERTY>
```

```

<PROPERTY NAME="State" TYPE="uint32"><VALUE TYPE="uint32">1</VALUE>
</PROPERTY>
<PROPERTY NAME="Quota" TYPE="uint64"><VALUE TYPE="uint64">20</VALUE>
</PROPERTY>
<PROPERTY NAME="IsHardQuota" TYPE="boolean"><VALUE TYPE="boolean">>false
</VALUE></PROPERTY>
<PROPERTY NAME="AlertPercentage" TYPE="uint16"><VALUE TYPE="uint16">19
</VALUE></PROPERTY>
<PROPERTY NAME="SizeAllocated" TYPE="uint64"><VALUE TYPE="uint64">10
</VALUE></PROPERTY>
<PROPERTY NAME="SizeAllocatedPercentage" TYPE="uint16"><VALUE
TYPE="uint16">50</VALUE></PROPERTY>
<PROPERTY NAME="LastPITCopyDate" TYPE="datetime"><VALUE TYPE="datetime">
20030206145415&#46;000000&#43;000</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>

```

- p** Specifies whether to display one page of text at a time or all text at once.
 - off** Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.
 - on** Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.
- hdr** Specifies whether to display the table header.
 - on** Displays the table header. This is the default value.
 - off** Does not display the table header.
- r *number*** Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.
- v** Specifies whether to enable verbose mode.
 - off** Disables verbose mode. This is the default value.
 - on** Enables verbose mode.
- l** Displays additional information about the filesets.
- s** Displays only the names of the filesets.
- state** Lists only the filesets that match the specified state. Possible states are:
 - attached** Lists only the filesets that are attached to the global namespace.
 - detached** Lists only the filesets that are not attached to the global namespace.
- server *server_name*** Lists only filesets assigned to the specified Metadata server.
- qtype** Lists only the filesets (containers) that match the specified quota type. Possible types are:
 - hard** Lists only the filesets that have hard quotas.
 - soft** Lists only the filesets that have soft quotas.
- container_name*** Specifies the names of one or more filesets (or containers) to list. If not

specified, this command lists all the filesets. If not specified, this command displays a list of all filesets attached to the global namespace.

- Specifies that you want to read the names of one or more filesets to list from stdin (for example, - << /work/cnt_list.txt).

Description

Prerequisite: You must be logged in to the engine hosting the master Metadata server to run this command.

When you specify multiple parameters in this command, these behaviors apply:

- Flags work as AND coordinators.
- Objects work as OR coordinators.
- When flags and objects are combined, they are listed in this manner: WHERE flag_1 AND flag_2 AND ... AND flag_n AND (object_1 OR object_2 OR ... OR object_m).
- When flags or objects are not required and not specified, this command searches all possible values unless a default overrides this rule.

For example, if you specify `-state`, `-qtype`, and two fileset names, the search would be performed in the following manner: WHERE state AND quota_type AND (fileset_name_1 OR fileset_name_2).

If you do not specify a listing parameter, this command displays the following default information for each fileset:

- Name of the fileset.
- State of the fileset (Detached, Idle, or Inuse).
- Quota type.
- Quota size, in MB.
- Amount of space used, in MB.
- Threshold, as a percentage of space used.
- Timestamp of the most recent FlashCopy image.

Note: If the quota is not set, the values for the amount of space used (Used %) and threshold (Threshold %) are displayed as 0 and have no meaning. Because the threshold value is based on the amount of space used, if the amount of space used is not defined, then any value specified for the threshold value will have no meaning and no alert will be sent if the displayed value is exceeded.

If you specify the `-l` listing parameter, this command displays the following information in addition to the default information for each fileset:

- Metadata server that hosts the fileset.
- State of the Metadata server.
- Number of FlashCopy images that exist for this fileset.
- Attach point.
- Directory name.
- Directory path.
- Parent fileset.
- Number of nested filesets (children).
- Fileset description.

Examples

Listing filesets The following example lists all the available information about all filesets in the global namespace.

```
tanktool> lscontainer -l
```

Name	Container State	Quota Type	Quota (MB)	Used (MB)	Used (%)
ROOT	Attached	Soft	20	10	50
cnt_B	Detached	Hard	88	100	48
cnt_C	Attached	Hard	120	400	45

```
Threshold (%) Most Recent Image
```

```
=====
          19 -
          70 Nov 6, 2003 20:01:48 PM
           2 Mar 7, 2003 10:36:37 PM
```

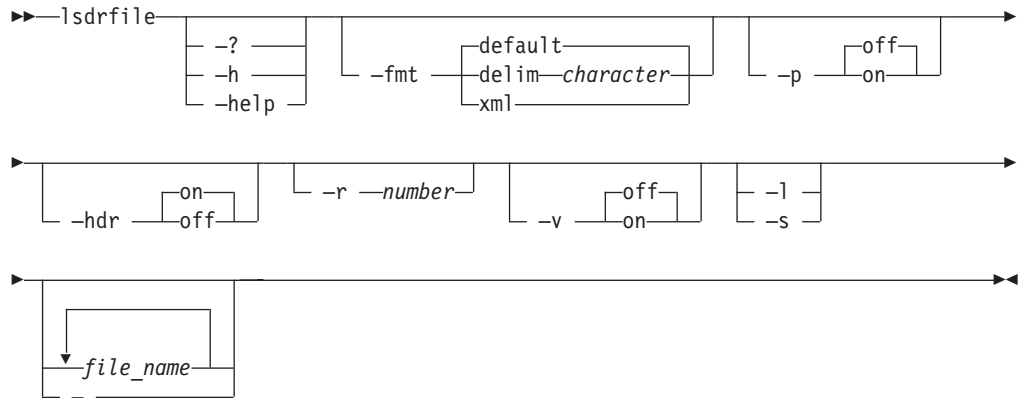
Related topics

- “attachcontainer” on page 151
- “chcontainer” on page 159
- “detachcontainer” on page 168
- “Filesets” on page 24
- “Listing volumes” on page 131
- “mkcontainer” on page 217
- “rmcontainer” on page 239
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314
- “Viewing fileset details” on page 106

lsdrfile

Displays a list of system-metadata disaster-recovery files.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Name          Date and Time          Size (KB)
=====
SysMetadata   Jan 3, 2003 4:35:46 PM  10000
```

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
SysMetadata,Jan 3, 2003 4:35:46 PM,10000 KB
```

xml

Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_SystemMDRAid">
<PROPERTY NAME="Name" TYPE="string"><VALUE TYPE="string">SysMetadata
</VALUE></PROPERTY>
<PROPERTY NAME="InstallDate" TYPE="datetime"><VALUE TYPE="datetime">
20030103043546&#46;000000&45;420</VALUE></PROPERTY>
<PROPERTY NAME="Size" TYPE="uint64"><VALUE TYPE="uint64">10000</VALUE>
</PROPERTY>
h</INSTANCE>
</IRETURNVALUE>
```

-p Specifies whether to display one page of text at a time or all text at once.

off

Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.

on Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.

-hdr

Specifies whether to display the table header.

on Displays the table header. This is the default value.

off Does not display the table header.

-r number

Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.

-v Specifies whether to enable verbose mode.

off Disables verbose mode. This is the default value.

on Enables verbose mode.

-l Displays the default information for the system-metadata disaster-recovery files.

-s Displays only the names of the system-metadata disaster-recovery files.

file_name

Specifies the name of one or more system-metadata disaster-recovery files to display. If not specified, this command displays information for all system-metadata disaster-recovery files.

– Specifies the file from which this commands reads the names of one or more system-metadata disaster-recovery files to display.

Description

Note: This command operates only against the local engine.

If you do not specify a listing parameter or if you specify the **-l** listing parameter, this command displays the following default information for each system-metadata disaster-recovery file:

- Name of the disaster-recovery file.
- Date and time the disaster-recovery file was created.
- Size of the disaster-recovery file, in KB.

Examples

List all disaster-recovery files The following example displays information for all disaster-recovery files.

```
tanktool> lsdrfile
```

Name	Date and Time	Size (KB)
SysMetadata	Jan 3, 2003 4:35:46 PM	10000

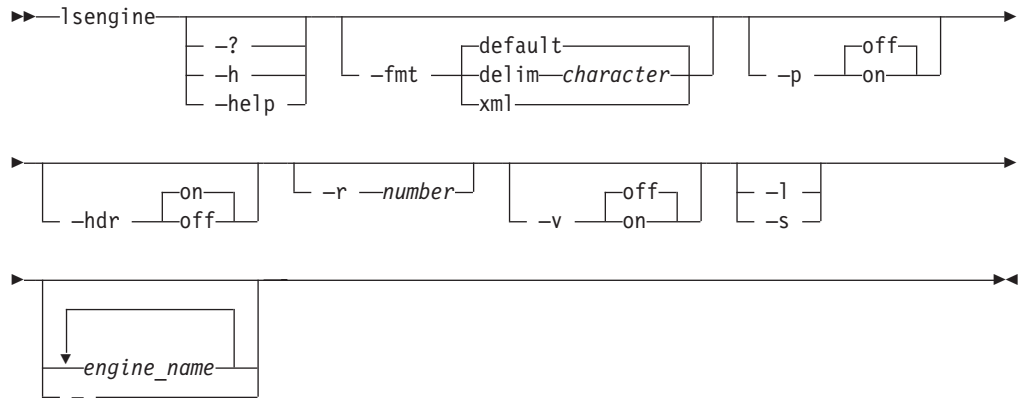
Related topics

- “buildscript” on page 153
- “mkdrfile” on page 220
- “rmdrfile” on page 241
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314

lsengine

Displays a list of storage engines and their attributes.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Name   Engine IP      Boot State  Temp      Fans      Voltage
=====
ST0    129.42.16.99  In POST    Warning   Normal    Normal
```

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
ST0,129.42.16.99,In POST,Warning,Normal,Normal
```

xml

Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_ComputerSystem">
<PROPERTY NAME="CurrentState" TYPE="uint32"><VALUE TYPE="uint32">1
</VALUE></PROPERTY>
<PROPERTY NAME="Name" TYPE="string"><VALUE TYPE="string">ST0</VALUE>
</PROPERTY>
</INSTANCE>
</IRETURNVALUE>
```

-p Specifies whether to display one page of text at a time or all text at once.

off

Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.

on Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.

-hdr

Specifies whether to display the table header.

on Displays the table header. This is the default value.

off Does not display the table header.

-r number

Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.

-v Specifies whether to enable verbose mode.

off Disables verbose mode. This is the default value.

on Enables verbose mode.

-l Displays the default information for each engine.

-s Displays only the IP address and name of each engine.

engine_name

Specifies the names of one or more storage engines to display. If not specified, this command displays information for all storage engines.

– Specifies the file from which this command reads the names of one or more storage engines to display.

Description

If you specify the **-s** listing parameter, this command displays the IP address and name of each engine.

If you do not specify a listing parameter or if you specify the **-l** listing parameter, this command displays the following default information for each engine:

- Engine IP address.
- Engine name.
- Boot state. Possible values are:

Before POST

The engine is powered on but has not started the power-on self test (POST).

Booted Flash

The engine has read the System BIOS but has not started loading the operating system.

Booting OS

The engine has started but not completed loading the operating system.

CPUs Held in Reset

The engine has been reset after a hardware fault.

In POST

The engine is running the POST.

In OS The engine is running in the normal state.

Stopped in POST (error detected)

The engine is powered on but has not completed the POST due to an error.

Unknown / Power Off

The boot state is unknown. The engine could be powered off.

- Temperature state. Possible values are:
 - The temperature thresholds are not set.

Normal

The temperatures of all components in all engine are below the Warning threshold.

Warning

The temperature of one or more engine components are above the Warning threshold.

Error The temperature of one or more engine components is higher than the Soft Shutdown threshold.

Unknown

The RSA card could not be accessed.

- Fan state. Possible values are:

Normal

All fans are operating above 15% of its fan-speed capacity.

Warning

One or more fans are operating below 15% of its fan-speed capacity.

Error The RSA card could not be accessed.

- Voltage state. Possible values are:
 - The voltage thresholds are not set.

Normal

The voltages of all components in all engines are above the Warning Low Voltage threshold and below the Warning High Voltage threshold.

Warning

The voltage of one or more engine components is below the Warning Low Voltage threshold or above the Warning High Voltage threshold.

Error The RSA card could not be accessed.

Examples

List engines The following example displays information about two engines: ST0 and ST1.

```
tanktool> lsengine ST1 ST2
```

Name	Engine IP	Boot State	Temp	Fans	Voltage
ST0	129.42.16.99	In POST	Warning	Normal	Normal
ST1	129.42.16.98	In OS	Normal	Normal	Normal

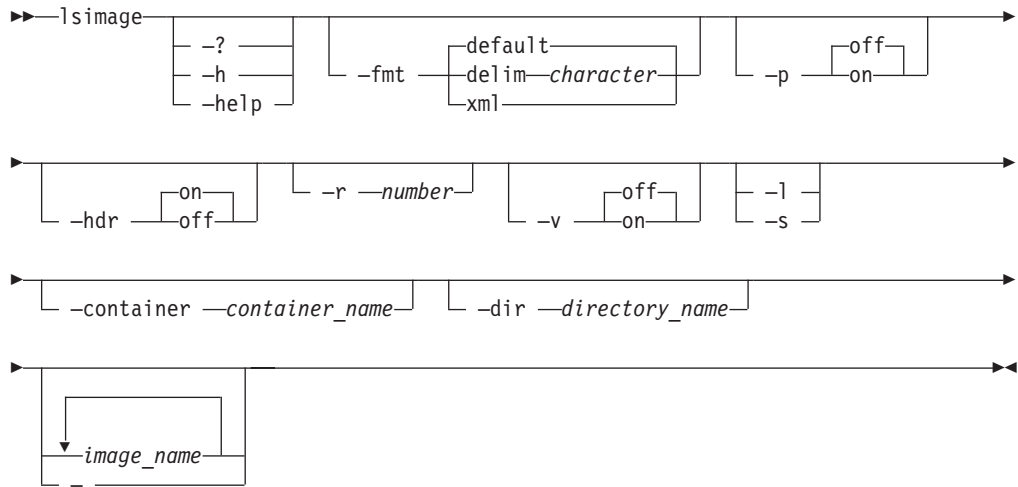
Related topics

- “Listing engines” on page 95
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314
- “startengine” on page 262
- “statengine” on page 272
- “stopengine” on page 281

lsimage

Displays a list of FlashCopy images in all filesets or a specified fileset.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Name          Container  Directory Name  Date
=====
FCA_image     cnt_A      personnel       Dec 3, 2003 5:48:46 PM
```

delim *character*

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
FCA_image,cnt_A,personnel,Dec 3, 2003 5:48:46 PM
```

xml

Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_PitImage">
<PROPERTY NAME="ContainerName" TYPE="string"><VALUE TYPE="string">cnt_A<
/VALUE></PROPERTY>
<PROPERTY NAME="Name" TYPE="string"><VALUE TYPE="string">FCA_imag<
/VALUE></PROPERTY>
<PROPERTY NAME="InstallDate" TYPE="datetime"><VALUE TYPE="datetime">
20031203174846&#46;000009&#45;420</VALUE></PROPERTY>
<PROPERTY NAME="DirectoryName" TYPE="string"><VALUE TYPE="string">
personnel</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>
```

- p** Specifies whether to display one page of text at a time or all text at once.
 - off** Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.
 - on** Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.
- hdr** Specifies whether to display the table header.
 - on** Displays the table header. This is the default value.
 - off** Does not display the table header.
- r *number*** Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.
- v** Specifies whether to enable verbose mode.
 - off** Disables verbose mode. This is the default value.
 - on** Enables verbose mode.
- l** Displays additional information about the FlashCopy images.
- s** Displays only the name of the FlashCopy images and the fileset associated with each.
- container *container_name*** Specifies the name of the fileset for which to list all FlashCopy images (or containers). If not specified, this command lists FlashCopy images in all filesets.
- dir *directory_name*** Specifies the directory under the `.flashcopy` directory from which to list the FlashCopy image. This directory can be up to 256 characters in length and must not contain backslash (\) or forward slash (/) characters as delimiters. If not specified, this command lists the FlashCopy images in all directories under the `.flashcopy` directory.
- image_name** Specifies the names of one or more FlashCopy images to display. If not specified, this command lists all FlashCopy images in the specified fileset and directory.
- Specifies that you want to read the names of one or more FlashCopy images to display from stdin (for example, `- << /work/image_list.txt`).

Description

Prerequisite: You must be logged in to the engine hosting the master Metadata server to run this command.

When you specify multiple parameters in this command, these behaviors apply:

- Flags work as AND coordinators.
- Objects work as OR coordinators.
- When flags and objects are combined, they are listed in this manner:
WHERE flag_1 AND flag_2 AND ... AND flag_n AND (object_1 OR object_2 OR ... OR object_m).

- When flags or objects are not required and not specified, this command searches all possible values unless a default overrides this rule.

For example, if you specify `-container`, `-dir`, and two FlashCopy image names, the search would be performed in the following manner: WHERE fileset AND directory AND (image_name_1 OR image_name_2).

The most-recently created FlashCopy image are listed first.

If no parameters are specified, this command lists all FlashCopy images in all filesets and directories.

If you do not specify a listing parameter, this command displays the following default information for each FlashCopy image:

- FlashCopy image name.
- Fileset name.
- Directory name.
- Timestamp when the FlashCopy image was created.

If you specify the `-l` listing parameter, this command displays the following information in addition to the default information for each FlashCopy image:

- Description.

Examples

Listing FlashCopy images The following example lists all the available information about the FlashCopy images.

```
tanktool> lsimage -l
```

Name	Container	Directory Name	Date	Description
FCC_image	cnt_C	payroll	Dec 3, 2003 5:59:14 PM	Payroll data
FCB_image	cnt_B	website	Dec 3, 2003 5:53:26 PM	Web site data
FCA_image	cnt_A	personnel	Dec 3, 2003 5:48:46 PM	Personnel data

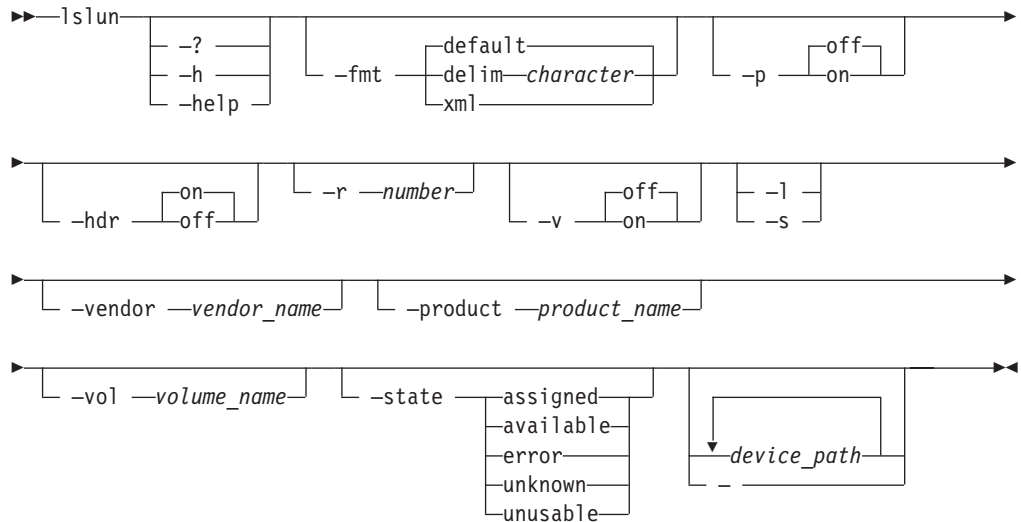
Related topics

- “Backup and restore” on page 5
- “Listing FlashCopy images” on page 108
- “FlashCopy images” on page 26
- “mkimage” on page 221
- “reverttoimage” on page 237
- “rmimage” on page 242
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314

lslun

Lists the LUNs that are accessible from SAN File System.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
OS Device Path LUN ID Engine WWN Vendor Product
=====
/dev/sda 2 12:34:56:78:91:23:45:67 IBM 2105F20
```

```
Size (MB) Volume State
=====
47000 Vol1 Assigned
```

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
/dev/sda,2,12:34:56:78:91:23:45:67,IBM,2105F20,47000,Vol1,Assigned
```

xml

Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_AvailableLUNs">
<PROPERTY NAME="LUNID" TYPE="uint64"><VALUE TYPE="uint64">2</VALUE>
</PROPERTY>
<PROPERTY NAME="NodeWWN" TYPE="string"><VALUE TYPE="string">
12&58;34&58;56&58;78&58;91&58;23&58;45&58;67</VALUE></PROPERTY>
```

```

<PROPERTY NAME="Vendor" TYPE="string"><VALUE TYPE="string">IBM</VALUE>
</PROPERTY>
<PROPERTY NAME="Product" TYPE="string"><VALUE TYPE="string">2105F20
</VALUE></PROPERTY>
<PROPERTY NAME="Size" TYPE="uint64"><VALUE TYPE="uint64">47000</VALUE>
</PROPERTY>
<PROPERTY NAME="State" TYPE="uint64"><VALUE TYPE="uint64">2</VALUE>
</PROPERTY>
<PROPERTY NAME="VolumeName" TYPE="string"><VALUE TYPE="string"></VALUE>
</PROPERTY>
<PROPERTY NAME="DeviceID" TYPE="string"><VALUE TYPE="string">
&47;dev&47;sda</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>

```

-p Specifies whether to display one page of text at a time or all text at once.

off Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.

on Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.

-hdr

Specifies whether to display the table header.

on Displays the table header. This is the default value.

off Does not display the table header.

-r *number*

Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.

-v Specifies whether to enable verbose mode.

off Disables verbose mode. This is the default value.

on Enables verbose mode.

-l Displays default information about the LUNs.

-s Displays default information about the LUNs.

-vendor *vendor_name*

Displays information for only the LUNs that match the specified vendor name.

-product *product_name*

Displays information for only the LUNs that match the specified product name.

-vol *volume_name*

Displays information for only the LUNs that match the specified SAN File System volume name.

-state

Displays information for only the LUNs that match the specified state. Possible states are:

assigned

Displays only those LUNs that are assigned to a storage pool and usable.

available

Displays only those LUNs that are visible to SAN File System and usable but are not assigned to a storage pool.

error Displays only those LUNs for which an error occurred while determining their availability.

unknown Displays only those LUNs whose availability could not be determined because the Metadata server is not running.

unusable Displays only those LUNs that are unsuitable to be a volume. A LUN is considered unsuitable if it is read-only, if its availability from all Metadata servers in the cluster is inconsistent, or if the LUN does not have an associated raw device.

device_path

Specifies one or more operating-system device paths for which to display LUNs.

- Specifies that you want to read one or more operating-system device paths for which to display LUNs from stdin (for example, - << /work/device_paths_list.txt).

Description

Note: This command operates only against the local engine.

If you do not specify any filtering parameters, this command lists all the LUNs that are accessible to SAN File System, including LUNs that have not been added to SAN File System, SAN File System volumes, and unusable LUNs.

This command displays the following default information for each LUN:

- Operating-system device path.
- LUN ID.
- Worldwide name of the engine.
- Vendor.
- Product.
- Size of the device, in MB.
- Volume name, if the LUN is assigned in SAN File System.
- State of the LUN.

Examples

Display LUNs The following example lists all the attributes of all the available LUNs:

```
tanktool> ls1un
```

OS Device Path	LUN ID	Storage Engine WWN	Vendor	Product
/dev/sda	2	12:34:56:78:91:23:45:67	IBM	2105F20
/dev/sda7	3	89:12:34:56:78:91:23:45	IBM	2105F20
/dev/sda8	4	88:19:34:56:78:91:23:41	IBM	2105800

Size (MB)	Volume	State
47000	Vol1	Assigned
12	-	Available
80000	-	Unusable

Related topics

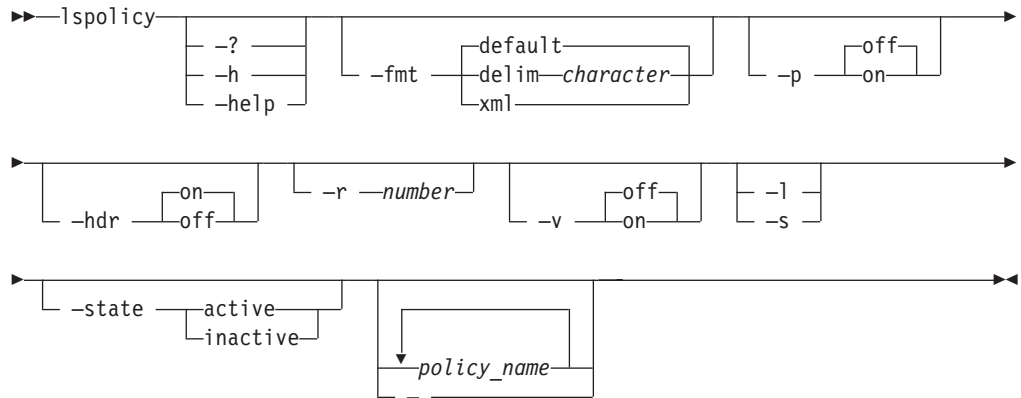
- “Listing available LUNs” on page 131
- “Listing LUNs” on page 131
- “lsvol” on page 213
- “Standard format parameters” on page 312

- “Standard listing parameters” on page 314
- “Viewing available LUN details” on page 134
- “Viewing LUN details” on page 134
- “Volumes” on page 49

lspolicy

Lists the active and inactive policies.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Name          State   Last Active          Modified
=====
policy1       Active  Jul 19, 2003 14:00:34 PM  Jul 19, 2003 13:59:34 PM

Description
=====
Active policy
```

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
policy1,Active,Jul 19, 2003 14:00:34 PM,Jul 19, 2003 13:59:34 PM,Active policy
```

xml

Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_PolicySet">
<PROPERTY NAME="Name" TYPE="string"><VALUE TYPE="string">policy1</VALUE>
</PROPERTY>
<PROPERTY NAME="State" TYPE="uint16"><VALUE TYPE="uint16">1</VALUE>
</PROPERTY>
<PROPERTY NAME="Description" TYPE="string"><VALUE TYPE="string">
Active&#32;policy</VALUE></PROPERTY>
<PROPERTY NAME="LastModificationDate" TYPE="datetime"><VALUE
TYPE="datetime">20030719135934&#45;000000&#45;420</VALUE></PROPERTY>
```

```
<PROPERTY NAME="LastActiveDate" TYPE="datetime"><VALUE TYPE="datetime">
20030719140034&#45;000000&#45;420</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>
```

- p** Specifies whether to display one page of text at a time or all text at once.
 - off** Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.
 - on** Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.
- hdr** Specifies whether to display the table header.
 - on** Displays the table header. This is the default value.
 - off** Does not display the table header.
- r number** Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.
- v** Specifies whether to enable verbose mode.
 - off** Disables verbose mode. This is the default value.
 - on** Enables verbose mode.
- l** Displays the default information about each policy.
- s** Displays only the name of each policy.
- state active | inactive** Specifies the state of the policies to display. You can specify one of the following values:
 - active** Displays only the active policy.
 - inactive** Displays only the inactive policies.
- policy_name* Specifies the names of one or more policies to list.
- Specifies that you want to read the names of one or more policies to list from stdin (for example, **- << /work/policies_list.txt**).

Description

Prerequisite: You must be logged in to the engine hosting the master Metadata server to run this command.

When you specify multiple parameters in this command, these behaviors apply:

- Flags work as AND coordinators.
- Objects work as OR coordinators.
- When flags and objects are combined, they are listed in this manner:
WHERE flag_1 AND flag_2 AND ... AND flag_n AND (object_1 OR object_2 OR ... OR object_m).
- When flags or objects are not required and not specified, this command searches all possible values unless a default overrides this rule.

For example, if you specify `-state` and two policy names, the search would be performed in the following manner: `WHERE state AND (policy_name_1 OR policy_name_2)`.

If you do not specify a listing parameter or if you specify the `-l` listing parameter, this command displays the following information for each policy:

- Name of the policy.
- State of the policy.
- Timestamp when the policy was last active, if applicable, based on coordinated universal time (UTC).
- Timestamp when the policy was last modified, based on coordinated universal time (UTC).
- Policy description.

Examples

Display policies The following example displays all the available information about all the policies:

```
tanktool> lspolicy -l
```

Name	State	Last Active	Modified
policy1	Active	Jul 19, 2003 14:00:34 PM	Jul 19, 2003 13:59:34 PM
testpolicy	Inactive	Mar 18, 2003 10:30:17 PM	Mar 18, 2003 10:06:25 PM
DEFAULT	Inactive	Jan 11, 2003 16:36:24 PM	Jan 11, 2003 15:36:14 PM

Description

=====

Active policy
Test policy
Default Policy

Display the rules of the active policy The following example displays the rules of the active policy:

```
$tanktool lspolicy -s -state active -hdr off | tanktool catpolicy -  
VERSION 1  
rule 'stgRule1' set stgpool 'pool1' for fileset ('cnt_A')  
rule 'stgRule2' set stgpool 'pool2' where NAME like '%.doc'  
rule 'stgRule3' set stgpool 'pool3' where DAYOFWEEK(CREATION_DATE) == 1  
rule 'stgRule4' set stgpool 'pool4' where USER_ID <= 100
```

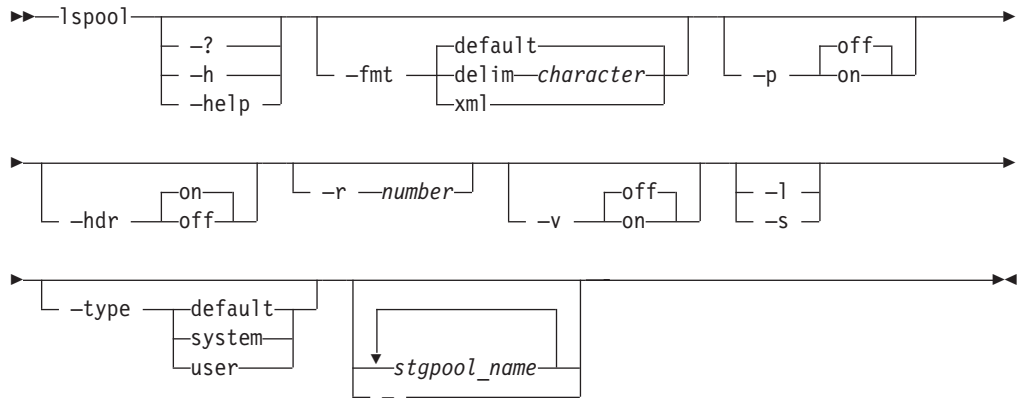
Related topics

- “catpolicy” on page 156
- “Listing policies” on page 120
- “mkpolicy” on page 223
- “Policies and rules” on page 44
- “rmpolicy” on page 244
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314
- “usepolicy” on page 287
- “Viewing policy details” on page 121
- “Viewing policy rules” on page 121

lspool

Displays a list of the existing storage pools and their attributes.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

Name	Type	Size (MB)	Used (MB)	Used (%)	Threshold (%)	Volumes
Poo11	User	1400	575	41	90	2

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation mark (') or double quotation mark ("). A blank space is not a valid character. For example:

Poo11,User,

xml Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_StoragePool">
<PROPERTY NAME="Name" TYPE="string"><VALUE TYPE="string">Poo11</VALUE>
</PROPERTY>
<PROPERTY NAME="PoolType" TYPE="uint32"><VALUE TYPE="uint32">0</VALUE>
</PROPERTY>
<PROPERTY NAME="AlertPercentage" TYPE="uint16"><VALUE TYPE="uint16">90
</VALUE></PROPERTY>
<PROPERTY NAME="Size" TYPE="uint64"><VALUE TYPE="uint64">1400</VALUE>
</PROPERTY>
<PROPERTY NAME="SizeAllocated" TYPE="uint64"><VALUE TYPE="uint64">575
</VALUE></PROPERTY>
<PROPERTY NAME="SizeAllocatedPercentage" TYPE="uint16"><VALUE
TYPE="uint16">41</VALUE></PROPERTY>
```

```
<PROPERTY NAME="NumberOfVolumes" TYPE="uint32"><VALUE TYPE="uint32">2
</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>
```

-p Specifies whether to display one page of text at a time or all text at once.

off Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.

on Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.

-hdr

Specifies whether to display the table header.

on Displays the table header. This is the default value.

off Does not display the table header.

-r *number*

Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.

-v Specifies whether to enable verbose mode.

off Disables verbose mode. This is the default value.

on Enables verbose mode.

-l Displays additional information about the storage pools.

-s Displays only the names of the storage pools.

-type

Displays information for only those storage pools that match the specified type. If not specified, all storage pools are displayed. Possible types are:

default

Displays information for the DEFAULT storage pool.

system

Displays information for the SYSTEM storage pool.

user

Displays information for all user storage pools.

stgpool_name

Specifies the names of one or more storage pools to display.

- Specifies that you want to read the names of one or more storage pools to display from stdin (for example, **- << /work/stgpools_list.txt**).

Description

Prerequisite: You must be logged in to the engine hosting the master Metadata server to run this command.

If you do not specify a listing parameter, this command displays the following default information for each storage pool:

- Name of the storage pool.
- Type of storage pool.
- Size of the storage pool, in MB.
- Amount of space used, in MB.
- Percent of space used.
- Alert threshold.

- Number of volumes in the storage pool.

If you specify the `-l` listing parameter, this command displays the following information in addition to the default information for each storage pool:

- Partition size, in MB (16, 64, or 256).
- Allocation size, in KB (auto, 4 or 128).
- Pool description.

Examples

Display a long listing The following example lists all the attributes of all the storage pools:

```
tanktool> lspool -l
```

Name	Type	Size (MB)	Used (MB)	Used (%)	Threshold (%)	Volumes
Pool1	User	1400	575	41	90	2
Pool2	User	2000	1000	50	90	3
DEFAULT	Default	10000	2500	25	80	10
SYSTEM	System	10000	7500	75	80	10

Partition Size (MB)	Allocation Size (KB)	Description
16	2	Accounting
16	3	Engineering
64	128	DEFAULT pool
64	128	SYSTEM pool

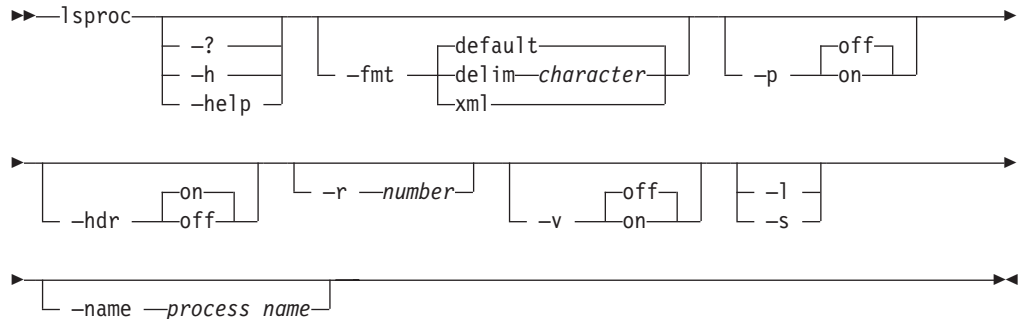
Related topics

- “chpool” on page 161
- “mkpool” on page 225
- “Storage pools” on page 37
- “rmpool” on page 245
- “setdefaultpool” on page 251
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314
- “Listing storage pools” on page 124
- “Viewing storage pool details” on page 125

lsproc

Displays a list of long-running processes that are not yet complete and their attributes.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
ID  Name                Started
=====
5   CHECKMETADATA      Feb 11, 2003 9:29:56:13 PM
```

delim *character*

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation mark (') or double quotation mark ("). A blank space is not a valid character. For example:

```
5,CHECKMETADATA,Feb 11, 2003 9:29:56:13 PM
```

xml

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_AdminProcesses">
<PROPERTY NAME="Id" TYPE="uint64"><VALUE TYPE="uint64">5</VALUE>
</PROPERTY>
<PROPERTY NAME="Command" TYPE="string"><VALUE TYPE="uint32">STOPSERVER
</VALUE></PROPERTY>
<PROPERTY NAME="InstallDate" TYPE="datetime"><VALUE TYPE="datetime">
20030211092956&#46;000000&#45;420</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>
```

-p

Specifies whether to display one page of text at a time or all text at once.

off Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.

on Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.

-hdr

Specifies whether to display the table header.

on Displays the table header. This is the default value.

off Does not display the table header.

-r *number*

Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.

-v Specifies whether to enable verbose mode.

off Disables verbose mode. This is the default value.

on Enables verbose mode.

-l Displays additional information about the processes.

-s Displays only the identifier of each process.

-name *process_name*

Specifies the name of a process to list.

Description

Prerequisite: You must be logged in to the engine hosting the master Metadata server to run this command.

If you do not specify a listing parameter or if you specify the **-l** listing parameter, this command displays the following information for each process:

- Identifier of the process.
- Name of the process.
- Time that the process was started.

Examples

Lists long-running processes The following example lists the long-running processes currently in progress.

```
tanktool> lsproc
```

ID	Name	Started
5	CHECKMETADATA	Feb 11, 2003 9:29:56:13 PM
7	STOPCLUSTER	Feb 10, 2003 9:30:20:24 PM
9	STARTSERVER	Feb 10, 2003 9:12:22:24 PM

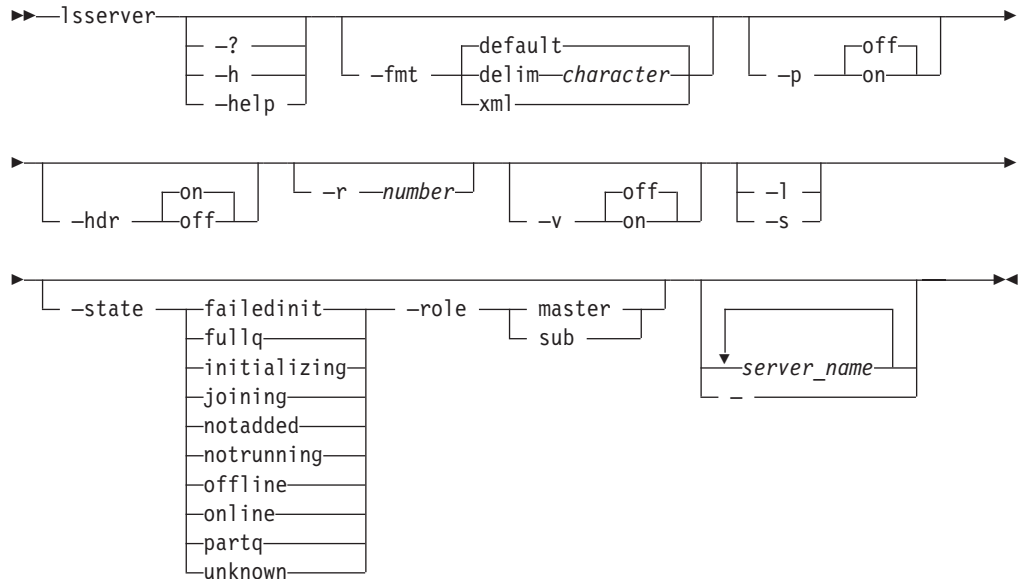
Related topics

- “Listing processes” on page 121
- “lsproc” on page 205
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314

Isserver

Displays a lists of all Metadata servers in the cluster and their attributes if issued from the master Metadata server, or displays attributes about the local Metadata server if issued from a subordinate Metadata server.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Name State Relationship Containers Last Boot
=====
ST1 ONLINE Subordinate 3 Dec. 31, 2002 5:00:34 PM
```

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation mark (') or double quotation mark ("). A blank space is not a valid character. For example:

```
ST1;ONLINE;Subordinate;3;Dec. 31, 2002 5:00:34 PM
```

xml

Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_TankService">
<PROPERTY NAME="CurrentState" TYPE="uint32"><VALUE TYPE="uint32">1
</VALUE></PROPERTY>
<PROPERTY NAME="LastBootUpTime" TYPE="datetime"><VALUE TYPE="datetime">
20031231050034&#46;000000&#45;420</VALUE></PROPERTY>
```

```

<PROPERTY NAME="SystemName" TYPE="string"><VALUE TYPE="string">
DEFAULT_POOL</VALUE>ST1</PROPERTY>
<PROPERTY NAME="IsMaster" TYPE="boolean"><VALUE TYPE="boolean">>false
</VALUE></PROPERTY>
<PROPERTY NAME="NumberOfContainers" TYPE="uint32"><VALUE TYPE="uint32">3
</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>

```

-p Specifies whether to display one page of text at a time or all text at once.

off Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.

on Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.

-hdr

Specifies whether to display the table header.

on Displays the table header. This is the default value.

off Does not display the table header.

-r *number*

Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.

-v Specifies whether to enable verbose mode.

off Disables verbose mode. This is the default value.

on Enables verbose mode.

-l Displays additional information about the Metadata servers.

-s Displays only the name of each Metadata server.

-state

Lists information for only those Metadata servers that match the specified state:

failedinit

Displays information for Metadata servers with the state FAILED INITIALIZATION. A fatal error occurred during initialization and the Metadata server is suspended. The Metadata server remains suspended so you can fix the problem that occurred during initialization.

fullq

Displays information for Metadata servers with the state FULLY QUIESCENT. The Metadata server suspends all current client sessions and prevents new client sessions from being started. All file-data and metadata activity is suspended. The Metadata server flushes all dirty cached pages and commits them to disk. This state is used to produce self-consistent LUN-based backups from a known point-in-time.

initializing

Displays information for Metadata servers with the state INITIALIZING. The Metadata server is running, but has not yet opened its communication ports.

joining

Displays information for Metadata servers with the state JOINING. The Metadata server is joining the cluster.

notadded

Displays information for Metadata servers with the state NOT ADDED. The Metadata server is not part of the cluster. Because the master Metadata server does not know about Metadata servers that are not part of the cluster, this state is only available by logging into that Metadata server.

notrunning

Displays information for Metadata servers with the state NOT RUNNING. The Metadata server is part of the cluster but is not running and cannot perform any functions.

offline

Displays information for Metadata servers with the state OFFLINE. The Metadata server terminates all current client sessions and prevents new client sessions from being started. This state allows all Metadata server I/O to continue. This state restricts the Metadata server from client access.

online Displays information for Metadata servers with the state ONLINE. The Metadata server has fully initialized all of its subsystems, is a member of the cluster, and is serving client requests.

partq Displays information for Metadata servers with the state PARTLY QUIESCENT. The Metadata server prevents new client sessions from being started but allows file I/O to continue for existing clients. The Metadata server suspends clients making file metadata updates (such as changing permissions or creating new files) and system metadata activity. The Metadata server also flushes any dirty metadata pages in the client and Metadata server caches. This state allows you to produce dirty LUN-based backups without requiring clients and applications to stop. You can back up applications using direct I/O (such as database servers) (no caching) in this state.

unknown

Displays information for Metadata servers with the state UNKNOWN.

-role master | sub

Displays only information for those Metadata servers that match the specified role:

master

Displays information for the master Metadata server.

sub

Displays information for subordinate Metadata servers.

server_name

Specifies the name of one or more names of Metadata servers to display. If not specified and if this command is issued from the master issued from the master Metadata server, lists information about all Metadata servers in the cluster.

- Specifies that you want to read the names of one or more of the Metadata servers to display from stdin (for example, - << /work/server_list.txt).

Description**Note:**

If you run this command from an engine hosting a subordinate Metadata server, you can display information about only the local Metadata server. If

you run this command from the engine hosting the master Metadata server, you can display information about any Metadata server.

When you specify multiple parameters in this command, these behaviors apply:

- Flags work as AND coordinators.
- Objects work as OR coordinators.
- When flags and objects are combined, they are listed in this manner:
WHERE flag_1 AND flag_2 AND ... AND flag_n AND (object_1 OR object_2 OR ... OR object_m).
- When flags or objects are not required and not specified, this command searches all possible values unless a default overrides this rule.

For example, if you specify `-state`, `-role`, and two Metadata server names, the search would be performed in the following manner: WHERE state AND role AND (server_name_1 OR server_name_2).

If you do not specify a listing parameter, this command displays the following default information for each Metadata server:

- Name of the Metadata server.
- State of the Metadata server.
- Role of the Metadata server (Master or Subordinate).
- Number of filesets that are hosted by the Metadata server.
- Timestamp of the last boot.

If you specify the `-l` listing parameter, this command displays the following information in addition to the default information for each Metadata server:

- Timestamp of the last status change.
- State to which the Metadata server changed.
- Timestamp of the last target state change.
- Current time.
- Pending software version.

Examples

Display online Metadata servers The following example lists all the attributes of all the Metadata servers with the state ONLINE:

```
tanktool> lsserver -state online
```

Name	State	Relationship	Containers	Last Boot
ST1	ONLINE	Subordinate	3	Dec. 31, 2002 5:00:34 PM
ST2	ONLINE	Master	2	Dec. 31, 2002 5:00:24 PM

Related topics

- “Listing Metadata servers” on page 112
- “Metadata server” on page 33
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314
- “statserver” on page 276
- “Viewing Metadata server details” on page 115

lssnmpmgr

Displays a list of SNMP managers and their attributes.

Syntax



Parameters

`-?` | `-h` | `-help`

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

Description

Prerequisite: You must be logged in to the engine hosting the master Metadata server to run this command.

This command displays the following information for each SNMP manager:

- SNMP manager identifier.
- IP address.
- Port number.
- Version.
- Community.

Examples

Lists the SNMP managers The following example lists the SNMP managers.

```
tanktool> lssnmpmgr
```

ID	IP	Port	Version	Community
1	192.168.0.1	8192	V1	Public
2	192.168.0.2	162	V2C	SNMPMgr2

Related topics

- “addsnmpmgr” on page 149
- “catlog” on page 154
- “clearlog” on page 165
- “Viewing SNMP managers” on page 71
- “lstrapsetting” on page 212
- “rmsnmpmgr” on page 246
- “settrap” on page 257
- “SNMP” on page 42
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314

Istrapsetting

Displays a list of event types that currently generate an SNMP trap.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

Description

Prerequisite: You must be logged in to the engine hosting the master Metadata server to run this command.

This command displays the whether the following event types are enabled:

- Information
- Error
- Warning
- Severe

Examples

Sets the SNMP traps The following example sets SNMP traps for severe and warning events.

```
tanktool> Istrapsetting  
Information      Disabled  
Error            Enabled  
Warning          Enabled  
Severe           Enabled
```

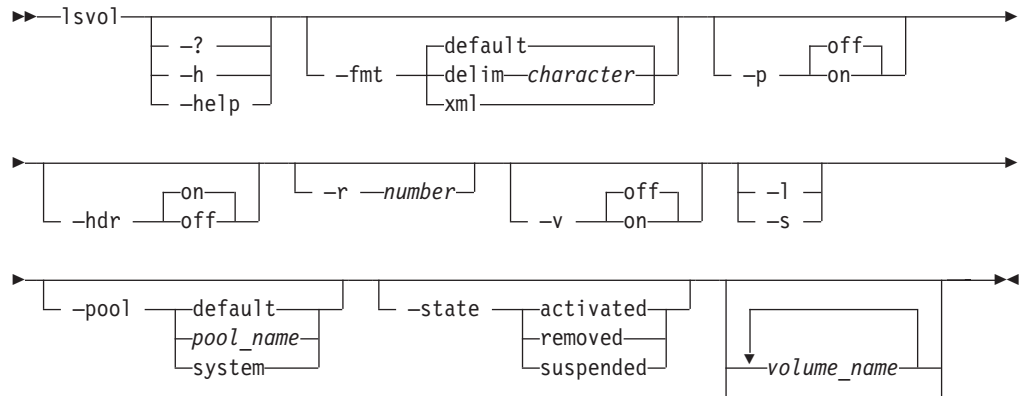
Related topics

- “catlog” on page 154
- “clearlog” on page 165
- “lssnmpmgr” on page 211
- “Modifying SNMP traps” on page 70
- “settrap” on page 257
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314

lsvol

Displays a list of available volumes.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

Name	State	Pool	Size (MB)	Used (MB)	Used (%)
Vol1	Activated	P_Accting	700	350	50

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
Vol1,Activated,P_Accting,700,350,50
```

xml

Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_Volume">
<PROPERTY NAME="StoragePoolName" TYPE="string"><VALUE TYPE="string">
P_Accting</VALUE></PROPERTY>
<PROPERTY NAME="Name" TYPE="string"><VALUE TYPE="string">Vol1</VALUE>
</PROPERTY>
<PROPERTY NAME="State" TYPE="uint32"><VALUE TYPE="uint32">0</VALUE>
</PROPERTY>
<PROPERTY NAME="Size" TYPE="uint64"><VALUE TYPE="uint64">700</VALUE>
</PROPERTY>
<PROPERTY NAME="SizeAllocated" TYPE="uint64"><VALUE TYPE="uint64">350
</VALUE></PROPERTY>
```

```
<PROPERTY NAME="SizeAllocatedPercentage" TYPE="uint16"><VALUE
TYPE="uint16">50</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>
```

- p** Specifies whether to display one page of text at a time or all text at once.
 - off** Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.
 - on** Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.
- hdr** Specifies whether to display the table header.
 - on** Displays the table header. This is the default value.
 - off** Does not display the table header.
- r** *number* Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.
- v** Specifies whether to enable verbose mode.
 - off** Disables verbose mode. This is the default value.
 - on** Enables verbose mode.
- l** Displays additional information about the volumes.
- s** Displays only the name of the volumes.
- pool** *pool_name* Specifies the name of the storage pool to display. You can specify one of the following values:
 - **default** (for the default storage pool)
 - User-defined storage pool name
 - **system** (for the system storage pool)
- state** Displays only information for those storage pools that match the specified type. Possible types are:
 - activated** Displays information for all volumes with a status of Activated.
 - removed** Displays information for all volumes with a status of Being Removed.
 - suspended** Displays information for all volumes with a status of Suspended.
- volume_name* Specifies the names of one or more volumes to list. If unspecified, displays all volume names.
- Specifies that you want to read the names of one or more volumes to list from stdin (for example, **- << /work/vol_list.txt**).

Description

Prerequisite: You must be logged in to the engine hosting the master Metadata server to run this command.

When you specify multiple parameters in this command, these behaviors apply:

- Flags work as AND coordinators.
- Objects work as OR coordinators.
- When flags and objects are combined, they are listed in this manner:
WHERE flag_1 AND flag_2 AND ... AND flag_n AND (object_1 OR object_2 OR ... OR object_m).
- When flags or objects are not required and not specified, this command searches all possible values unless a default overrides this rule.

For example, if you specify `-pool`, `-state`, and two volume names, the search would be performed in the following manner: WHERE pool AND state AND (volume_name_1 OR volume_name_2).

If you do not specify a listing parameter, this command displays the following default information for each volume:

- The name of the volume.
- The state of the volume.
- The size of the volume, in MB.

Note: When the Metadata server creates a volume, it uses some space for its internal bookkeeping (including the disk label). This command reports the size of the volume as the amount of space that can be used to store data, which does not include the space used for bookkeeping.

- The amount of space being used, in MB.
- The percent of space being used.

If you specify the `-l` listing parameter, this command displays the following information in addition to the default information for each volume:

- The operating-system device path.
- The SAN File System volume ID.
- The volume description.

Examples

Display a list of volumes The following example lists all the attributes of all the volumes:

```
tanktool> lsvol -l
```

Name	State	Pool	Size (MB)	Used (MB)	Used (%)
Vol1	Activated	P_Accting	700	350	50
Vol2	Being Removed	P_Accting	700	175	25

Device Path	Vol ID	Description
/dev/hrvpatha	912478	Department 1
/dev/hrvpathb	807890	Department 2

Related topics

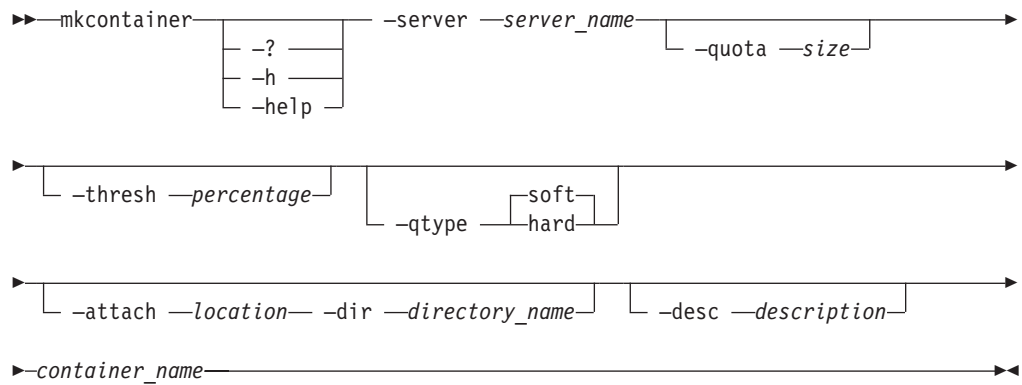
- “activatevol” on page 148
- “chvol” on page 163
- “Listing volumes” on page 131
- “Listing volumes in a storage pool” on page 125
- “mkvol” on page 227
- “reportvolfiles” on page 232
- “rmvol” on page 247

- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314
- “suspendvol” on page 285
- “Viewing volume details” on page 135
- “Viewing volume settings” on page 135
- “Volumes” on page 49

mkcontainer

Creates a new fileset (container).

Syntax



Parameters

-? | **-h** | **-help**

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-server *server_name*

Specifies the Metadata server to host the fileset. This name can be up to 32 characters in length.

Note: This Metadata server can be in any state as long as it is part of the cluster.

-quota *size*

Specifies the maximum size limit, in megabytes, for the specified fileset that, when exceeded, causes the Metadata server to generate an alert. You can specify a number from 0 to 1 073 741 824 MB (1 PB). The default size is 0.

If not specified or set to 0, there is no quota size limit for this fileset, and alerts are not sent.

-thresh *percentage*

Specifies the maximum percentage (alert threshold) of the specified quota size for this fileset that, when exceeded, causes the Metadata server to generate an alert. You can specify a value between 0 and 100. The default alert percentage is 80.

If the **-thresh** parameter is set to 0 or if the quota size is not specified, no alerts are generated.

-qtype **hard** | **soft**

Specifies the quota type for the fileset. You can specify one of the following values:

hard A hard quota produces a log message and potential alert when the quota is met, and denies requests for more space.

soft A soft quota produces a log message and potential alert when the quota size is exceeded, but grants requests for more space.

This is the default type.

-attach *location*

Specifies the existing directory path (relative to the root of the global namespace and without the / prefix) at which to attach the fileset. This directory *must* already exist.

The root of the global namespace must be included in the directory path. For example, to attach the fileset to the /homes location, specify `-attach sanfs/homes`. To attach the fileset directly to the root of the global namespace, use `-attach sanfs`.

Use only forward slashes (/) in the directory path for delimiters.

Note: If the `-attach` parameter is not specified, the fileset will not be attached to the global namespace.

-dir *directory_name*

Specifies the directory name for the root of the fileset directory tree. The directory name *must not* already exist. This directory can be up to 256 characters in length.

The directory must not contain backslash (\) or forward slash (/) characters as delimiters.

The attach point (*location/directory_name*) appears as a drive and directory on Windows clients or as a file system on UNIX-based clients. It is recommended that you map the name of the directory to its associated fileset by using the same name as the fileset plus an additional suffix to distinguish it as the directory and not the fileset. For example, if you want to attach the fileset named `work` to the root of the global namespace, you would specify `"sanfs"` for the *location* and specify `"work_ap"` for the *directory_name*. On a Windows client, the `"work_ap"` directory would appear under the `"sanfs"` drive.

-desc *description*

Specifies a description for the fileset. The description must be enclosed in matching single (') or double (") quotation marks if it contains any blank characters. This description can be up to 256 characters in length.

The default is an empty string.

container_name

Specifies the name to assign to the new fileset (or container). This name can be up to 256 characters in length.

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Note: You must attach the fileset to the global namespace using the `-attach` and `-dir` parameters in this command or the `attachcontainer` command before clients can access the fileset.

Newly created filesets have permissions set to the following:

- 555 permissions when viewed from UNIX-based clients.
- Read control (to Everyone) and Write DAC/Owner (for administrator or root clients) from Windows-based clients.

Examples

Create filesets The following example creates a fileset named `cnt_A`, assigns the fileset to the ST1 Metadata server, sets the quota to 1 000 MB, specifies to send alerts when the quota reaches 70%, and attaches the fileset to the `sanfs/homes` directory.

```
tanktool> mkcontainer -server ST1 -quota 1000 -thresh 70 -qtype soft
-attach sanfs/homes -dir cntA_ap cnt_A
Container "cnt_A" created.
```

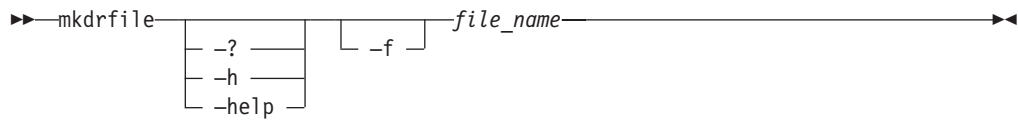
Related topics

- “attachcontainer” on page 151
- “chcontainer” on page 159
- “Filesets” on page 24
- “Creating a fileset” on page 104
- “detachcontainer” on page 168
- “lscontainer” on page 183
- “Managing alerts and logs” on page 67
- “rmcontainer” on page 239
- “Standard format parameters” on page 312

mkdrfile

Creates a new system-metadata disaster-recovery dump file.

Syntax



Parameters

-? | **-h** | **-help**

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-f Overwrites the system-metadata disaster-recovery dump file if a file with the specified name already exists.

file_name

Specifies the name of the new system-metadata disaster-recovery dump file. This name can be up to 250 characters in length.

Description

Prerequisites:

1. You must have Backup, Operator, or Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

This command is used in conjunction with the **builddrsript** command to create a set of recovery scripts that are used to recreate the system metadata in the event of a disaster. The **builddrsript** command is run against the system-metadata disaster-recovery file created by the **mkdrfile** command.

Examples

Create a dump file The following example creates a disaster-recovery dump file named "dr1."

```
tanktool> mkdrfile dr1  
Disaster recovery file "dr1" was created successfully.
```

Related topics

- "builddrsript" on page 153
- "lsdrfile" on page 187
- "rmdrfile" on page 241
- "Standard format parameters" on page 312
- "Creating a recovery file" on page 87

mkimage

Creates a near-instantaneous FlashCopy image of the file layout and contents of the specified fileset and stores it in that fileset.

Syntax

```
▶▶ mkimage container container_name dir directory_name image_name
    -? -h -help
    -f -desc description
```

Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-container *container_name*

Specifies the name of the fileset for which to create a FlashCopy image (or container).

-dir *directory_name*

Specifies the file-system directory under which you want to access the FlashCopy image. This directory can be up to 256 characters in length and must not contain backslash (\) or forward slash (/) characters as delimiters.

The directory name is created under the `.flashcopy` directory in the fileset. The full path of any file in the FlashCopy image is under the directory `attach_point/.flashcopy/directory`, where `attach_point` is the location to which the fileset attaches in the global namespace (for example, `sanfs/cnt_A/.flashcopy/fc1dir`).

Note: The `.flashcopy` directory is a hidden directory. Windows-based clients must use special operating-system commands to view this directory.

-f Forces the Metadata server to delete the oldest FlashCopy image if the fileset already contains the maximum number of FlashCopy images. The oldest image is deleted even if the image is currently in use.

Attention: Any open files associated with the oldest image will be lost.

-desc *description*

Specifies a description for the FlashCopy image. The description must be enclosed in matching single (') or double (") quotation marks if it contains any blank characters. This description can be up to 256 characters in length.

The default is an empty string.

image_name

Specifies the name of the FlashCopy image to create. This name must be unique within the fileset and can be up to 256 characters in length.

Note: This name must not contain colons (:).

Description

Prerequisites:

1. You must have Backup, Operator, or Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

This command does not create FlashCopy images of nested filesets. You must create a FlashCopy image for each fileset in the hierarchy.

After you create a FlashCopy image, you cannot modify the name, description, or directory.

Until the **mkimage** command completes, you can view files in the fileset but not modify them.

You cannot use the **mkimage** command while the **reverttoimage** command is in progress.

A fileset is limited to 32 FlashCopy images at one time.

Examples

Create a FlashCopy image The following example creates a FlashCopy image for the fileset `cnt_A`.

```
tanktool> mkimage -container cnt_A -dir website cntA_image  
Image "cntA_image" successfully created.
```

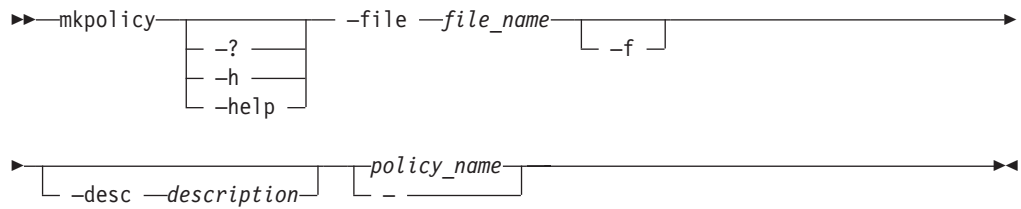
Related topics

- “Backup and restore” on page 5
- “Creating a FlashCopy image” on page 107
- “FlashCopy images” on page 26
- “lsimage” on page 192
- “reverttoimage” on page 237
- “rmimage” on page 242
- “Standard format parameters” on page 312

mkpolicy

Creates a policy.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-file *file_name*

Specifies the fully qualified path and name of the file that contains the set of rules to be included in the new policy. The maximum size of this file is 8 KB.

Use the following format to add rules to this file:

```
VERSION 1
```

```
rule 'stgRule1' set stgpool 'pool1' for fileset ('cnt_A')
rule 'stgRule2' set stgpool 'pool2' where NAME like '%.doc'
rule 'stgRule3' set stgpool 'pool3' where DAYOFWEEK(CREATION_DATE) == 1
rule 'stgRule4' set stgpool 'pool4' where USER_ID <= 100
```

-f Forces the Metadata server to overwrite an existing policy with the same name as the specified policy.

-desc *description*

Specifies a description for the policy. The description must be enclosed in matching single (') or double (") quotation marks if it contains any blank characters. This description can be up to 256 characters in length.

The default is an empty string.

policy_name

Specifies the names of the policy to create. This name can be up to 256 characters in length.

- Specifies that you want to read the names of the policy to create from stdin (for example, - << /work/policies_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Each policy can be up to 32 KB.

To update the policy rules or attributes through the Administrative command-line interface, you must create a new policy.

Examples

Create a policy The following example creates a policy (*test_policy*) using the rules in the */tmp/my_rules.txt* file:

```
tanktool> mkpolicy -file /tmp/my_rules.txt -desc "Test Policy" test_policy
Policy test_policy created.
```

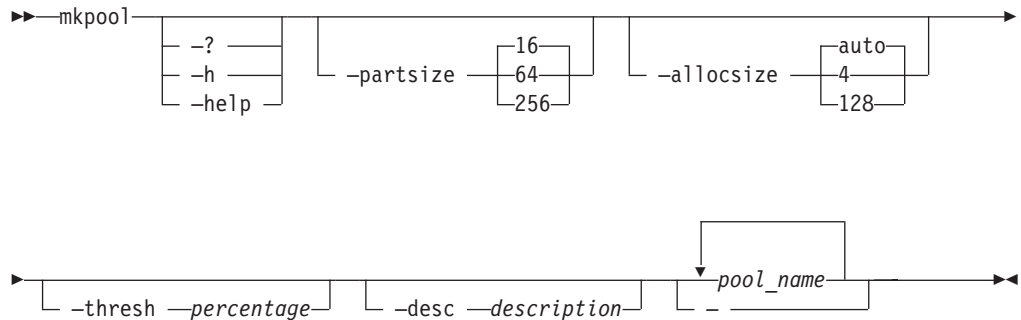
Related topics

- “catpolicy” on page 156
- “Creating a policy” on page 119
- “lspolicy” on page 199
- “Policies and rules” on page 44
- “rmpolicy” on page 244
- “usepolicy” on page 287

mkpool

Defines one or more new user storage pools.

Syntax



Parameters

`-? | -h | -help`

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

`-partsize size`

Specifies the partition size, in megabytes, to use when a fileset allocates space. The partition size can be set to 16, 64, or 256 MB. The default size is 16 MB.

Attention: You cannot change the partition size after it is set.

`-allocsize size`

Specifies the allocation size, in kilobytes, by which a file that resides on a storage pool is to be extended. The allocation size can be set to auto, 4, or 128. If you specify auto, the system sets the size automatically. The default value is auto.

Attention: You cannot change the block size after it is set.

`-thresh percentage`

Specifies a percentage of the storage pool's estimated capacity that, when reached or exceeded, causes the Metadata server to generate an alert. If the capacity drops and then reaches the percentage again, additional alerts are sent.

You can specify a value between 0 and 100. The default alert threshold percentage is 80. If set to 0, no alerts are generated.

`-desc description`

Specifies a description for the storage pool. The description must be enclosed in matching single (') or double (") quotation marks if it contains any blank characters. This description can be up to 256 characters in length.

The default is an empty string.

`pool_name`

Specifies the names of one or more storage pools to create. Each name can be up to 256 characters in length.

- Specifies that you want to read the names of one or more storage pools to create from stdin (for example, `- << /work/stgpools_list.txt`).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

You can create multiple storage pools with the same values for the partition size, allocation size, and alert threshold in a single command; however, you cannot set the name or description for multiple storage pools.

Examples

Create two storage pools The following example creates two storage pools (stgpool1 and stgpool2), sets the partition size on both to 256 MB, and specifies to send alerts when the allocated capacity reaches 70%:

```
tanktool> mkpool -partsize 256 -thresh 70 stgpool1 stgpool2  
Storage pool stgpool1 created.  
Storage pool stgpool2 created.
```

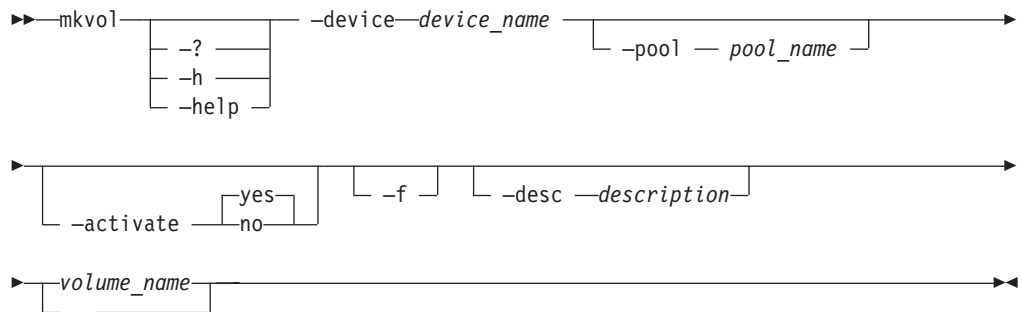
Related topics

- “chpool” on page 161
- “Creating a storage pool” on page 123
- “lspool” on page 202
- “Storage pools” on page 37
- “rmpool” on page 245
- “setdefaultpool” on page 251
- “Standard format parameters” on page 312

mkvol

Adds one or more volumes to a storage pool.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-device *device_name*

Specifies the name of the local device to add to the specified storage pool (for example, */dev/rsdc* on Linux).

-pool *pool_name*

Specifies the name of the storage pool to which to add the new volumes. The storage pool is either a user-defined, default, or system storage pool. If not specified, this command adds the volume to the default storage pool.

-activate

Specifies whether the volume is activated. The Metadata server can allocate new data only on activated volumes. Possible values are:

yes Specifies that the Metadata server can allocate new data on the volume. This is the default value.

no Specifies that the Metadata server cannot allocate new data on the volume.

-f Forces the Metadata server to add the volume and write a new label to the volume if the volume already has a valid SAN File System label.

Note: You can use **-f** only if the volume is not assigned to another storage pool in the same cluster.

-desc *description*

Specifies the description for the volume. The description must be enclosed in matching single (') or double (") quotation marks if it contains any blank characters. This description can be up to 256 characters in length.

The default is an empty string. Specifying this parameter with no description results in a syntax error.

volume_name

Specifies the names of one or more volumes to add. This name must be unique within the fileset and can be up to 256 characters in length.

- Specifies that you want to read the names of one or more volumes to add from stdin (for example, - << /work/vol_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.
3. You must format the applicable LUNs before invoking this command.

Examples

Create a volume The following example creates a volume and adds it to the storage pool named fast:

```
tanktool> mkvol -device /dev/rvpatha -pool fast -desc "fast vol 1" first_vol
Volume first_vol created and added to pool fast.
```

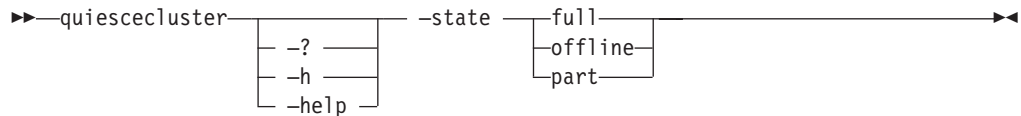
Related topics

- “Adding volumes to a storage pool” on page 130
- “activatevol” on page 148
- “chvol” on page 163
- “lsvol” on page 213
- “reportvolfiles” on page 232
- “rmvol” on page 247
- “Storage pools” on page 37
- “suspendvol” on page 285
- “Volumes” on page 49

quiescecluster

Changes the state of all Metadata servers in the cluster to one of three quiescent states.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-state

Specifies the state to which you want to change the cluster. You can specify one of these states:

full Changes the state to fully quiescent. This parameter flushes all client caches and long-running processes, resulting in no activity on the Metadata servers.

You would put the cluster in this state to produce self-consistent LUN-based backups for a known FlashCopy image.

offline

Changes the state to offline. This parameter terminates all current client sessions and prevents new client sessions from being started, but allows all Metadata server I/O to continue. The offline state is intended to be used as an administrative mode to, for example, reassign a fileset to another Metadata server.

You would put the cluster in this state to restrict clients from accessing the cluster.

part

Changes the state to partly quiescent. This parameter prevents new client sessions from starting. Open files are still accessible, but not new files can be opened or created while in this state.

You would put the cluster in this state to produce *dirty* LUN-based backups without requiring clients and applications to stop, and to allow client applications that use direct I/O (such as database servers) to back up their data.

Description

Prerequisites:

1. You must have Operator or Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

The quiescent states restrict activity on all Metadata servers in the cluster. You would put the Metadata servers in this state in preparation for backup tasks that require the client sessions, file I/O, client metadata, or system metadata to be inactive.

Examples

Change the cluster state The following example changes the cluster state to fully quiescent.

```
tanktool> quiescecluster-state full  
Cluster successfully in fully quiescent state.
```

Related topics

- “Cluster” on page 16
- “Changing active cluster states” on page 82
- “resumecluster” on page 236
- “Standard format parameters” on page 312
- “startcluster” on page 261
- “statcluster” on page 267
- “stopcluster” on page 280

quit

Ends the tanktool session.

Syntax



Parameters

`-?` | `-h` | `-help`

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

Examples

Ends a tanktool session The following example ends the tanktool session:

```
tanktool> quit
shell>
```

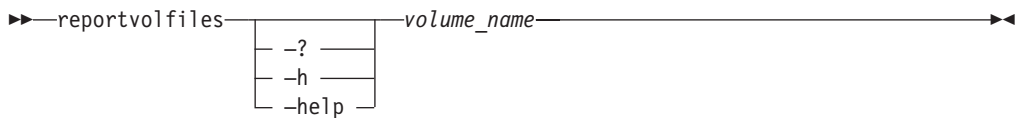
Related topics

- “Command modes” on page 311
- “exit” on page 170
- “Naming guidelines” on page 311
- “Standard format parameters” on page 312
- “Syntax diagram conventions” on page 315
- “tanktool” on page 140
- “User assistance for commands” on page 142

reportvolfiles

Displays a list of files (and their attributes) in the specified volume.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

volume_name

Specifies the name of the volume to report.

Description

Prerequisites:

1. You must have Backup, Operator, or Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Description

This command displays the following information for each file in the specified volume:

- Fileset name
- Fileset directory
- Relative file path
- File name

This information is displayed in the following format:

```
fileset:fileset_directory/relative_file_path/file_name
```

This command is useful for performing a file-based restore operation after a user volume failure because it can list the files in a failed user volume; however, it cannot list files on a failed system volume.

Examples

List files in a volume The following example displays a list of files in volume vol2.

```
tanktool> reportvolfiles vol2
Root:Work/doc/language.doc
NotesR5D:InfoDev/IDPlan.doc
Resources:Personnel/dept/manager/people.txt
```

Related topics

- “Listing files on a volume” on page 131
- “lsvol” on page 213
- “lscontainer” on page 183
- “Standard format parameters” on page 312

- “Volumes” on page 49

resetadmuser

Forces all administrative users to log in again.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

Description

Prerequisite: You must have Administrator privileges to use the command.

Note: This command operates only against the local engine.

To increase Common Information Model (CIM) performance, the Administrative agent retains a copy of the administrative user's password after the LDAP server validates it. This copy is used for a certain period of time to avoid the overhead of going to the LDAP server to authenticate the user for every CIM request. If a user's password has been changed in the LDAP server, use this command to ensure that the user must immediately use the new password rather than continuing to use the Administrative agent's copy.

Examples

Reset the Administrative Users' Passwords The following example resets all administrative user's passwords in the CIM cache:

```
tanktool> resetadmuser
Users reset.
```

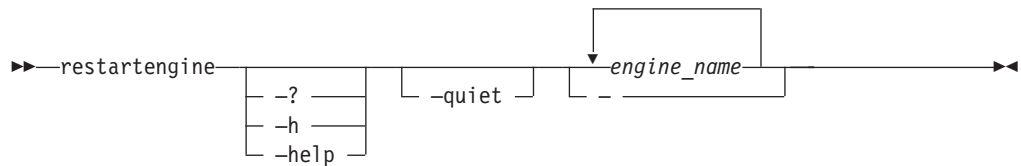
Related topics

- "Administrative security" on page 39
- "Locks and leases" on page 30
- "lsadmuser" on page 172
- "Timing out all user authorizations" on page 128
- "Security" on page 39
- "Standard format parameters" on page 312

restartengine

Reboots one or more storage engines.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off confirmation prompts for this command.

engine_name

Specifies the names of one or more storage engines to restart.

- Specifies that you want to read the names of one or more storage engines to restart from stdin (for example, - << /work/engine_list.txt).

Description

Prerequisite: You must have Operator or Administrator privileges to use the command.

You can stop and restart the local engine, but you cannot start the local engine.

Examples

Restart engines The following example restarts engine ST1.

```
tanktool> restartengine ST1
Are you sure you want to restart engine ST1? [y/n] y
Engine ST1 restarted successfully.
```

Related topics

- “lsengine” on page 189
- “Powering on the engine” on page 96
- “Restarting the engine” on page 97
- “Standard format parameters” on page 312
- “startengine” on page 262
- “statengine” on page 272
- “stopengine” on page 281

resumecluster

Brings all Metadata servers in the cluster to the online state.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

Description

Prerequisites:

1. You must have Operator or Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Examples

Resume the cluster The following example resumes normal activity on the cluster.

```
tanktool> resumecluster  
Cluster successfully returned to online state.
```

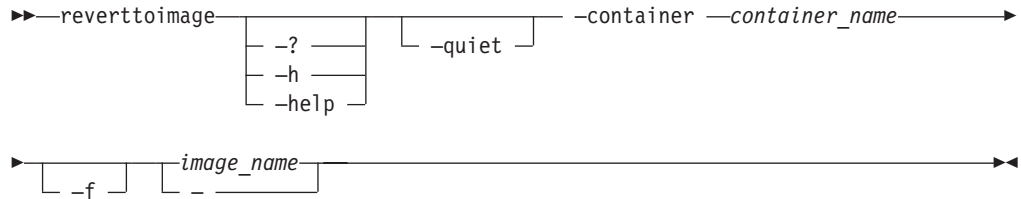
Related topics

- “Cluster” on page 16
- “quiescecluster” on page 229
- “Starting the cluster” on page 84
- “Standard format parameters” on page 312
- “startcluster” on page 261
- “statcluster” on page 267
- “stopcluster” on page 280

reverttoimage

Reverts the current fileset to a specified FlashCopy® image of the file layout and contents.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the revert operation.

-container *container_name*

Specifies the name of the fileset (or container) that you want to revert to a previous FlashCopy image.

-f Forces the Metadata server to revert to the specified FlashCopy image if a client is reading files in the current image. Changes made to files in the fileset since the specified FlashCopy image was created will be lost.

image_name

Specifies the name of the FlashCopy images to which you want to revert.

- Specifies that you want to read the name of the FlashCopy image to which you want to revert from stdin (for example, - << /work/image_list.txt).

Description

Attention: When you revert a fileset to a specified target FlashCopy image, the target FlashCopy image and all FlashCopy images taken between the current fileset and target FlashCopy image are deleted. The target FlashCopy image becomes the primary image for the fileset and no longer appears as an image listed in the .flashcopy directory.

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Unless you specify the **-f** parameter, there **must not** be any client activity in the FlashCopy images being deleted or in the current fileset image.

You cannot revert to a FlashCopy image when nested filesets exist within the fileset. You must manually detach the nested filesets before running the **reverttoimage** command.

The .flashcopy directory is a hidden directory. Windows-based clients must use special operating-system commands to view this directory.

Depending on the age of the specified FlashCopy image and the amount of unique file data in the image tree, the revert operation could result in significant background activity to clean up the file-system objects that are no longer referenced.

Until the **reverttoimage** command completes, you cannot perform any activity on the fileset except the **rmimage** and **reverttoimage** commands. If you issue a second **reverttoimage** command before the first one completes, you can only revert to a FlashCopy image that existed before the FlashCopy image being reverted by the first command.

Because the specified FlashCopy image is deleted after you issue the **reverttoimage** command, it is recommended that you preserve a secondary backup of the fileset before using the command for future use or disaster recovery.

To preserve the fileset hierarchy, you must reattach any nested filesets that you detached before issuing the command when this command completes. It is recommended that you note the hierarchy using the **lscontainer** command before reverting to a FlashCopy image.

Examples

Revert to a previous FlashCopy image The following example reverts the fileset *cnt_A* to the FlashCopy image *cntA_image*.

```
tanktool> reverttoimage -container cnt_A cntA_image
Are you sure you want to revert to FlashCopy image cntA_image? y/n y
Image "cntA_image" successfully restored.
```

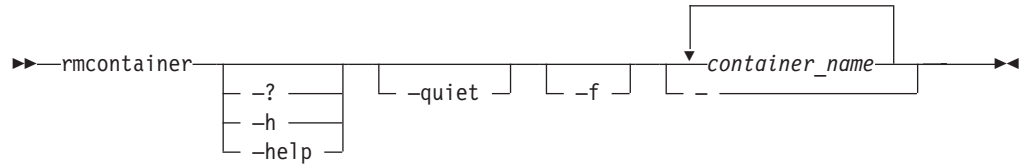
Related topics

- “Backup and restore” on page 5
- “Creating a FlashCopy image” on page 107
- “detachcontainer” on page 168
- “FlashCopy images” on page 26
- “lscontainer” on page 183
- “lsimage” on page 192
- “mkimage” on page 221
- “rmimage” on page 242
- “Standard format parameters” on page 312

rmcontainer

Removes one or more empty, detached filesets (containers) and optionally the files in the filesets, including any FlashCopy® images.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the remove operation.

-f Forces the Metadata server to delete any files and directories before removing the filesets.

container_name

Specifies the names of one or more filesets (or containers) to remove.

- Specifies that you want to read the names of one or more filesets to remove from stdin (for example, **- << /work/cnt_list.txt**).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Unless you specify the **-f** parameter, the fileset *must* be detached and empty before it can be deleted.

You cannot remove the ROOT fileset or a fileset that is reference by the active policy.

Examples

Remove a fileset The following example removes the fileset named cnt_A.

```
tanktool> rmcontainer cnt_A
Are you sure you want to delete container cnt_A? [y/n] y
Container cnt_a removed.
```

Related topics

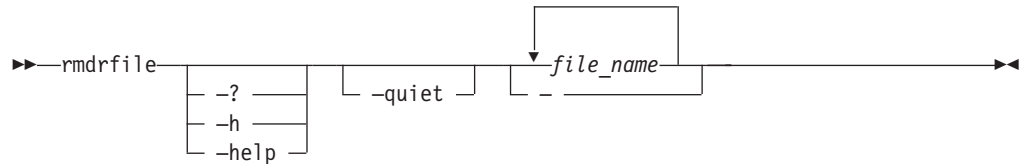
- “attachcontainer” on page 151
- “chcontainer” on page 159
- “Deleting a fileset” on page 105
- “Filesets” on page 24
- “detachcontainer” on page 168

- “lscontainer” on page 183
- “mkcontainer” on page 217
- “Standard format parameters” on page 312

rmdrfile

Deletes an existing system-metadata disaster-recovery dump file.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the change operation

file_name

Specifies the names of one or more system-metadata disaster-recovery dump files to remove.

- Specifies that you want to read the names of one or more system-metadata disaster-recovery dump files to remove from stdin.

Description

Prerequisite: You must have Backup, Operator, or Administrator privileges to use the command.

Note: This command operates only against the local engine.

Examples

Remove a dump file The following example removes the dump file named dr1.

```
tanktool> rmdrfile dr1
Disaster recovery file "dr1" removed.
```

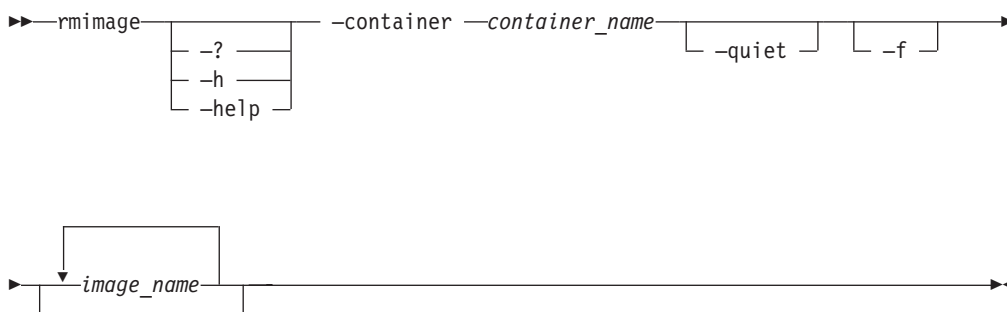
Related topics

- “builddrscript” on page 153
- “lsdrfile” on page 187
- “mkdrfile” on page 220
- “Standard format parameters” on page 312
- “Deleting a recovery file” on page 87

rmimage

Deletes one or more FlashCopy® images for a specific fileset.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-container *container_name*

Specifies the name of the fileset (or container from which to delete one or more FlashCopy images).

-quiet

Turns off the prompt to confirm the delete operation.

-f

Forces the Metadata server to delete a FlashCopy image if one or more files are open in the image tree. Changes that were made to the open files but not saved will be lost.

image_name

Specifies the names of one or more FlashCopy images to delete.

- Specifies that you want to read the names of one or more FlashCopy images to delete from stdin (for example, - << /work/image_list.txt).

Description

Prerequisites:

1. You must have Backup, Operator, or Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Unless you specify the **-f** parameter, you cannot delete a FlashCopy image if there is client activity on that image.

Depending on the age of the FlashCopy image and the amount of unique file data in the image tree, the delete operation might result in significant background activity to clean up the file-system objects that are no longer referenced.

Examples

Delete a FlashCopy image The following example deletes the FlashCopy image (*CA_image*) from the fileset *cnt_A*.

```
tanktool> rmimage -container cnt_A CA_image  
Are you sure you want to delete FlashCopy image CA_image? [y/n] y  
Image "CA_image" successfully deleted.
```

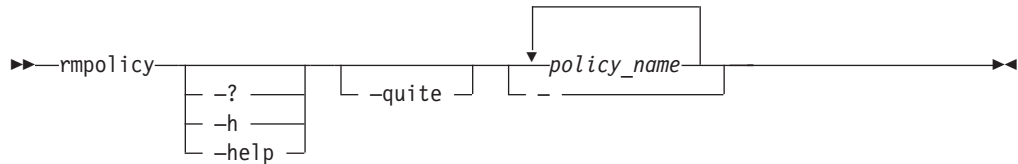
Related topics

- “Backup and restore” on page 5
- “Deleting a FlashCopy image” on page 108
- “FlashCopy images” on page 26
- “lsimage” on page 192
- “mkimage” on page 221
- “reverttoimage” on page 237
- “Standard format parameters” on page 312

rmpolicy

Deletes one or more inactive policies.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the remove operation.

policy_name

Specifies the names of one or more inactive policies to delete.

- Specifies that you want to read the names of one or more inactive policies to delete from stdin (for example, - << /work/policies_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Examples

Remove a policy The following example removes the policy named *test_policy*.

```
tanktool> rmpolicy test_policy
Are you sure you want to delete policy test_policy? [y/n] y
Policy test_policy removed.
```

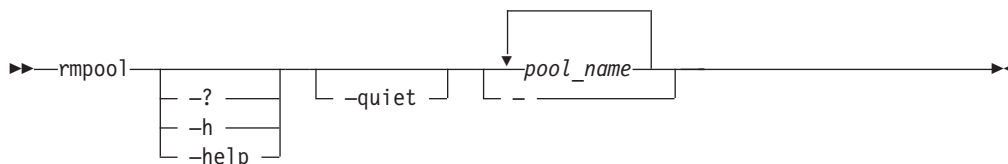
Related topics

- “catpolicy” on page 156
- “Deleting a policy” on page 120
- “lspolicy” on page 199
- “mkpolicy” on page 223
- “Policies and rules” on page 44
- “Standard format parameters” on page 312
- “usepolicy” on page 287

rmpool

Deletes one or more empty, unreferenced storage pools.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the delete operation.

pool_name

Specifies the names of one or more storage pools to delete.

- Specifies that you want to read the names of one or more storage pools to delete from stdin (for example, - << /work/stgpools_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

You cannot delete a storage pool that is not empty. You must remove all volumes from the storage pool before you can delete it.

You cannot delete a storage pool that is referenced by the active policy.

Examples

Remove storage pools The following example removes storage pool stgpool1.

```
tanktool> rmpool stgpool1
Are you sure you want to delete storage pool stgpool1?
Storage pool stgpool1 removed.
```

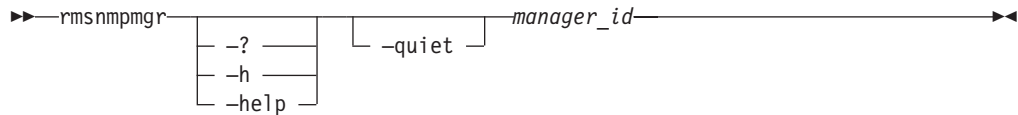
Related topics

- “chpool” on page 161
- “Deleting a storage pool” on page 124
- “lspool” on page 202
- “mkpool” on page 225
- “Storage pools” on page 37
- “setdefaultpool” on page 251
- “Standard format parameters” on page 312

rmsnmpmgr

Removes an SNMP manager (recipient).

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the delete operation.

manager_id

Specifies the number that identifies the SNMP manager (recipient) to delete. Because this number is not static, you must acquire the new number each time a change is made to the list of managers.

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

If all SNMP managers are removed, no SNMP traps are generated.

When you change a disruptive cluster setting, dynamic cluster settings (such as SNMP settings) cannot be modified until you reboot the cluster.

Examples

Removes an SNMP manager The following example removes an SNMP manager.

```
tanktool> rmsnmpmgr 1
Are you sure you want to remove SNMP manager 1? [y/n] y
SNMP manager 1 successfully removed.
```

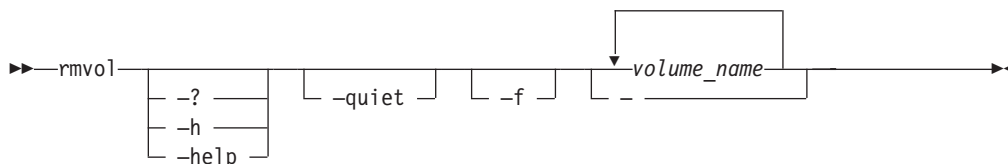
Related topics

- “addsnmpmgr” on page 149
- “catlog” on page 154
- “clearlog” on page 165
- “Deleting SNMP managers” on page 69
- “Issnmpmgr” on page 211
- “Istrapsetting” on page 212
- “settrap” on page 257
- “SNMP” on page 42
- “Standard format parameters” on page 312

rmvol

Removes one or more volumes from a storage pool and redistributes the contents to other volumes in the same storage pool.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the delete operation.

-f Removes all files in the specified volumes and disables draining the files to another volume in the storage pool. If there is a file in the volume, specifying this parameter causes the entire file to be deleted, even if parts of the file reside on multiple volumes.

The **-f** parameter is not allowed when you remove volumes from the system storage pool.

Note: There is no automatic-recovery process when you specify this parameter. You can look at the activity log to get a list of the files for which failures occurred, and then restore those files manually. You can also use the **reportvolfiles** command before invoking the **rmvol** command and review the file list to make sure that the files are not needed or are archived.

volume_name

Specifies the names of one or more volumes to delete.

- Specifies that you want to read the names of one or more volumes to delete from stdin (for example, - << /work/vol_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

You do not need to deactivate a volume before you remove it.

Before removing a volume, SAN File System moves (drains) the contents of the volume across other available volumes in the same storage pool. If the storage pool does not have sufficient space available in other volumes to move all of the data contained in the specified volume, the removal fails and the Metadata server suspends the volume (the Metadata server cannot allocate new data on that volume).

A volume must be empty to be removed from a storage pool. If one or more files cannot be moved to another volume in the same storage pool, you can specify the `-f` parameter to discard those files. If you do not specify the `-f` parameter and the files cannot be removed, the removal fails.

If an error occurs during volume removal, the volume remains in a suspended state so new allocations will not be satisfied from the volume being removed. To activate the volume, use the `activatevol` command.

You must explicitly remove a volume if you want to reuse it. Removing a volume removes the label and frees the device for reuse. If you want to add a device that is in the in-use state, you must remove it using the `-f` option before you can add it as a volume.

Examples

Remove two volumes The following example removes volumes `vol1` and `vol2`.

```
tanktool> rmvol vol1 vol2
Are you sure you want to delete volume vol1? [y/n] y
Volume vol1 removed.
Are you sure you want to delete volume vol2? [y/n] y
Volume vol2 removed.
```

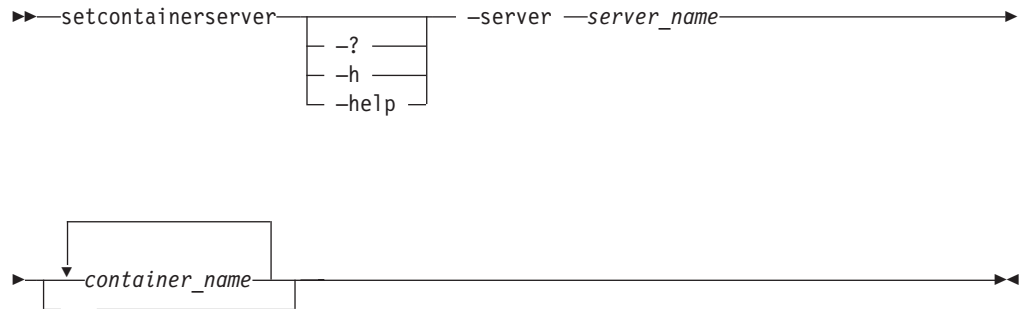
Related topics

- “`activatevol`” on page 148
- “`chvol`” on page 163
- “`lsvol`” on page 213
- “`mkvol`” on page 227
- “`reportvolfiles`” on page 232
- “Removing volumes from a storage pool” on page 133
- “Standard format parameters” on page 312
- “`suspendvol`” on page 285
- “Volumes” on page 49

setcontainerserver

Reassigns an existing fileset (container) to be hosted by a different Metadata server.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-server *server_name*

Specifies the new Metadata server to host the fileset.

Note: If you specify this parameter, the Metadata server currently hosting the fileset must be either offline or not running, and the new host Metadata server must be part of the cluster. The new host Metadata server must be in the online, offline, or not running state.

container_name

Specifies the names of one or more filesets (or containers) to reassign to a new Metadata server.

- Specifies that you want to read the names of one or more filesets to reassign from stdin (for example, - << /work/cnt_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.
3. The current Metadata server must be in one of the following states:
 - Not running
 - Offline
4. The target Metadata server must be part of the cluster and must be in one of the following states:
 - Not running
 - Offline
 - Online

If current Metadata server is online and the target Metadata server is either online or offline, then the cluster is set to the offline state (administrative mode). If

current Metadata server is offline, and the target Metadata server is either online or offline, then the state of the cluster does not change.

Examples

Reassign a fileset The following example reassigns two filesets (*cnt_A* and *cnt_B*) to Metadata server ST1.

```
tanktool> setcontainerserver -server ST1 cnt_A cnt_B
Container cnt_A assigned to server ST1.
Container cnt_B assigned to server ST1.
```

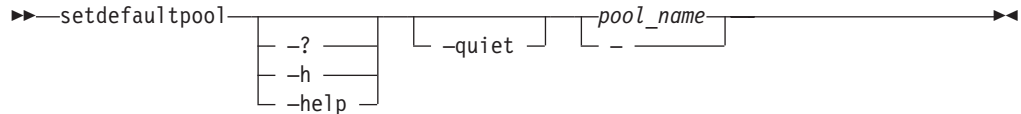
Related topics

- “attachcontainer” on page 151
- “chcontainer” on page 159
- “Filesets” on page 24
- “detachcontainer” on page 168
- “lscontainer” on page 183
- “mkcontainer” on page 217
- “rmcontainer” on page 239
- “Standard format parameters” on page 312

setdefaultpool

Changes a user storage pool to the default storage pool, and changes the previous default storage pool to a regular, nondefault user storage pool.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the change operation.

pool_name

Specifies the name of the storage pool to set as the default.

- Specifies that you want to read the name of the storage pool to set as the default from stdin (for example, - << /work/stgpool_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

You cannot change the SYSTEM storage pool using this command.

Examples

Set the DEFAULT storage pool The following example changes the storage pool named default_stgpool to be the new DEFAULT storage pool:

```
tanktool> setdefaultpool default_stgpool
Are you sure you want to set default_stgpool as the default pool? [y/n] Y
Pool default_stgpool is now the DEFAULT pool.
```

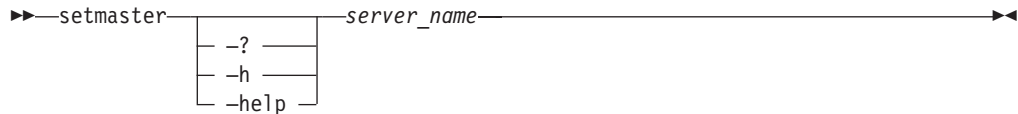
Related topics

- “chvol” on page 163
- “lsvol” on page 213
- “mkpool” on page 225
- “Storage pools” on page 37
- “rmvol” on page 247
- “Setting the default storage pool” on page 125
- “Standard format parameters” on page 312

setmaster

Sets a subordinate Metadata server as the new master Metadata server in the event of an irrecoverable loss of the current master Metadata server.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

server_name

Specifies the name of the subordinate Metadata server to set as the master Metadata server.

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the *subordinate* Metadata server to run this command.

Attention: When you invoke the **setmaster** command, applications will notice a pause in service, which could adversely affect application that are sensitive to timeouts.

Attention: If the master Metadata server is not shut down, you could destroy all of your data. Before using the `setmaster` command, perform the following steps to verify that the master Metadata server is not shut down:

1. From the engine hosting the master Metadata server, perform the following steps:
 - a. Verify that the master Metadata server is *offline* by invoking the `lsserver` command.
 - b. If the master Metadata server is *online*, invoke the `stopservr` command to stop the master Metadata server.
Do not stop the subordinate Metadata servers.
 - c. Verify again that the master Metadata server is *offline* by invoking the `lsserver` and `statcluster -netconfig` commands.
2. Verify that all other Metadata servers are subordinates by invoking the `lsserver` command from each engine hosting the remaining Metadata servers.
3. From the engine hosting the subordinate Metadata server that you want to become the master, perform the following steps:
 - a. Verify that the engine hosting the master Metadata server is shut down by invoking the `lengine` command.
If the engine is running, use the `stopengine` or `stopengine -f` command to shut it down.

Note: If you cannot shut down the engine remotely or through the Administrative CLI, manually power off the engine.
 - b. Verify again that the engine hosting the master Metadata server is shut down by invoking the `lengine` command.

Before stopping the master Metadata server, you should move the filesets that are being served by the master Metadata server to another Metadata server.

Setting a new master Metadata server may affect the clients. If the clients mount to the global namespace using the IP address of the master Metadata server, this address will become unavailable if the client reboots. Therefore, you must reset the IP address to which each client mounts. For AIX client, edit the `/usr/tank/client/config/stclient.conf` file, and set the `server_hot_name` variable.

The `setmaster` command demotes the master Metadata server to a subordinate. It does not drop the Metadata server from the cluster. When you restart the engine, the Metadata server joins the cluster as subordinate.

This command is useful when you want to replace hardware components of or upgrade the engine hosting the master Metadata server.

Examples

Set the master Metadata server The following example verifies that the master Metadata server ST0 is shut down and then sets the subordinate Metadata server ST1 as the master Metadata server.

From Metadata server ST0:

```
tanktool> lsserver
Name State Server Role Containers Last Boot
-----
ST0 Online Master 0 Jul 10, 2003 7:00:17 AM
ST1 Online Subordinate 2 Jul 10, 2003 7:00:24 AM

tanktool> stopservr ST0
```

Server "ST0" stopped gracefully.

```
tanktool> lsserver
Name State      Server Role  Containers  Last Boot
=====
ST0  Not Running  Master      -           -
ST1  Joining     Subordinate  2 Jul 10, 2003 7:00:24 AM
```

```
tanktool> statcluster -netconfig
Could not connect to the server. Please start the server.
```

From Metadata server ST1:

```
tanktool> lsserver
Name State      Server Role  Containers  Last Boot
=====
ST1  Joining     Subordinate  2 Jul 10, 2003 7:00:24 AM
```

```
tanktool> statcluster -netconfig
statcluster must be issued from the admin master server.
```

```
tanktool> stopengine ST0
Are you sure you want to stop engine "ST0"? [y/n] y
Engine "ST0" was shutdown successfully.
```

```
tanktool> lsengine
Name Engine IP      Boot State      Temp  Fans  Voltage
=====
ST1  555.168.10.25  In OS           Normal Normal Normal
ST0  555.168.10.24  Unknown/Power Off -      -      -
```

```
tanktool> setmaster ST1
Server "ST1" is the new master server.
```

```
tanktool> lsserver
Name State      Server Role  Containers  Last Boot
=====
ST1  Online     Master        2 Jul 10, 2003 7:05:12 AM
ST0  Unknown   Subordinate   2 Jul 10, 2003 7:05:12 AM
```

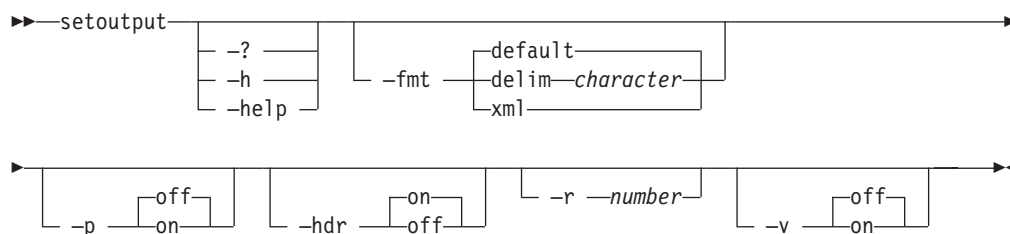
Related topics

- "Changing the master Metadata server" on page 110
- "lsengine" on page 189
- "lsserver" on page 207
- "Metadata server" on page 33
- "Powering off the engine" on page 95
- "Powering on the engine" on page 96
- "Standard format parameters" on page 312
- "startserver" on page 265
- "stopcluster" on page 280
- "stopengine" on page 281

setoutput

Sets the output format for the Administrative command line interface.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Name      Type      Size (GB)  Used (GB)  Used (%)  Alert (%)
=====
DEFAULT   Default   10000      2500       25        80

Volumes   Partition Size (MB)  Description
=====
10        64      Default Storage Pool
```

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation mark (') or double quotation mark ("). A blank space is not a valid character. For example:

```
DEFAULT,Default,10000,2500,25,80,10,64,Default Storage Pool
```

xml Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_StoragePool">
<PROPERTY NAME="Name" TYPE="string"><VALUE>DEFAULT_POOL</VALUE>
</PROPERTY>
<PROPERTY NAME="PoolType" TYPE="uint32"><VALUE>1</VALUE>
</PROPERTY>
<PROPERTY NAME="PartitionSize" TYPE="uint64"><VALUE>16</VALUE>
</PROPERTY>
<PROPERTY NAME="AlertPercentage" TYPE="uint16"><VALUE>80</VALUE>
</PROPERTY>
<PROPERTY NAME="Size" TYPE="uint64"><VALUE>0</VALUE></PROPERTY>
<PROPERTY NAME="SizeAllocated" TYPE="uint64"><VALUE>0</VALUE>
</PROPERTY>
<PROPERTY NAME="SizeAllocatedPercentage" TYPE="uint16"><VALUE>0
</VALUE></PROPERTY>
<PROPERTY NAME="NumberOfVolumes" TYPE="uint32"><VALUE>0</VALUE>
</PROPERTY>
```

```
<PROPERTY NAME="Description" TYPE="string"><VALUE>Default storage pool
</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>
```

-p Specifies whether to display one page of text at a time or all text at once.

off Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.

on Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.

-hdr

Specifies whether to display the table header.

on Displays the table header. This is the default value.

off Does not display the table header.

-r *number*

Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.

-v Specifies whether to enable verbose mode.

off Disables verbose mode. This is the default value.

on Enables verbose mode.

Description

The output format set by this command remains in effect for the duration of the tanktool session or until the options are reset either by using this command or by specifying a output-format parameter as part of a command.

Running this command with no parameters displays the current output settings in the default output format, for example:

```
Paging  Rows  Format  Header  Verbose
=====
off     -      default  on      off
```

Note: The output formats do not apply to help pages.

Examples

Set the output format The following example sets the output format to display in tabular form using a comma as the delimiter without header information

```
tanktool>setoutput -fmt delim , -hdr off
tanktool>lspool -l -type default
DEFAULT,Default,10000,2500,25,80,10,64,Default Storage Pool
```

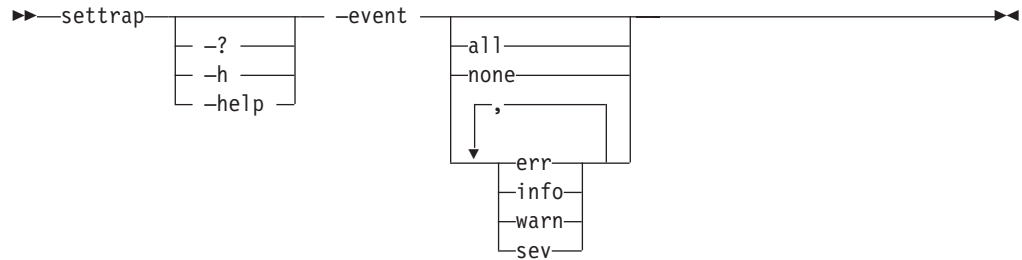
Related topics

- “Command modes” on page 311
- “Standard format parameters” on page 312

settrap

Specifies whether an SNMP trap is generated and sent to all SNMP managers when a specific type of event occurs on the Metadata server.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-event

Specifies the events for which traps are to be sent. You can specify any of the following values:

- all** Generates an SNMP trap when an information, error, warning, or severe event occurs. This parameter cannot be combined with other values.
- none** Turns off SNMP traps for all events. This value cannot be combined with other values.
- info** Generates an SNMP trap when an information event occurs. This value can be combined with any value except all or none. Multiple values must be separated by a comma but no space.
- err** Generates an SNMP trap when an error event occurs. This value can be combined with any value except all or none. Multiple values must be separated by a comma but no space.
- warn** Generates an SNMP trap when a warning event occurs. This value can be combined with any value except all or none. Multiple values must be separated by a comma but no space.
- sev** Generates an SNMP trap when a severe event occurs. This value can be combined with any value except all or none. Multiple values must be separated by a comma but no space.

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Note: To generate traps, you must set the SNMP event types that you want to generate traps and you must add an SNMP manager (using the **addsnmpmgr** command).

SNMP traps of the specified type are generated for all SNMP managers.

When you change a disruptive cluster setting, dynamic cluster settings (such as SNMP settings) cannot be modified until you reboot the cluster.

Examples

Sets the SNMP traps The following example sets SNMP traps for severe and warning events.

```
tanktool> settrap -event sev,warn  
SNMP trap event level was successfully set.
```

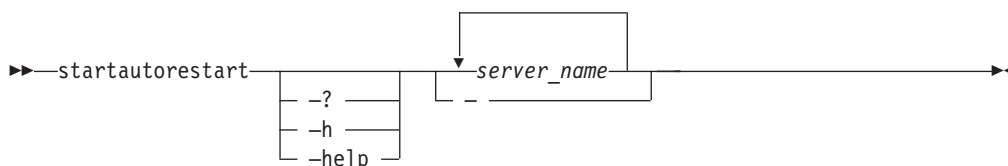
Related topics

- “addsnmpmgr” on page 149
- “catlog” on page 154
- “clearlog” on page 165
- “lssnmpmgr” on page 211
- “lstrapsetting” on page 212
- “rmsnmpmgr” on page 246
- “Setting up SNMP traps” on page 70
- “Standard format parameters” on page 312

startautorestart

Enables the Metadata server to restart automatically if it is down.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

server_name

Specifies the names of one or more Metadata servers to enable them to restart automatically.

- Specifies that you want to read the names of one or more Metadata servers to enable to restart automatically from stdin (for example, - << /work/server_list.txt).

Description

Prerequisite: You must have Administrator privileges to use the command.

Note: If you run this command from an engine hosting a subordinate Metadata server, you can enable the Metadata server restart service on only the local Metadata server. If you run this command from the engine hosting the master Metadata server, you can enable the Metadata server restart service on any Metadata server.

The Metadata server restart service is enabled by default.

When a Metadata server is enabled to restart automatically, an SNMP trap is not sent when the Metadata server is restarted.

Manually stopping a Metadata server or cluster disables the Metadata server restart service for that Metadata server or cluster. Manually starting the Metadata server or cluster reenables the Metadata server restart service for that Metadata server or cluster.

If the Metadata server restart service attempts to restart a Metadata server and fails, the Metadata server restart service tries again to restart the Metadata server, up to the retry limit. After the retry limit, the Metadata server restart service is disabled.

Examples

Enable the Metadata server restart service The following example enables the Metadata server restart service for Metadata server ST1.

```
tanktool> startautorestart ST1
```

The automatic restart service for server ST1 successfully enabled.

Related topics

- “lsautorestart” on page 175
- “Metadata server” on page 33
- “Standard format parameters” on page 312
- “Starting the Metadata server restart service” on page 113
- “stopautorestart” on page 279

startcluster

Starts all Metadata servers in the cluster and brings them to the full online state.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

Description

Prerequisites:

1. You must have Operator or Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

None of the Metadata servers in the cluster can be running when using this command.

When you start all of the Metadata servers in the cluster using the **startcluster** command, all of the subordinate Metadata servers are brought to the online state. When you start the Metadata server using the **startserver** command, the subordinate Metadata servers in the joining state are placed in their previous state after the master Metadata server is up and running.

Note: A message stating that the cluster started successfully does not necessarily mean that the cluster is online.

Examples

Start the cluster The following example starts all Metadata servers in the cluster.

```
tanktool> startcluster  
Cluster started successfully.
```

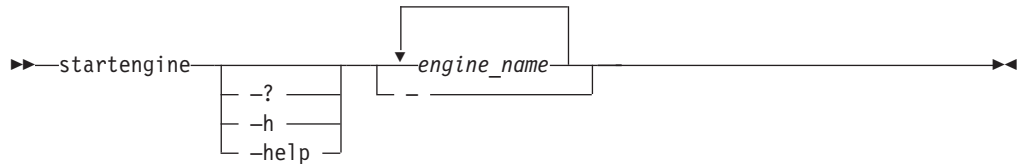
Related topics

- “Cluster” on page 16
- “quiescecluster” on page 229
- “resumecluster” on page 236
- “Standard format parameters” on page 312
- “Starting the cluster” on page 84
- “startserver” on page 265
- “statcluster” on page 267
- “stopcluster” on page 280

startengine

Starts the CPU and operating system on one or more storage engines.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

engine_name

Specifies the name of one or more storage engines to start.

- Specifies that you want to read the names of one or more storage engines to start from stdin (for example, - << /work/engine_list.txt).

Description

Prerequisite: You must have Operator or Administrator privileges to use the command.

You can stop and restart the local engine, but you cannot start the local engine.

Examples

Start an engine. The following example starts engine ST1.

```
tanktool> startengine ST1
```

A start request has been sent to Engine ST1.

Tip: Run the `lsengine` command for current Engine status.

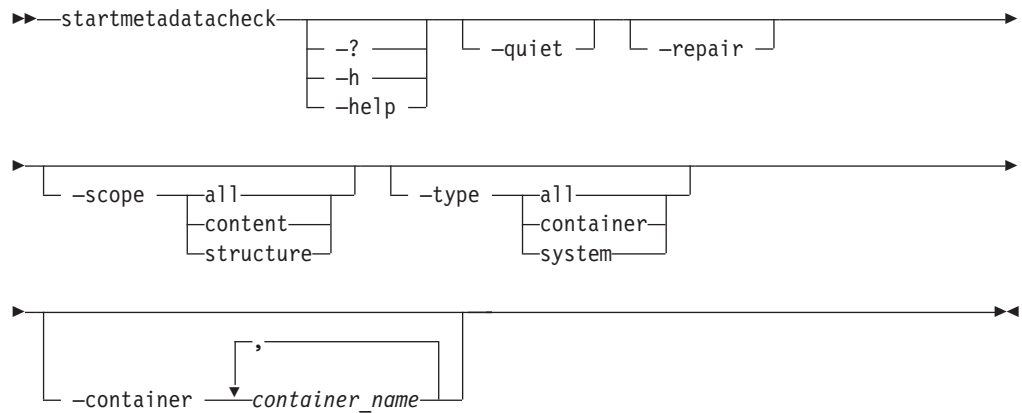
Related topics

- “Engines” on page 23
- “lsengine” on page 189
- “restartengine” on page 235
- “statengine” on page 272
- “stopengine” on page 281
- “Powering on the engine” on page 96
- “Standard format parameters” on page 312

startmetadatascheck

Starts the utility that performs a consistency check on the metadata for the entire system or a set of filesets, generates reports in the cluster log, and optionally repairs inconsistencies in the metadata.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the repair operation.

-repair

Repairs inconsistencies in the metadata and reports the changes in the cluster log. If not specified, repairs are not made.

Note: In some cases, manual intervention might be needed to repair the metadata that requires you to take the cluster offline.

-scope all | content | structure

Specifies the scope of the metadata check. You can specify one of the following values:

all Checks both the metadata content and structure. This is the default value.

content

Checks only the metadata content.

structure

Checks only the metadata structure.

-type all | container | system

Specifies the type of metadata to check. You can specify one of the following values:

all Checks both the system and fileset metadata. This is the default value.

container

Checks only the fileset (or container) metadata.

system

Checks only the system metadata.

- container*container_name*

Specifies the names of one or more filesets (or container) to check, separated by a comma but no spaces. (Do not include a space after the comma.) If not specified, all filesets are checked.

Note: You must specify the **-type container** parameter with this parameter to perform a check on a limited set of filesets. You cannot use this parameter with the **-type all** or **-type system** parameters.

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

This command is useful when you suspect inconsistencies in the metadata or after a major restoration of the system.

You must have Administrator privileges to use all parameters in this command. Users with Backup privileges are not allowed to use the **-repair** parameter.

Note: The metadata checker uses space temporarily in each fileset.

Examples

Check the system-metadata structure The following example checks the system-metadata structure and makes necessary repairs.

```
tanktool> startmetadatabackup -repair -scope structure -type system
Are you sure you want to start a metadata check process? [y/n] y
```

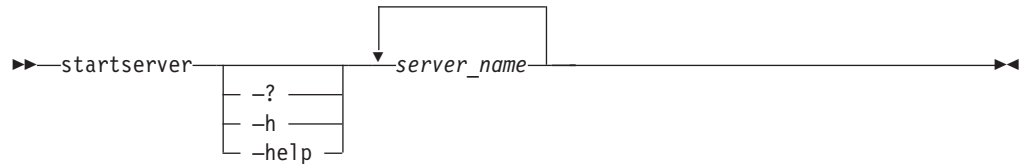
Related topics

- “lscontainer” on page 183
- “stopmetadatabackup” on page 282
- “Checking metadata” on page 112

startserver

Starts the specified Metadata server.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

server_name

Specifies the name of one or more Metadata servers to start. This Metadata server must be part of the cluster.

Description

Prerequisite: You must have Operator or Administrator privileges to use the command.

Note: If you run this command from an engine hosting a subordinate Metadata server, you can only start the local Metadata server. If you run this command from the engine hosting the master Metadata server, you can start any Metadata server.

If the master Metadata server is not running, you cannot start a subordinate Metadata server.

When you start the Metadata server using the **startserver** command, the subordinate Metadata servers in the joining state are placed in their previous state after the master Metadata server is up and running. When you start the cluster using the **startcluster** command, all of the subordinate Metadata servers are brought to the online state.

The state of the Metadata server matches the state of the cluster. Use the **statcluster** command to check on the state of the cluster, or use the **lsserver** command to check on the state of the Metadata server.

When you start a Metadata server using the **startserver** command, the Metadata server restart service on that Metadata server is enabled only if Metadata server restart service was previously in the STANDBY state. If you manually disable the Metadata server restart service and then stop and restart the Metadata server, the Metadata server restart service *will not* become enabled.

Note: A message stating that the Metadata server started successfully does not necessarily mean that the Metadata server is online.

Examples

Start a Metadata servers. The following example starts the Metadata server ST1.

```
tanktool> startserver ST1  
Server ST1 started successfully
```

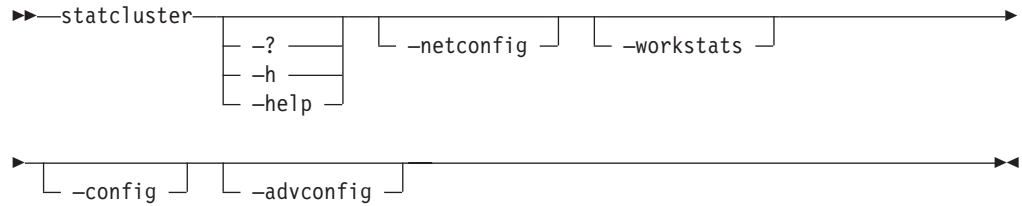
Related topics

- “Metadata server” on page 33
- “lsserver” on page 207
- “statcluster” on page 267
- “statsserver” on page 276
- “stopserver” on page 283
- “Starting a Metadata server” on page 112
- “Standard format parameters” on page 312

statcluster

Displays status, network, workload, and configuration information about the cluster.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-netconfig

Displays network configuration information for the cluster.

-workstats

Displays workload statistics related to the total number of transactions, updates, and buffers for the master Metadata server.

-config

Displays cluster configuration settings that, if changed, do not require that you restart the cluster.

-advconfig

Displays cluster configuration settings that, if changed, require that you restart the cluster.

Description

Prerequisite: You must be logged in to the engine hosting the master Metadata server to run this command.

If you do not specify a parameter, this command displays the following default information:

- Name of the cluster.
- ID of the cluster.
- Current state of the cluster. Possible states are:

Forming

The cluster has a master Metadata server and is in the process of forming. The first time a cluster is formed, its initial state is FORMING.

Fully quiescent

One or more Metadata servers in the cluster are in the FULLY QUIESCENT state.

Not running

One or more Metadata servers in the cluster are not added.

Offline

One or more Metadata servers in the cluster are offline.

Online

All Metadata servers in the cluster are online.

Partly quiescent

One or more Metadata servers in the cluster are in the PARTLY QUIESCENT state.

Unknown

The state of the cluster is not known.

- Target state to which the cluster is switching.
- Timestamp of the last current state change.
- Timestamp of the last state change.
- Number of Metadata servers in the cluster.
- Number of Metadata servers that are actively participating in the cluster (in the offline, online, or quiescent state).
- Committed software version.
- Software version.
- Timestamp when the latest software was committed.
- Software commit status (In Progress or Not In Progress).
- Timestamp of the last installation.

If you specify the **-netconfig** parameter, this command displays the following information in addition to the default information:

- Name of the master Metadata server.
- IP address.
- Cluster port number.
- Heartbeat port number.
- Client-Metadata server port number.
- Administrative port number.

If you specify the **-workstats** parameter, this command displays the following information in addition to the default information:

- Number of system-update transactions involving the creation, modification, and deletion of system objects.
- Number of system-object transactions, involving the creation, modification, and deletion of objects. The system objects include storage pools, filesets, volumes, policies, and engines.
- Current number of buffers, which are used for system metadata activity, that contain data but are available for reuse (clean buffers).
- Current number of buffers, which are used for system metadata activity, that contain data awaiting I/O to disk (dirty buffers).
- Current number of buffers, which are used for system metadata activity, that are not in use (free buffers).
- Current total number of buffers for system metadata activity.

If you specify the **-config** parameter, this command displays the following information in addition to the default information:

- Pool-space reclamation interval, in minutes.

- List of clients with root or administrative privileges.
- Size, in 4-KB pages, of the master Metadata server buffer.
- Size, in pages, of the subordinate Metadata server buffer.
- Metadata server workload-process limit.
- Administrative process limit.

If you specify the **-advconfig** parameter, this command displays the following information in addition to the default information:

- Threshold that specifies the maximum number of missed network heartbeats.
- Maximum number of missed network heartbeats.
- Network heartbeat interval, in milliseconds.
- Cluster timeout, in milliseconds.
- Maximum number of retries to client.
- Client timeout, in milliseconds.
- Client lease period, in seconds.
- Client-lease-period multiplier.
- Metadata server timeout, in milliseconds.

Examples

List network information for the cluster The following example displays the default and network information for the cluster:

```
tanktool> statcluster -netconfig
Name                sanfs
ID                  2802
State               ONLINE
Target State        FULLY QUIESCENT
Last State Change   Sep 24, 2003 3:31:52 PM
Last Target State Change Sep 24, 2003 2:40:20 PM
Servers             4
Active Servers      3
Committed Software Version 1.1.4.2
Software Version    1.00
Last Software Commit Jan 3, 2003 1:40:20 PM
Software Commit Status In Progress
Installation Date   Jan 3, 2003 4:56:59 PM

=====Network Configuration=====
Master Server       ST1
IP                  128.0.0.1
Cluster Port        1737
Heartbeat Port      1738
Client-Server Port  1700
Admin Port          1800
```

Related topics

- “Cluster” on page 16
- “chclusterconfig” on page 157
- “quiescecluster” on page 229
- “resumecluster” on page 236
- “startcluster” on page 261
- “stopcluster” on page 280
- “Standard format parameters” on page 312
- “Viewing cluster details” on page 85
- “Viewing the cluster software version” on page 86

- “Viewing cluster statistics” on page 85
- “Viewing cluster tuning details” on page 85

statcontainer

Displays the number of started and completed transactions for the filesets (containers) being served by the local Metadata server.

Syntax

▶▶—statcontainer—▶▶

Description

Note: This command is run from the shell prompt. It is not run inside of tanktool.

A *transaction* is a work request that is handled by the Metadata server. You would use the number of transactions performed by each fileset on the Metadata server to aid in balancing the workload among all of the Metadata servers in the cluster.

This command displays the following information for each fileset:

- Name of the fileset.
- Number of transactions that have been initiated since the Metadata server started (including those transactions that are initiated during startup). This count includes the transactions that are in progress, stopped, and completed.
- Number of transactions that completed successfully.

Examples

Displaying number of transactions The following example displays the number of started and completed transactions for filesets being served by the local Metadata server.

```
tanktool> statcontainer
```

Name	Transactions Started	Transactions Completed
cnt_A	12765	12751
cnt_B	9478	9465

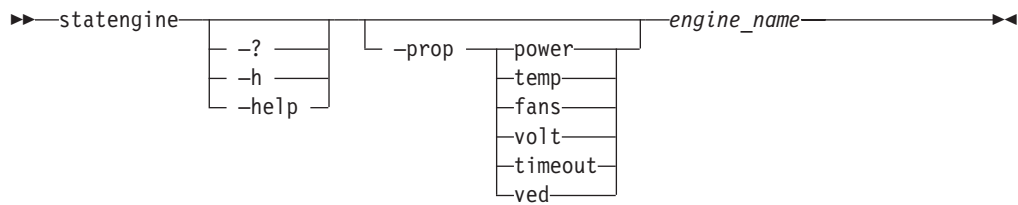
Related topics

- “attachcontainer” on page 151
- “chcontainer” on page 159
- “Filesets” on page 24
- “detachcontainer” on page 168
- “lscontainer” on page 183
- “mkcontainer” on page 217
- “rmcontainer” on page 239
- “Standard format parameters” on page 312
- “Standard listing parameters” on page 314
- “Viewing fileset details” on page 106

statengine

Displays status information about a specific storage engine.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-prop

Specifies the property status to display. Possible values are:

power Displays power status for the specified engine and operating system.

temp Displays temperature status for the components of the specified engine.

fans Displays fan status for the specified engine.

volt Displays voltage status for the specified engine.

timeouts

Displays timeout information for the certain hardware and software functions in specified engine environment.

ved Displays vital engine data for the specified engine.

engine_name

Specifies the name of the storage engine to display.

Description

If you do not specify a property parameter, this command displays the following default information for each engine:

- Engine IP address.
- Engine name.
- Boot state. Possible values are:

Before POST

The engine is powered on but has not started the power-on self test (POST).

Booted Flash

The engine has read the System BIOS but has not started loading the operating system.

Booting OS

The engine has stated but not completed loading the operating system.

CPUs Held in Reset

The engine has been reset after a hardware fault.

In POST

The engine is running the POST.

In OS The engine is running in the normal state.

Stopped in POST (error detected)

The engine is powered on but has not completed the POST due to an error.

Unknown / Power Off

The boot state is unknown. The engine could be powered off.

- Temperature state. Possible values are:

- The temperature thresholds are not set.

Normal

The temperatures of all components in all engines are below the Warning threshold.

Warning

The temperature of one or more engine components are above the Warning threshold.

Error The temperature of one or more engine components is above than the Soft Shutdown threshold.

Unknown

The RSA card could not be accessed.

- Fan state. Possible values are:

Normal

All fans are operating above 15% of its fan-speed capacity.

Warning

One or more fans are operating below 15% of its fan-speed capacity.

Error The RSA card could not be accessed.

- Voltage state. Possible values are:

- The voltage thresholds are not set.

Normal

The voltages of all components in all engines are above the Warning Low Voltage threshold and below the Warning High Voltage threshold.

Warning

The voltage of one or more engine components is below the Warning Low Voltage threshold or above the Warning High Voltage threshold.

Error The RSA card could not be accessed.

If you specify the **-prop power** property parameter, this command displays the following for each engine:

- Engine IP address.
- Engine name.
- Power state (On or Off).
- Number of hours the engine has been powered on.
- Number of times the engine has been restarted. This counter is cleared when the Advanced System Management (ASM) device is cleared to factory defaults.
- Current time on the ASM device's local clock. This time is independent of the time on the engine and is the time used to schedule a power off.

If you specify the **-prop temp** property parameter, this command displays the following for each engine component:

- Engine component. This command displays information for the following components. (Note that the actual component labels and numbers may vary depending on your environment.):

CPU1

CPU2

DASD1

DASD2

Ambient (overall temperature)

- Temperature state per component.
- Current temperature, in degrees Celsius.
- Warning threshold, in degrees Celsius.
- Soft Shutdown threshold, in degrees Celsius.
- Hard Shutdown threshold, in degrees Celsius.

If you specify the **-prop fans** property parameter, this command displays the following for each engine:

- Fan name. This command displays information for each of the eight fans, labeled Fan 1 through Fan 8.
- Fan state.
- Fan speed as a percentage of the maximum capacity speed.

If you specify the **-prop volt** property parameter, this command displays the following for each engine:

- Engine component. This command displays information for the following components. (Note that the actual component labels and numbers may vary depending on your environment.):

VRM 1 (1.5 V)

VRM 2 (1.5 V)

System board (12 V)

System board (5 V)

System board (3.3 V)

System board (2.5 V)

System board (1.5 V)

- Voltage state.
- Voltage.
- Description.
- Warning Low voltage threshold.
- Warning High voltage threshold.

If you specify the **-prop timeouts** property parameter, this command displays the following for each engine:

- POST Watchdog threshold. This is the amount of time the engine will wait for POST to complete before sending an alert and automatically restarting the system. If the threshold is not set, the value is -.
- Loader Watchdog threshold. This is amount of time the engine will wait for the operating system to load before sending an alert and automatically restarting the system. If the threshold is not set, the value is -.

- OS Watchdog timeout. This is the amount of time for the operating system to respond before sending an alert and automatically restarting the system. If the threshold is not set, the value is –.
- OS Watchdog check interval. This is the amount of time between the OS Watchdog checks. If the threshold is not set, the value is –.
- Power Off Delay threshold. The amount of time to wait for the operating system to shut down before powering off the system. If the threshold is disabled, the value is –.

If you specify the **–prop ved** property parameter, this command displays the following for each engine:

- Engine model number.
- Engine serial number.
- Unique universal identifier, which uniquely identifies the engine.
- Firmware revision. This is the revision numbers of the application firmware and startup ROM firmware.
- Logical device’s firmware revision date.
- File names of the application and startup ROM firmware.
- Build identifier of the application and startup ROM firmware.

Examples

Display power status for an engine The following example displays temperature status information for the engine ST1.

```
tanktool> statengine –prop power ST1
```

```
Engine IP          39.47.25.19
Name              ST1
Power State       Off
Power-On Hours    256 hours
Restarts          56
Current ASM Time  Aug 12, 2003 4:59:47 PM
```

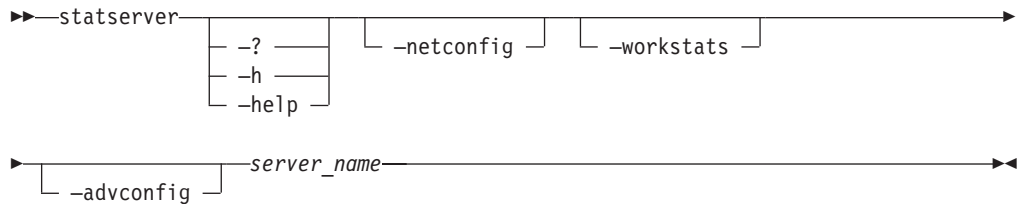
Related topics

- “Engines” on page 23
- “startengine” on page 262
- “lsengine” on page 189
- “stopengine” on page 281
- “Standard format parameters” on page 312
- “Viewing engine statistics” on page 99

statserver

Displays status, configuration, and workload information for a specific Metadata server in the cluster, if issued from the master Metadata server. Lists status, configuration, and workload information for the local Metadata server, if issued from a subordinate Metadata server.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-netconfig

Displays network configuration information for the specified Metadata server.

-workstats

Displays statistical information for the local Metadata server.

-advconfig

Displays advanced settings, which require a reboot when changed, for the specified Metadata server.

server_name

Specifies the Metadata server to display. If you issue this command from a subordinate Metadata server, this must be the name of the local Metadata server.

Description

Note:

If you run this command from an engine hosting a subordinate Metadata server, you can display information about only the local Metadata server. If you run this command from the engine hosting the master Metadata server, you can display information about any Metadata server.

If you do not specify any parameters, the following default statistics are displayed:

- Metadata server name.
- Role of the Metadata server (Master or Subordinate).
- Pending software version.

If you specify the **-netconfig** parameter, the following statistics are displayed in addition to the default statistics:

- Master Metadata server name.
- Metadata server-to-Metadata server transport protocol (UDP)
- Client-to-Metadata server transport protocol (TCP)
- IP address.

- Cluster port number.
- Heartbeat port number.
- Client-server port number.
- Administrative port number.

If you specify the **-workstats** parameter, the following statistics are displayed in addition to the default statistics:

- Number of file-system-update transactions, involving creating, modifying, and deleting system objects.
- Number of file-system transactions, involving reading, creating, modifying, and deleting objects. The system objects include storage pools, containers, volumes, policies, and engines.
- Number of buffers that contain data awaiting I/O to disk (dirty buffers).
- Number of buffers that contain data but are available for reuse (clean buffers).
- Number of buffers that are not in use (free buffers).
- Total number of available buffers.
- Number of session locks held by the engine hosting the Metadata server. Clients acquire session locks to perform file management operations.
- Number of data locks held by the engine hosting the Metadata server. Clients hold data locks to cache data pages and attributes of files and to cache read-only attributes and contents of directories and links.
- Number of byte-range locks held by the engine hosting the Metadata server. Clients use byte-range locks to implement POSIX, SYSV and Berkeley lock system calls.

If you specify the **-advconfig** parameter, the following statistics are displayed in addition to the default statistics:

- File-space reclamation process limit.
- Synchronous commit mode
 - sync** Schedules and buffers metadata changes instead of an immediate write.
 - async** Metadata changes are written to disk immediately.
- List of current profile parameters being used.
- List of current trace parameters being used as output.
- Time interval, in seconds, at which the user and system-time statistics are gathered.

Examples

List information about the Metadata server The following example displays all information about the server:

```
tanktool> statserver -netconfig ST1
```

```
Server ST1
Server Role Subordinate
Most Current Software Version V1.003
=====Network Configuration=====
Master Server ST4
Server-to-Server Transport Protocol UDP
Client-to-Server Transport Protocol TCP
IP 128.0.0.1
Cluster Port 1737
Heartbeat Port 1738
Client-Server Port 1700
Admin Port 1800
```

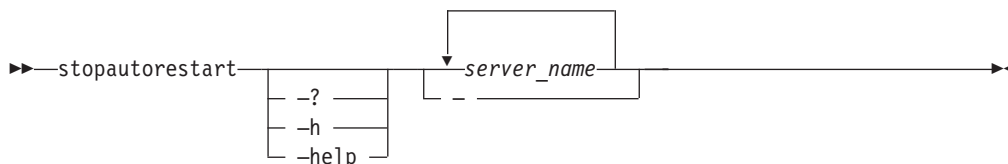
Related topics

- “Metadata server” on page 33
- “Isserver” on page 207
- “Standard format parameters” on page 312
- “Viewing Metadata server networking details” on page 116

stopautorestart

Disables the Metadata server from restart automatically if it is down.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

server_name

Specifies the names of one or more Metadata servers to disable from restarting automatically.

- Specifies that you want to read the names of one or more Metadata servers to enable from restarting automatically from stdin (for example, - << /work/server_list.txt).

Description

Prerequisite:: You must have Administrator privileges to use the command.

Note: If you run this command from an engine hosting a subordinate Metadata server, you can stop the Metadata server restart service on only the local Metadata server. If you run this command from the engine hosting the master Metadata server, you can stop the Metadata server restart service on any Metadata server.

Examples

Disable the automatic-restart service The following example disables the automatic-restart service for Metadata server ST1.

```
tanktool> stopautorestart ST1
```

The automatic restart service for server ST1 successfully disabled.

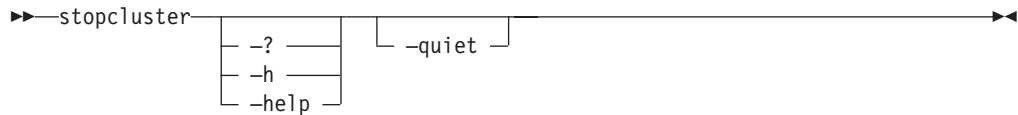
Related topics

- “Isautorestart” on page 175
- “Metadata server” on page 33
- “Standard format parameters” on page 312
- “startautorestart” on page 259

stopcluster

Stops all Metadata servers in the cluster gracefully.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the change operation

Description

Prerequisites:

1. You must have Operator or Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

This command does not shut down the engine on which the specified Metadata server runs.

When you stop the master Metadata server using the **stopserver** command, the subordinate Metadata servers are abruptly moved to the joining state. When you stop all of the Metadata servers in the cluster using the **stopcluster** command, all of Metadata servers are brought down gracefully.

Examples

Stop the cluster The following example stops the cluster gracefully.

```
tanktool> stopcluster
Are you sure you want to shut down the cluster? [y/n] y
Cluster shut down gracefully.
```

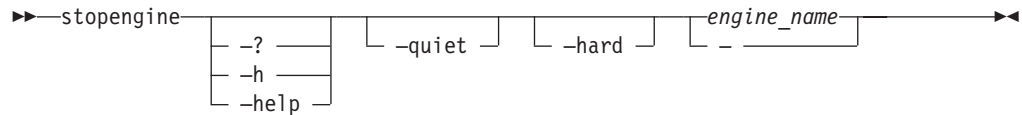
Related topics

- “Cluster” on page 16
- “quiescecluster” on page 229
- “resumecluster” on page 236
- “Standard format parameters” on page 312
- “startcluster” on page 261
- “statcluster” on page 267
- “Stopping the cluster” on page 84

stopengine

Shuts down the operating system and powers off one or more storage engines.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-hard

Powers off the specified engines without first shutting down the operating system. If not specified, this command shuts down the operating system before powering off the storage engines.

-quiet

Turns off confirmation prompts for this command.

engine_name

Specifies the name of the storage engines to power off.

- Specifies that you want to read the names of one or more storage engines to power off from stdin (for example, - << /work/engine_list.txt).

Description

Prerequisite: You must have Operator or Administrator privileges to use the command.

You can stop and restart the local engine, but you cannot start the local engine.

Examples

Power off an engine. The following example shuts down the operating system and then powers off engine ST1.

```
tanktool> stopengine ST1
Are you sure you want to stop engine ST1? [y/n] y
A start request has been sent to Engine ST1.
Tip: Run the lsengine command for current Engine status.
```

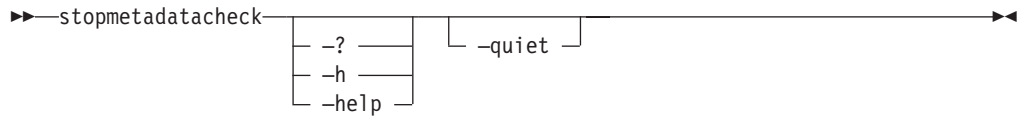
Related topics

- “Engines” on page 23
- “lsengine” on page 189
- “Powering off the engine” on page 95
- “restartengine” on page 235
- “Standard format parameters” on page 312
- “startengine” on page 262
- “statengine” on page 272

stopmetadacheck

Stops the metadata check utility that is currently in progress.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

-quiet

Turns off the prompt to confirm the stop operation.

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Examples

Stop the metadata check process The following example stops the current metadata check process.

```
tanktool> stopmetadacheck
```

```
Are you sure you want to stop the metadata check process? y/n y
```

```
The metadata check is stopping. This may take a few minutes to complete.
```

Related topics

- “Iscontainer” on page 183
- “startmetadacheck” on page 263

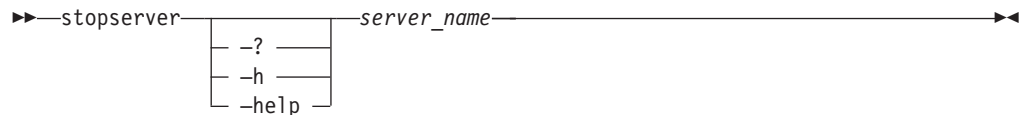
stopserver

Shuts down a subordinate Metadata server gracefully.

Prerequisites:

1. You must have Operator or Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

server_name

Specifies the name of the subordinate Metadata server to shut down. This Metadata server must be part of the cluster.

Description

To use this command, you must be logged into the master Metadata server and the master Metadata server must be running.

This command does not shut down the engine on which the specified Metadata server runs and does not remove the Metadata server from the cluster.

When you stop the master Metadata server using the **stopserver** command, the subordinate Metadata servers are abruptly moved to the JOINING state. When you stop all of the Metadata server in the cluster using the **stopcluster** command, all of Metadata servers are brought down gracefully.

When you stop a Metadata server using the **stopserver** command, the Metadata server restart service on that Metadata server is changed to the STANDBY state only if Metadata server restart service was previously in the ON state.

Examples

Stop a Metadata server The following example stops the Metadata server ST1.

```
tanktool> stopserver ST1
```

Server ST1 shutdown successfully.

Related topics

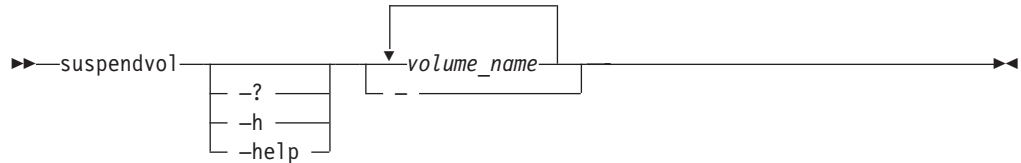
- “Metadata server” on page 33
- “Isserver” on page 207
- “Standard format parameters” on page 312
- “startserver” on page 265
- “statsserver” on page 276
- “stopcluster” on page 280

- “stopengine” on page 281
- “Stopping a Metadata server” on page 114

suspendvol

Suspends one or more volumes so that the Metadata server cannot allocate new data on the volumes.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

volume_name

Specifies the names of one or more volumes to suspend.

- Specifies that you want to read the names of one or more volumes to suspend from stdin (for example, - << /work/vol_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Examples

Suspend a volume The following example suspends activity on volume vol2.

```
tanktool> suspendvol vol2
Volume vol2 suspended.
```

Related topics

- “mkvol” on page 227
- “chvol” on page 163
- “lsvol” on page 213
- “activatevol” on page 148
- “reportvolfiles” on page 232
- “Standard format parameters” on page 312
- “Suspending a volume” on page 133
- “Volumes” on page 49

upgradeclasser

Verifies that all the engines in the cluster have been upgraded to the new version of the software and then initiates the cluster upgrade process.

Syntax

```
▶▶—upgradeclasser—┬───┬───▶▶  
                    └───┴───  
                    -quiet
```

Parameters

-quiet

Turns off the prompt to confirm the upgrade operation.

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

This command only works after all engines have been upgraded to the same software version.

You do not need to stop the cluster to perform the upgrade.

Examples

Upgrade the cluster The following example upgrades the cluster:

```
tanktool> upgradeclasser  
Are you sure you want to upgrade the cluster software? [y/n] Y  
Cluster upgrade successful.
```

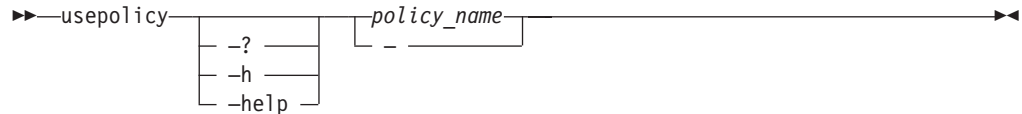
Related topics

- “Cluster” on page 16
- “startcluster” on page 261
- “quiescecluster” on page 229
- “resumecluster” on page 236
- “Standard format parameters” on page 312
- “statcluster” on page 267

usepolicy

Directs the Metadata server to make an existing policy the active policy and applies its rules to all subsequent file creations.

Syntax



Parameters

-? | -h | -help

Displays a detailed description of this command, including syntax, parameter descriptions, and examples. If you specify a help option, all other command options are ignored.

policy_name

Specifies the name of the policy to activate.

- Specifies that you want to read the name of the policy to activate from stdin (for example, - << /work/policies_list.txt).

Description

Prerequisites:

1. You must have Administrator privileges to use the command.
2. You must be logged in to the engine hosting the master Metadata server to run this command.

Examples

Activate a policy The following example activates the policy *test_policy*.

```
tanktool> usepolicy test_policy
Policy test_policy is now the active policy.
```

Related topics

- “Activating a policy” on page 118
- “catpolicy” on page 156
- “lspolicy” on page 199
- “mkpolicy” on page 223
- “Policies and rules” on page 44
- “rmpolicy” on page 244

Client commands

There is a command-line interface for each client operating system that SAN File System supports.

Related topics:

- “AIX-client commands”
- “Windows-client command” on page 307

AIX-client commands

The following table provides a brief description for each AIX-client command.

Note: You must have root privileges to use these commands.

Command	Description
Migration	
migratedata	Migrates data to SAN File System.
Status	
stfsstatus	Displays the version of the file-system drive for the specified virtual client.
Volumes and LUNs	
stfsdisk	Scans the SAN File System for new and removed volumes.
Virtual client setup and removal	
rmstclient	Unmounts the global namespace, removes the SAN File System client, and unloads the file-system driver.
setupstclient	Loads the file-system driver from a standard location, creates the SAN File System client, and mounts the global namespace.
stfsclient	Creates or destroys a virtual client.
stfsdriver	Loads the file-system driver as a kernel extension.
stfsmount	Mounts the global namespace.
stfsumount	Unmounts the global namespace.

should run this command in planning mode first. You can stop the migration process at any point and resume from the last completed file or block (using the **-resume** parameter).

This is the default value.

verify Verifies the integrity of the migrated data using the Message Digest 5 verification algorithm on the contents of the file, as well as verifying consistency of the metadata (such as owner and modification time stamp settings) between the source and destination files.

You can specify more than one phase. For example, to plan, migrate, and verify the data, specify **-phase plan -phase migrate -phase verify**. Although you can specify the phases in any order, this command always estimates the completion time, migrates data, and then verifies the migrated data.

If the **-phase** parameter is not specified, this command runs only the migration phase.

-checkpoint *blocks*

Writes a checkpoint in the log file after each specified number of blocks of file data has completed the migrate phase. (The block size depends on the client platform.) For example, if you specify **-checkpoint 20**, this command makes an entry in the log file each time 20 blocks of file data is migrated. On a platform with a block size of 16 MB, this command writes to the log file after each 2 560-MB of the file data has been migrated. If the process is interrupted, you can resume the migration at the place it left off. If unspecified, the **migratedata** command makes an entry in the log file after each complete file has been migrated. You can resume the migration at the point of the last migrated file.

-resume

Resumes the migration from the last completed block or file (logged in the log file specified by **-log**). If the log file indicates that some files in the source directory are migrated and this parameter is not specified, this command restarts the migration process from the beginning (performs a fresh migration).

-data

Verifies every block of source data (file data and metadata) with the destination data. If not specified, this command verifies only the metadata unless there is a mismatch in the file attributes, in which case this command then verifies the file data.

Note: Verifying all data is very time consuming and can take as long as the migration itself.

-destdir *dest_directory_name*

Specifies the name of the destination directory for the migrated data. The directory can either exist or be a new directory name. IBM recommends that you create the directory before beginning the migration process. If the directory does not exist, this command creates the directory using the default permissions.

source_path

Specifies one or more paths of directories or files to migrate.

Description:

Prerequisite:

1. You must have root privileges on AIX® or Administrative privileges on Windows to use this command.

2. All storage pools, all filesets, and at least one policy must be set up. All activity (from applications, such as database servers and application servers, or users) that modifies data on the source and destination file systems must be stopped and remain stopped to guarantee consistency of the migrated data.
3. The destination directory must exist with correct set of permissions and appropriate storage policies must be configured.

Examples: Migrating data from a client for AIX This example migrates data from the work/capital directory on the client machine to the sanfs/cnt1 directory in the global namespace. A checkpoint is written to the mgrt_capital.log log file each time 20 blocks of file data is migrated.

```
migratedata -log /mgrtlogs/mgrt_capital.log -phase migrate -checkpoint 20  
-destdir /mnt/tank/sanfs/cnt1 work/capital
```

Migrating data from a client for Windows This example migrates data from the C:\Capital directory on the client machine to the sanfs\cnt1 directory in the global namespace. A checkpoint is written to the mgrt_capital.log log file each time 20 blocks of file data is migrated.

```
migratedata -log c:\mgrtlogs\mgrt_capital.log -phase migrate -checkpoint 20  
-destdir t:\cnt1 C:\Capital
```

Related topics:

- “Data migration” on page 22
- Chapter 4, “Migrating data”, on page 63

rmstclient

Unmounts the global namespace, removes the virtual client for AIX, and unloads the file system driver from the local client machine.

Syntax:



Parameters:

-prompt

Prompts for required parameters, using values from the configuration file, if available.

-noprompt

Runs silently, using parameters from the configuration file (/usr/tank/client/config/stclient.conf). If a required parameter is not available, the command exists with an error.

Description:

Prerequisite:

1. You must have root privileges to use this command.
2. You must unmount the SAN File System before invoking this command.

Examples: Remove a client for AIX The following example removes the local SAN File System client for AIX without prompting.

```
rmstclient -noprompt
```

Related topics:

- "setupstclient" on page 293
- "stfsmount" on page 302
- "stfsumount" on page 305
- "Clients" on page 11
- "UNIX-based clients" on page 13

setupstclient

Sets the client configuration and saves the configuration, and optionally loads the file-system driver, creates the client for AIX, and mounts the global namespace.

Syntax:



Parameters:

-prompt

Prompts for required parameters, using values from the configuration file, if available.

-noprompt

Runs silently, using parameters from the configuration file (/usr/tank/client/config/stclient.conf). If a required parameter is not available, the command exists with an error.

Description:

Prerequisite: You must have root privileges to use this command.

This command prompts you for information necessary to set the client configuration or retrieves the information from the configuration file.

If you do not specify a parameter, this command run silently using values from the configuration files as defaults and prompts for any required information.

Examples: Setup a client for AIX The following example loads the file-system driver, creates the client, and mounts the global namespace.

setupstclient

Related topics:

- “rmstclient” on page 292
- “Clients” on page 11
- “UNIX-based clients” on page 13

stfsclient

Creates or destroys a virtual client.

Syntax:

```
▶▶ stfsclient -create [client_name] [server_name] [server_IP_address] [:-port] ▶▶
▶ -kmname -kernel_ext_name -converter - 8859-1 [-quiet] ▶▶

or

▶▶ stfsclient -destroy [client_name] -kmname -kernel_ext_name ▶▶
▶ [-quiet] ▶▶
```

Parameters:

-create

Creates a new virtual client.

-destroy

Destroys an existing virtual client.

client_name

Identifies the unique name of the virtual client that you want to create or destroy. The default client name is the host name of the client system.

server_name

Specifies the host name of a Metadata server in the SAN File System. The Metadata server that you specify informs the global namespace image of all other Metadata servers.

This parameter is not required if this is not the first mount for a particular virtual client.

server_IP_address

Specifies the IP address, in dotted decimal notation, of a Metadata server in the SAN File System.

port

Specifies the port number of the specified Metadata server. The default is 1700.

-kmname *kernel_ext_name*

Identifies kernel-extension name of the file-system-driver instance associated with the virtual client.

The file-system driver is loaded as a kernel extension. To identify the instance of the file-system driver, you identify the kernel extension. The kernel-extension name is the same as name and location of the file-system driver that was used to load the driver (for example, /usr/tank/client/bin/stfs for AIX).

-devices

Determines which devices (also called disks or LUNs) that the virtual client

considers as SAN File System volumes. The default is the value of the STFS_DEVICES environment variable or, if that is not set, "-devices=pat=/dev/rhdisk*."

In addition to creating the virtual client, this command discovers which disks, or candidates, are available to the virtual client as volumes and transmits the candidate list to the virtual client. The **-devices** parameter controls the candidates list.

dir=directory

The candidates list is made up of those devices that have device special files in the specified directory (for example: -devices=dir=/dev/stfsdisk).

The easiest way to mount the global namespace is to specify -devices=pat=/dev/rhdisk*, which looks at every SCSI-disk-like device in the system and whatever looks like a SAN File System disk is accessed when the Metadata server refers to that disk's SAN File System disk identifier.

If you want the client to be more selective about what disks it considers available, you can create a /dev/stfsdisk directory, put device-special files (or symbolic links) for your candidates in it, and just let -devices=dir=/dev/stfsdisk default.

pat=pattern

The candidates list is made up of those devices that have device-special files whose file specifications match the specified pattern. You can use * wildcards in the last (filename) component but not in the directory components (for example, -devices=pat=/dev/rhdisk*).

none The candidates list is empty. Use this value when you want to establish the candidate list with a separate command, perhaps using a selection method more sophisticated than the stfsclient command offers.

-quiet

Specifies not to display informational messages. This parameter does not affect error messages.

Description:

Prerequisite: You must have root privileges to use this command.

This command creates or destroys a virtual client. A *virtual client* is an entity that communicates with a Metadata server and, indirectly, with other SAN File System client. In this release, only one virtual client is supported per client machine. The terms virtual client and client can be used interchangeably.

A virtual client is associated with exactly one SAN File System. There is one file cache and one set of disk candidates per virtual client. Each virtual client running on the same system is as separate as if it were running on a different system. They share nothing except the file-system drive code that they execute.

A SAN File System virtual client is uniquely identified in the context of its file-system driver, and in the context of its SAN File System, by its client name.

To use the files in a global namespace, the virtual client must have a global namespace image. Creating a global namespace image makes the directory

structure in the global namespace appear in the client's file structure. To create a global namespace image, use the **stfsmount** command.

A client can access and create data that is stored in a global namespace. Each virtual client can access data on multiple images in the same global namespace.

The client considers a file to be one file even if it appears with two different file names in two different global namespace images.

Examples: Create a virtual client The following example creates a virtual client.
stfsclient **-create** **MDS1:1700** **-kmname** **/usr/tank/client/bin/stfs**
-converter **8859-1**

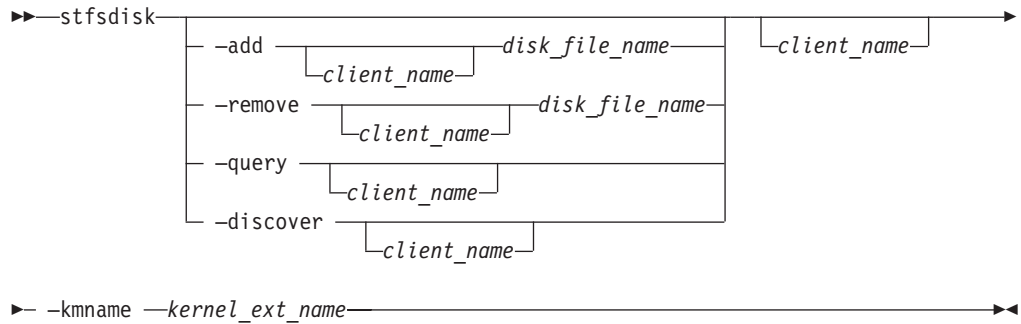
Related topics:

- "Clients" on page 11
- "stfsmount" on page 302
- "stfsdriver" on page 300
- "stfsdisk" on page 297
- "UNIX-based clients" on page 13

stfsdisk

Controls the SAN File System volumes (disks) that a client can access.

Syntax:



Parameters:

-add

Adds the specified disk-specific file name to the disk-candidate list. If the disk-specific file name is already in the list, the client performs a disk discovery procedure on it, updating its database if indicated.

-remove

Removes the specified disk-specific file name from the disk candidate list. If the disk-specific file name is not in the list, this command does nothing, but does not consider it an error.

-query

Displays the list of disk-specific file names in the current disk-candidate list and the status of each. Possible status values are:

ACTIVE

Indicates that the disk is a valid SAN File System user-data volume and is available to the client to perform file reads and write operations.

INACTIVE

Indicates that the disk is not a valid SAN File System user-data volume and is not available to the client to perform file reads and writes. A disk can be inactive if there were I/O errors when the client tried to access the disk, if the disk does not contain a SAN File System label, or if the disk's SAN File System label says it is something other than a user-data volume.

-discover

Rebuild the database of usable disks by going through the current candidate-disk list and attempting to access each disk, determine if it is a valid SAN File System user-data volume, and read its SAN File System global disk ID. If a disk has become accessible or inaccessible, or changed its identity since the last time this disk-discovery procedure was run, the virtual client updates its candidate-disk list accordingly.

This parameter causes the disk-discovery procedure to start. The procedure typically ends before the disk-discovery procedure completes. While the disk-discovery procedures are in progress, any file-system access that would fail because the virtual client cannot find a specific disk will wait until the disk-discovery procedure completes, and then proceed on the basis of the new disk-accessibility information.

disk_file_name

Specifies the file name of the disk to add to or remove from the disk candidate list. This must be a raw disk file such as `/dev/vpath0`, not a logical volume file such as `/dev/hdisk0`.

client_name

Specifies the name of the virtual client whose disk-access you are controlling.

-kmname *kernel_ext_name*

Identifies kernel-extension name of the file-system-driver instance associated with the client.

The file-system driver is loaded as a kernel extension. To identify the instance of the file-system drive, you identify the kernel extension. The kernel-extension name is the same as the name and location of the file-system driver that was used to load the driver (for example, `/usr/tank/client/bin/stfs` for AIX). Note that the kernel extension name might not be unique.

Description:

Prerequisite: You must have root privileges to use this command.

A client reads and writes files by accessing the disks on which the file data resides. To control which disks that a client can access, SAN File System identifies that disk by a SAN File System global disk identifier, and the disk-access subsystem associates that identifier with the name that the AIX operating system uses to identify that disk. The disk-access subsystem maintains a database that correlates global-disk identifiers with AIX device numbers. When the client needs to access a data block of a file, it consults that database.

The disk-access subsystem maintains the database by reading certain disks at certain times and looking for a SAN File System global disk identifier. If it finds the identifier, it determines whether the disk is a SAN File System user-data volume. If the disk is a volume, it adds the disk to its database.

The set of disks that the disk-access subsystem searches is called the *disk-candidate list*. The **stfsclient** command creates the disk-candidate list when it creates the virtual client. You can modify the list using the **-add** and **-remove** parameters.

The candidate-disk list is a list of unique disk-special file names. Because a disk can be referred to by more than one disk-special file name, the list is not strictly a list of unique devices. Actually examining disks and updating the database of valid user-data volumes is separate from maintaining the candidate-disk list.

When you add a disk to the candidate-disk list, the client immediately tries to read it and adds it to the database. But the disk becomes and stays a candidate regardless of the results of that operation.

You can force the client to rescan the entire list of candidate disks using the **-discover** parameter. The client updates its database of user-data volumes according to the results of this discovery, adding and removing disks as necessary. The results of the discovery do not affect the candidate-disk list, however.

Note that device file names can change as the client runs. Such a change has no effect on the client unless something causes a disk-discovery procedure to run. For example, if you add `/dev/rhdisk35` as a candidate disk, and the client successfully identifies it as a SAN File System user-data volume, and then you delete `/dev/rhdisk35`, the client continues accessing that disk as before. The disk

/dev/rhdisk35 continues to be a candidate. But the next time a disk-discovery procedure runs, the candidate will be found invalid and the client will no longer have access the disk.

Examples: Query the disk-candidate list The following example queries disk-candidate list and displays the status of each disk.

```
stfsdisk -query -kmname /usr/tank/client/bin/stfs
```

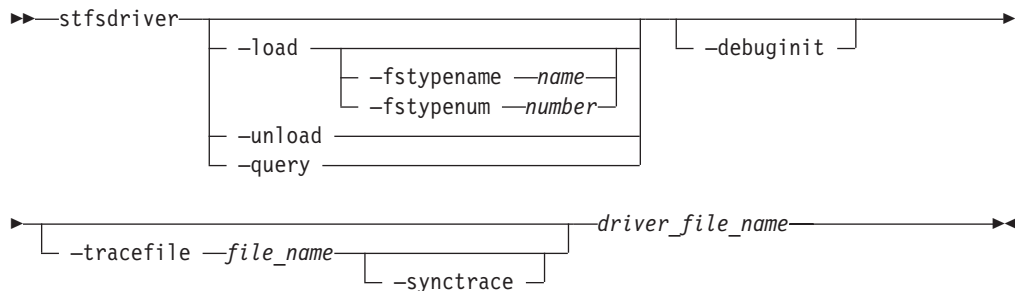
Related topics:

- “Clients” on page 11
- “UNIX-based clients” on page 13
- “stfsclient” on page 294
- “stfsdriver” on page 300

stfsdriver

Loads the file-system driver as a kernel extension.

Syntax:



Parameters:

-load

Loads the kernel extension and create an instance of the file-system driver.

-unload

Unloads the kernel extension and destroys the instance of the file-system driver.

-query

Displays information about the kernel extension matching the specified criteria. For example, you can query the kernel extension ID to use in commands instead of the kernel module name.

-fstypename *name*

Specifies the name of the file-system type to use for the file-system-driver instance. This name relates to a specific file-system-type number. The file `/etc/vfs` maps the file-system-type name to the number.

If you do not specify a file-system-type name or number, the system defaults to the file-system-type named "sanfs". If there is no such type in the `/etc/vfs` file, the system defaults to the file-system-type number 20.

You will use this name to create the virtual client.

-fstypenum *number*

Identifies the number associated with the file-system type for the file-system-driver instance. All mount requests for a file system of this type are routed to this file-system-driver instance.

You would use this parameter only when you load multiple instances of the file-system driver on the same client system.

-debuginit

Enables the file-system driver to issue diagnostic messages of the CONFIG class. Messages in this class are issued only during initialization.

If specified, the file-system driver does not issue diagnostic messages. You can turn the messages on after the file-system driver is running using the `stfsdebug` command.

Note: This parameter is intended for use only by trained service technicians.

-tracefile *file_name*

Specifies that the file-system driver is to write diagnostic information to the specified file.

Note: The specified file must already exist.

-synctrace

Specifies that the file-system drive is to write diagnostic information to the specified trace file synchronously rather than using buffered writes.

driver_file_name

Specifies the name and location of the file-system driver that you want to load, unload, or query. The file name is typically "sanfs".

The file-system driver is loaded as a kernel extension. To identify the instance of the file-system drive, you identify the kernel extension. The kernel-extension name is the same as name and location of the file-system driver that was used to load the driver (for example, /usr/tank/client/bin/sanfs).

Description:

Prerequisite: You must have root privileges to use this command.

This command creates a file-system-driver instance by loading the file-system driver as a kernel extension. This command also unloads or queries the kernel extension.

After loading the file-system driver, you can use the **stfsclient** command to create a virtual client and then use the **stfsmount** command to mount the global namespace.

Examples: Loads the file-system driver The following example loads the file-system driver on a client for AIX.

```
stfsdriver -load /usr/tank/client/bin/sanfs
```

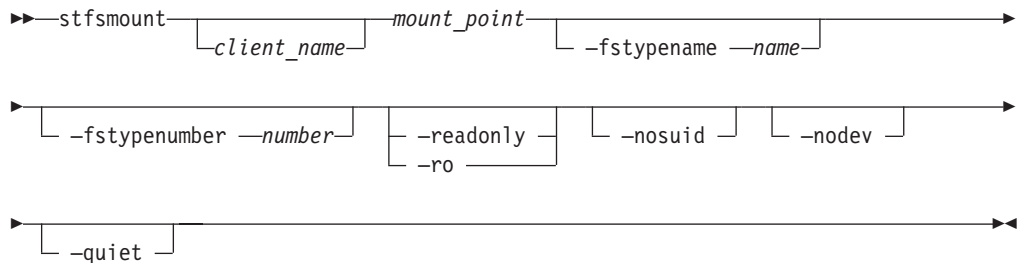
Related topics:

- "Clients" on page 11
- "AIX" on page 78
- "stfsclient" on page 294
- "stfsmount" on page 302
- "UNIX-based clients" on page 13

stfsmount

Mounts the global namespace.

Syntax:



Parameters:

client_name

Identifies the unique name of the virtual client to which you want to mount the global namespace. The client must be up and running. The default client name is the host name of the client system.

mount_point

Specifies the directory associated with the global namespace image that you want to mount.

-fstypename *name*

Specifies the name of the file-system type to use for the file-system-driver instance. This is the same name used to load the file-system driver.

This name relates to a specific file-system-type number. The file `/etc/vfs` maps the file-system-type name to the number.

If you do not specify a file-system-type name or number, the system defaults to the file-system-type named "sanfs". If there is no such type in the `/etc/vfs` file, the system defaults to the file-system-type number 20.

You would use this parameter only when you load multiple instances of the file-system driver on the same client system.

-fstypenumber *number*

The number that identifies the file-system type for the file-system-driver instance. All mount requests for a file system of this type are routed to this file-system-driver instance.

-readonly | **-ro**

Sets the global namespace image to read only. If specified, an attempt to update data or metadata in the global namespace will fail, and an attempt to access a file-system object will not update its access-time attribute.

-nosuid

Disallows any invocation of the `setuid` or `setgid` commands from this file-system image.

-nodev

Disallows any attempts to open device nodes in this file-system image.

-quiet

Specifies not to display informational messages. This parameter does not affect error messages.

Description:

Prerequisite: You must have root privileges to use this command.

This command creates an image of the global namespace on the client system by mounting a directory. The global namespace maintains a list of its directories that are available to the clients. When a client mounts a directory in the global namespace, that directory and its subdirectories become part of the client's directory hierarchy.

Note: This command is used in place of the **mount** command to mount the global namespace.

Before you can mount the global namespace, you must have a virtual client running on the client system. To create the virtual client, use the **stfsclient -create** command.

Remounting the global namespace image is not the same as unmounting the global namespace and then mounting it again. Rather, it changes the attributes of an existing global namespace image, such as changing from read-write to read-only mode. To remount the global namespace image or to see what global namespace images currently exist, use the **stfsmount** command.

To unmount the global namespace, use the **stfsumount** command.

Examples: Mount the global namespace The following example mounts the global namespace.

```
stfsmount mnt/SANFS_MOUNTPT -fstypename sanfs
```

Related topics:

- "Clients" on page 11
- "UNIX-based clients" on page 13
- "AIX" on page 78
- "stfsclient" on page 294
- "stfsumount" on page 305

stfsstatus

Displays the version of the file-system driver for the specified virtual client for AIX.

Syntax:

```
▶▶—stfsstatus— -kmname —kernel_ext_name—————▶▶
```

Parameters:

—**kmname** *kernel_ext_name*

Identifies kernel-extension name of the file-system driver associated with the virtual client.

The file-system driver is loaded as a kernel extension. To identify the instance of the file-system driver, you identify the kernel extension. Each kernel extension has a name, but this name is not unique. This name is usually the file name of the object file from which you loaded the kernel extension (for example, /usr/tank/client/bin/stfs). To determine the kernel-extension name, use the **genkex | grep stfs** command.

Description:

Prerequisite: You must have root privileges to use this command.

After issuing this command, if the client is running, the version of the file-system driver is displayed. If the file-system driver is not loaded, an error message is displayed stating that system could not determine the file-system driver instance.

Examples: Display the file-system-driver version The following example displays the version of the file-system driver for the local client.

```
stfsstatus -kmname /usr/tank/client/bin/stfs
```

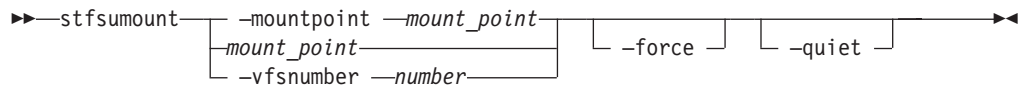
Related topics:

- “Clients” on page 11
- “stfsclient” on page 294
- “stfsdriver” on page 300
- “UNIX-based clients” on page 13

stfsumount

Unmounts the global namespace.

Syntax:



Parameters:

-mountpoint *mount_point* | *mount_point*

Specifies the directory associated with the global namespace image that you want to destroy. This must be the same directory that you specified to mount the global namespace.

If you created multiple global namespace images over the same directory (mount point), this command chooses the most recently created directory.

-vfsnumber *number*

Identifies the virtual file-system (VFS) number associated with the global namespace image that you want to destroy.

In AIX, every global namespace image has a unique VFS number. The **stfsmount** command displays this number when it creates the global namespace image.

Note to reviewers:

What VFS stand for virtual file system?

-force

Unmounts the file system even if it is in use.

-quiet

Specifies not to display informational messages. This parameter does not affect error messages.

Description:

Prerequisite: You must have root privileges to use this command.

This command destroys a global namespace image on the client system. It is used in place of the AIX **umount** command.

After you destroy all global namespace images that are linked to a specific virtual client, you can destroy the virtual client using the **stfsclient -destroy** command.

To see what global namespace images currently exist, use the AIX **mount** command with no parameters.

Examples: Unmount the global namespace The following example unmounts the global namespace on the local client.

```
stfsumount -mountpoint sanfs/cnt1
```

Related topics:

- “Clients” on page 11
- “UNIX-based clients” on page 13
- “AIX” on page 78

- “stfsmount” on page 302

Windows-client command

The following table provides a brief description for the Windows-client command.

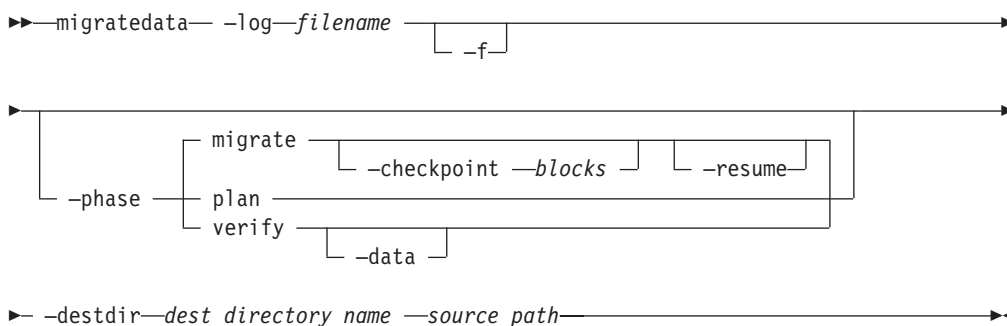
Note: You must have Administrator privileges on the Windows client to use these commands.

Command	Description
Migration	
migratedata	Migrates data to SAN File System.

migratedata

Migrates data to SAN File System.

Syntax:



Parameters:

`-log filename`

Specifies the location of a file in which to log migration activities, warnings, and errors. When used with the `-plan migrate -resume` parameter, the `-log` parameter specifies the log file from which to read information about the last completed block or file.

Attention: You must specify the correct log file with `migrate -resume` and verify that the source and destination directories specified on the command line match those in the log file.

- If you specify an incorrect log file and the `-f` parameter, `-resume` displays a warning and overwrites the target file-system data with wrong information.
- If you specify an incorrect log file, but do not specify the `-f` parameter, this command displays an error and exits.

`-f` Specifies that the migration should continue even if there is an error with a file. If specified with the `-phase migrate` parameter, this command skips any files with errors, but continues with the migration process. If not specified, an error results in the entire migration being stopped before the file that caused the error. You can then restart the migration after fixing the error.

If specified with the `-phase verify` parameter, this command adjusts any missing metadata attributes, such as permissions and times. If there is a mismatch in size, however, this command will not remigrate the file.

`-phase`

Specifies the migration phase to run. Choices include:

plan Gathers information about the available system resources (available memory, number of CPUs, size of the source tree and space available on the destination file system), copies sample files from source directory to estimate transfer rates, and provides an estimated time for the migration of the data set.

migrate

Reads data from source file system and writes the data to the destination file system. Although not required, for large data sets, you should run this command in planning mode first. You can stop the migration process at any point and resume from the last completed file or block (using the `-resume` parameter).

This is the default value.

verify Verifies the integrity of the migrated data using the Message Digest 5 verification algorithm on the contents of the file, as well as verifying consistency of the metadata (such as owner and modification time stamp settings) between the source and destination files.

You can specify more than one phase. For example, to plan, migrate, and verify the data, specify **-phase plan -phase migrate -phase verify**. Although you can specify the phases in any order, this command always estimates the completion time, migrates data, and then verifies the migrated data.

If the **-phase** parameter is not specified, this command runs only the migration phase.

-checkpoint *blocks*

Writes a checkpoint in the log file after each specified number of blocks of file data has completed the migrate phase. (The block size depends on the client platform.) For example, if you specify **-checkpoint 20**, this command makes an entry in the log file each time 20 blocks of file data is migrated. On a platform with a block size of 16 MB, this command writes to the log file after each 2 560-MB of the file data has been migrated. If the process is interrupted, you can resume the migration at the place it left off. If unspecified, the **migratedata** command makes an entry in the log file after each complete file has been migrated. You can resume the migration at the point of the last migrated file.

-resume

Resumes the migration from the last completed block or file (logged in the log file specified by **-log**). If the log file indicates that some files in the source directory are migrated and this parameter is not specified, this command restarts the migration process from the beginning (performs a fresh migration).

-data

Verifies every block of source data (file data and metadata) with the destination data. If not specified, this command verifies only the metadata unless there is a mismatch in the file attributes, in which case this command then verifies the file data.

Note: Verifying all data is very time consuming and can take as long as the migration itself.

-destdir *dest_directory_name*

Specifies the name of the destination directory for the migrated data. The directory can either exist or be a new directory name. IBM recommends that you create the directory before beginning the migration process. If the directory does not exist, this command creates the directory using the default permissions.

source_path

Specifies one or more paths of directories or files to migrate.

Description:

Prerequisite:

1. You must have root privileges on AIX or Administrative privileges on Windows to use this command.
2. All storage pools, all filesets, and at least one policy must be set up. All activity (from applications, such as database servers and application servers, or users)

that modifies data on the source and destination file systems must be stopped and remain stopped to guarantee consistency of the migrated data.

3. The destination directory must exist with correct set of permissions and appropriate storage policies must be configured.

Examples: Migrating data from a client for AIX This example migrates data from the work/capital directory on the client machine to the sanfs/cnt1 directory in the global namespace. A checkpoint is written to the mgrt_capital.log log file each time 20 blocks of file data is migrated.

```
migratedata -log /mgrtlogs/mgrt_capital.log -phase migrate -checkpoint 20  
-destdir /mnt/tank/sanfs/cnt1 work/capital
```

Migrating data from a client for Windows This example migrates data from the C:\Capital directory on the client machine to the sanfs\cnt1 directory in the global namespace. A checkpoint is written to the mgrt_capital.log log file each time 20 blocks of file data is migrated.

```
migratedata -log c:\mgrtlogs\mgrt_capital.log -phase migrate -checkpoint 20  
-destdir t:\cnt1 C:\Capital
```

Related topics:

- “Data migration” on page 22
- Chapter 4, “Migrating data”, on page 63

Command modes

You can work with the Administrative CLI in one of three modes: single-shot, interactive, and script.

Single-shot mode:

If you only want to run a single command, specify the tanktool utility and the command that you want to run from the shell prompt, for example:

```
shell> tanktool lspool -l -type default
DEFAULT,Default,10000,2500,25,80,10,64,Default Storage Pool
shell>
```

Interactive mode:

If you want to run several commands, start a tanktool session using the tanktool utility with no parameters, and then enter each command at the tanktool> prompt, for example:

```
shell> tanktool
tanktool> lspool -l -type default
DEFAULT,Default,10000,2500,25,80,10,64,Default Storage Pool
tanktool> exit
shell>
```

Script mode:

If you want to run a set of commands that you defined in a file, use the tanktool utility with the `-script` parameter, for example:

```
shell> tanktool -script ~/bin/listpools.bat
```

You can add comments to the script file by placing a pound sign (#) in the first column, for example:

```
# This script file lists the pools in the
# SAN File System
lspool -l -type default
```

Note: Output from successful commands routes to stdout. Output from unsuccessful commands route to stderr. If an error occurs while one of the commands in the script is running, the script will exit at the point of failure and return to the system prompt.

Related topics:

- Appendix B, “Commands”, on page 139

Naming guidelines

The following guidelines help you define objects and descriptions used in the Administrative CLI.

Objects:

Use the following guidelines when specifying names for objects:

- You can use alphanumeric characters, dashes (-), underscores (_), and periods (.) in the object names; however, object names cannot start with a dash or underscore character and must contain at least one alphanumeric character.
- The object name must not contain blank spaces.

- Most object names can contain up to 256 characters; the exception is cluster and Metadata server names, which can contain up to 32 characters.
- Object names are case-sensitive.

Descriptions:

Use the following guidelines when specifying descriptions:

- A description can contain up to 256 characters and cannot start with a blank space.
- For administrative commands, if a description contains spaces, enclose the description in single quotation marks (') or double quotation marks (").
- For administrative commands, if a description contains quotation marks, enclose the description in opposite quotation marks (for example, 'This is a pool named \'Foo\>').
- For administrative commands, if a description that is enclosed in quotation marks also contains single quotation marks ('), double quotation marks ("), or asterisks (*) within the string, precede the character with a backslash (for example, "**This is a test**").

Ports:

You can specify a value from 1 024 to 65 535 for port numbers.

Host and domain names:

Machine host names must not include the underscore character (_). Internet standards dictate that domain names conform to the host name requirements described in Internet Official Protocol Standards RFC 952 and RFC 1123.

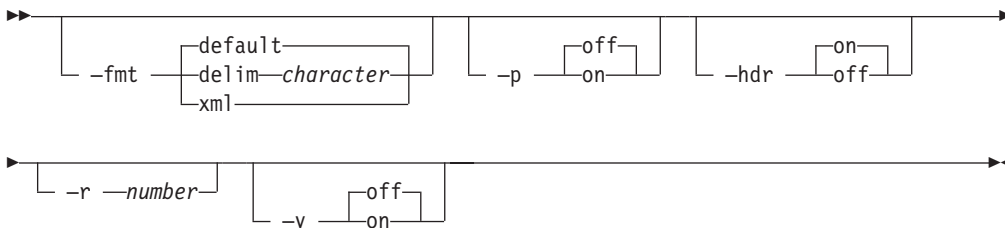
Domain names must contain only letters (upper or lower case) and digits. Domain names can also contain dash characters (-) as long as the dashes are not on the ends of the name.

Standard format parameters

The standard format parameters set the output format of the listing (ls*) commands in the the Administrative CLI. These parameters can be used either in the listing command syntax or in the **setoutput** command.

The format settings remain in effect for the duration of the session or until you reset the parameters either by specifying these parameters in a listing command (those commands that start with ls) or using the **setoutput** command.

Format:



Parameters:

-p Specifies whether to display one page of text at a time or all text at once.

off Displays all text at one time. This is the default value when the **tanktool** command is run in single-shot mode.

on Displays one page of text at time. Pressing any key displays the next page. This is the default value when the **tanktool** command is run in interactive mode.

-r number

Specifies the number of rows per page to display when the **-p** parameter is on. The default is 24 rows. You can specify a value from 1 to 100.

-fmt

Specifies the format of the output. You can specify one of the following values:

default

Specifies to display output in a tabular format using spaces as the delimiter between the columns. This is the default value. For example:

```
Name      Type      Size (GB)  Used (GB)  Used (%)  Alert (%)
=====
DEFAULT   Default   10000     2500      25       80

Volumes   Partition Size (MB)  Description
=====
         10             64  Default Storage Pool
```

delim character

Specifies to display output in a tabular format using the specified character to separate the columns. If you use a shell metacharacter (for example, * or \t) as the delimiting character, enclose the character in single quotation marks (') or double quotation marks ("). A blank space is not a valid character. For example:

```
DEFAULT,Default,10000,2500,25,80,10,64,Default Storage Pool
```

xml Specifies to display output using XML format, for example:

```
<IRETURNVALUE>
<INSTANCE CLASSNAME="STC_StoragePool">
<PROPERTY NAME="Name" TYPE="string"><VALUE>DEFAULT_POOL</VALUE>
</PROPERTY>
<PROPERTY NAME="PoolType" TYPE="uint32"><VALUE>1</VALUE>
</PROPERTY>
<PROPERTY NAME="PartitionSize" TYPE="uint64"><VALUE>16</VALUE>
</PROPERTY>
<PROPERTY NAME="AlertPercentage" TYPE="uint16"><VALUE>80</VALUE>
</PROPERTY>
<PROPERTY NAME="Size" TYPE="uint64"><VALUE>0</VALUE></PROPERTY>
<PROPERTY NAME="SizeAllocated" TYPE="uint64"><VALUE>0</VALUE>
</PROPERTY>
<PROPERTY NAME="SizeAllocatedPercentage" TYPE="uint16"><VALUE>0
</VALUE></PROPERTY>
<PROPERTY NAME="NumberOfVolumes" TYPE="uint32"><VALUE>0</VALUE>
</PROPERTY>
<PROPERTY NAME="Description" TYPE="string"><VALUE>Default storage pool
</VALUE></PROPERTY>
</INSTANCE>
</IRETURNVALUE>
```

-hdr

Specifies whether to display the table header.

on Displays the table header. This is the default value.

off Does not display the table header.

- v Specifies whether to enable verbose mode.
 - off** Disables verbose mode. This is the default value.
 - on** Enables verbose mode.

Related topics:

- “setoutput” on page 255

Standard listing parameters

The standard listing parameters specify whether to display the default, long, or short output for Administrative CLI listing commands. You can specify these parameters in addition to the standard parameters available for the listing commands.

If you do not specify a listing parameter, the default listing displays all objects and the most vital column information, for example:

Name	User Role	Authorization
JohnDoe	Admin	Current
MaryBlack	Backup	Not Current
JimSmith	Operator	Current
TomJones	Monitor	Not Current

The format set using these parameters remains in effect for the duration of the command.

Format:



Parameters:

- s Displays the list of objects with minimal information, for example,

```
Name
=====
JohnDoe
MaryBlack
JimSmith
TomJones
```

- l Displays the list of all objects with detailed information, for example:

Name	User Role	Authorization	Authorization Timeout (Secs)
JohnDoe	Admin	Current	300
MaryBlack	Backup	Not Current	0
JimSmith	Operator	Current	465
TomJones	Monitor	Not Current	0

Related topics:

- “setoutput” on page 255

Syntax diagram conventions

To read syntax diagrams, follow the path of the line. Read from left to right, and top to bottom.

- The ►► symbol indicates the beginning of the syntax diagram.
 - The → symbol at the end of a line indicates that the syntax diagram continues on the next line.
 - The ► symbol at the beginning of a line indicates that the syntax diagram continues from the previous line.
- The →◄ symbol indicates the end of the syntax diagram.

Syntax diagrams use *position* to indicate required, optional, and default values for keywords, variables, and operands:

- On the line (required element)
- Above the line (default element)
- Below the line (optional element)

Abbreviations:

The Administrative command-line interface does not currently support aliases or abbreviations; however, you can use aliases that you set up in the shell environment.

Dash:

A dash (-) indicates that you want to supply parameters from stdin rather than entering parameters. In the following example, the command line gets input from the file /work/myfile:

```
cmd - >> work/myfile
```

In the following example, the command line gets input from the keyboard:

```
cmd - >>
parameter_1
parameter_2
```

Defaults:

Default values are above the main line. If the default is a keyword, it appears only above the main line. You can specify this item or allow it to default. In the following example, the keyword A is the default. You can override it by choosing B or C. You can also specify the default value explicitly.



If an operand has a default value, the operand appears both above and below the main line. A value below the main line indicated that if you specify the operand, you must also specify either the default value or another value shown. If you do not specify an operand, the default value above the main line is used. In the following example, the operand A=* is the default. You can override it by choosing A=C. You can also specify the default value explicitly.



Optional items:

When one or more items are below the main line, all of the items are optional. In the following example, you can choose A, B, C, or nothing at all.

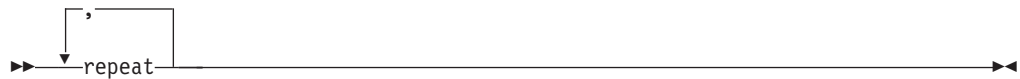


Repeatable items:

An arrow returning to the left means you can repeat the item, for example:



A character or space within the arrow means you must separate repeated items with that character or space, for example:



A stack of items followed by an arrow returning to the left means that you can select more than one item or, in some cases, repeat a single item. In the following example, you can choose any combination of A, B, or C.



Required items:

When a keyword, variable, or operand appears on the main line, you must specify that item. In the following example, you must choose A and B.



When two or more items are in a stack and one of them is on the main line, you must specify one item. In the following example, you must choose A, B, or C.



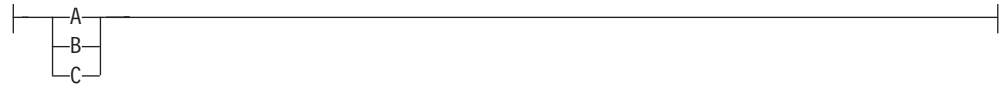
Syntax fragments:

Commands that contain lengthy groups or a section that is used more than once in a command are shown as separate fragments following the main diagram. The fragment name appears between vertical bars in the diagram. The expanded

fragment also appears between vertical bars after the heading with the same fragment name.

▶▶ | The fragment name | ◀◀

The fragment name:



Variables:

Italicized, lowercase elements denote variables. In the following example, you must specify a variable name when you enter the KEYWORD command:

▶▶ —keyword—*variable*— ◀◀

Related topics:

- Appendix B, “Commands”, on page 139

Appendix C. Environment variables

Administrative CLI variables:

The following environment variables modify the behavior of the Administrative CLI:

ADMINDIR = *directory*

Specifies the directory where the administrative binaries and the truststore are located. The default directory is /usr/tank/admin.

SFS_CLI_PASSWDFILE = *directory/file*

Specifies the full-qualified name of the password file that stores the user ID and password information.

SFS_CLI_VERBOSE = *flag*

Turns verbose mode on or off. You can specify a value of ON or OFF.

TANKDIR=*directory*

Specifies the directory in which to save diagnostic results.

Appendix D. Getting help, service, and information

If you need help, service, technical assistance, or just want more information about IBM products, you will find a wide variety of sources available from IBM to assist you.

IBM maintains pages on the World Wide Web where you can get information about IBM products and services and find the latest technical information.

Table 10 lists some of these pages.

Table 10. IBM Web sites for help, services, and information

www.ibm.com/	Main IBM home page
www.ibm.com/storage/	IBM Storage home page
www.ibm.com/storage/support	IBM Support home page

Services available and telephone numbers listed are subject to change without notice.

Software Maintenance

All distributed software licenses include Software Maintenance (software subscription and technical support) for a period of 12 months from the date of acquisition providing a streamlined way to acquire IBM software and assure technical support coverage for all licenses. Extending coverage for a total of three years from date of acquisition may be elected. While your Software Maintenance is in effect, IBM will provide you assistance for your 1) routine, short duration installation and usage (how-to) questions; and 2) code-related questions. IBM provides assistance via telephone and, if available, electronic access, only to your information systems (IS) technical support personnel during the normal business hours (published prime shift hours) of your IBM support center. (This assistance is not available to your end users.) IBM provides Severity 1 assistance 24 hours a day, every day of the year.

Hardware Warranty

For a period of one year, if required, IBM provides repair or exchange service depending on the type of warranty service specified for your machine. An IBM technician will attempt to resolve your problem over the telephone; you must follow IBM's problem determination and resolution procedures. Scheduling of service will depend upon the time of your call and is subject to parts availability. Service levels are response time objectives and are not guaranteed. The specified level of warranty service may not be available in all worldwide locations; additional charges may apply outside IBM's normal service area. Contact your local IBM representative or your reseller for country and location specific information.

IBM On-Site Repair (IOR) IOR, 24 hours a day, 7 days a week, same-day response.

IBM will provide repair services for the failing machine at your location and verify its operation. You must provide suitable working area to allow disassembly and

reassembly of the IBM machine. The area must be clean, well lit, and suitable for the purpose. Depending on the proximity of the master console to the SAN File System cluster, you may also need to provide a keyboard, monitor, and mouse for attachment to a SAN File System engine.

Service support

With the original purchase of an IBM hardware product, you have access to extensive support coverage. During the IBM hardware product warranty period, you may call the IBM Support Center (1-800-426-7378 in the U.S.) for hardware product assistance covered under the terms of the IBM hardware warranty. See Table 10 on page 321 for Support Center telephone numbers in other countries.

The following services are available during the warranty period:

- Problem determination - Trained personnel are available to assist you with determining if you have a hardware problem and deciding what action is necessary to fix the problem.
- IBM hardware repair - If the problem is determined to be caused by IBM hardware under warranty, trained service personnel are available to provide the applicable level of service.
- Engineering change management - Occasionally, there might be changes that are required after a product has been sold. IBM or your reseller, if authorized by IBM, will make Engineering Changes (ECs) available that apply to your hardware.

Be sure to retain your proof of purchase to obtain warranty service.

Please have the following information ready when you call:

- Machine Type and Model
- Serial numbers of your IBM hardware products
- Description of the problem
- Exact wording of any error messages
- Hardware and software configuration information

If possible, be at your computer when you call.

A compatible monitor, keyboard, and mouse may be required for some service activities.

The following items are not covered:

- Replacement or use of non-IBM parts or nonwarranted IBM parts

Note: All warranted parts contain a 7-character identification in the format IBM FRU XXXXXXX.

- Identification of software problem sources
- Configuration of BIOS as part of an installation or upgrade
- Changes, modifications, or upgrades to device drivers
- Installation and maintenance of network operating systems (NOS)
- Installation and maintenance of application programs

Refer to your IBM hardware warranty for a full explanation of IBM's warranty terms.

Before you call for service

Some problems can be solved without outside assistance, by using the online help, by looking in the online or printed documentation that comes with the SAN File System, or by consulting the support Web page noted in Table 10 on page 321. Also, be sure to read the information in any README files and release notes that come with the SAN File System.

Getting help online

Be sure to visit the support page for the SAN File System, complete with FAQs, parts information, technical hints and tips, technical publications, and downloadable files, if applicable. This page is at: www.ibm.com/storage/support.

Getting help by telephone

With the original purchase of the SAN File System, you have access to extensive support coverage. During the product warranty period, you may call the IBM Support Center (1 800 426-7378 in the U.S.) for product assistance covered under the terms of the hardware IBM warranty or the software maintenance contract that comes with product purchase.

Please have the following information ready when you call:

- Machine type and model or the SAN File System software identifier. The software identifier can be either the product name (SAN File System) or the Product Identification (PID) number.
- Serial numbers of the SAN File System engines, or your proof of purchase
- Description of the problem
- Exact wording of any error messages
- Hardware and software configuration information

If possible, have access to your computer when you call.

In the U.S. and Canada, these services are available 24 hours a day, 7 days a week. In the U.K., these services are available Monday through Friday, from 9:00 a.m. to 6:00 p.m. In all other countries, contact your IBM reseller or IBM marketing representative.¹

1. Response time will vary depending on the number and complexity of incoming calls.

Appendix E. Purchasing additional services

During and after the warranty period, you can purchase additional services, such as support for other IBM and non-IBM hardware, operating systems, and application programs; network setup and configuration; extended hardware repair services; and custom installations. Service availability and name might vary by country.

Appendix F. Logs

You can display the following logs from either the Administrative command-line interface or the SAN File System console.

- Administrative log
- Audit log
- Cluster log
- Event log
- Security log

Note: The cluster log is a consolidated view of the server log on each engine in the cluster. The event log is a subset of the messages stored in the cluster log; it contains only those messages that have an message type of event.

Related topics:

- “Logs” on page 30
- “Viewing logs” on page 71
- “catlog” on page 154

Administrative log

The administrative log contains messages generated by the Administrative server. If you display the administrative log from either the Administrative command-line interface or the SAN File System console, all administrative logs on all engines in the cluster are consolidated into a single view.

Fields:

Log entries contain the following fields:

Message ID

A unique identifier for the message.

Severity level

Indicates whether the entry is an informational, warning, error, or severe message.

Message type

Specifies whether the message is a normal log entry or one that was generated as a result of an event on the Administrative server.

Administrative server ID

A unique identifier for the Administrative server on which the command was issued.

Timestamp

A date followed by a local time.

Message

A textual explanation of the message.

Example:

The following example illustrates the consolidated view of the administrative log that is displayed through the Administrative command-line interface:

```
CIMServer: Info Normal mds_engine_0 May 16, 2003 7:27:33 AM
  Namespace \root\cimv2 initialized
CMMOM0411I Info Normal mds_engine_0 May 16, 2003 7:27:33 AM
  Authorization is not active
CMMOM0901I Info Normal mds_engine_0 May 16, 2003 7:27:33 AM
  IndicationProcessor started
CMMOM0906I Info Normal mds_engine_0 May 16, 2003 7:27:33 AM
  No pre-existing indication subscriptions
CMMOM0404I Info Normal mds_engine_0 May 16, 2003 7:27:33 AM
  Security server starting on port 5989
CMMOM0402I Info Normal mds_engine_0 May 16, 2003 7:27:33 AM
  Platform is Unix
```

Related topics:

- “Logs” on page 30
- “Viewing logs” on page 71
- “catlog” on page 154

Audit log

The Administrative audit log contains all administrative actions issued to the SAN File System. It contains a record of every command issued by a SAN File System administrator, from either the Administrative command-line interface or the SAN File System console, that changes the state of the Metadata server in some way.

The Administrative audit log contains a record of every command issued by a SAN File System administrator, from either the command-line interface or the SAN File System console, that changes the state of the Metadata server in some way.

Each Administrative server has its own audit log. If you display the audit log from either the Administrative command-line interface or the SAN File System console, all audit logs on all engines in the cluster are consolidated into a single view.

Fields:

Log entries contain the following fields:

Timestamp

A date followed by a local time.

Severity level

Set to a value of Informational (console) if the command succeeded. Otherwise, it is set to a value of Error.

Message ID

A unique identifier for the message.

Message type

Set to Audit. This field is contained in the audit log, but it is not displayed in the consolidated view.

Metadata server ID

A unique identifier for the Metadata server on which the command was issued.

Message

Contains the user name of the SAN File System administrator who issued the command followed by a functional replica of the message itself.

Example:

The following example shows a message from the consolidated view of the audit log that is displayed through the Administrative command-line interface:

```
2003-04-21 18:36:32 INFORMATIONAL HSTAD0083I A mds_engine_0
User Name: jozvold Command Name: MasterServiceAddServer
Parameters: SYSTEMCREATIONCLASSNAME=STC_Cluster
SYSTEMNAME=testnode CREATIONCLASSNAME=STC_MasterService
NAME=MasterService CLUSTERPORT=10001 IP=9.29.25.136
Command Succeeded.
```

Related topics:

- “Logs” on page 30
- “Viewing logs” on page 71
- “catlog” on page 154

Event log

The event log contains all messages from the cluster message log that have a message type of event.

Fields:

Log entries contain the following fields:

Message ID

A unique identifier for the message.

Severity level

The severity indicates whether the entry is an informational, warning, or error message.

Metadata server ID

A unique identifier for the Metadata server on which the command was issued.

Timestamp

The timestamp consists of a date followed by a local time.

Message

The message is a textual explanation.

Example:

The following example illustrates the event log that is displayed through the Administrative command-line interface:

ID	Level	Server	Date and Time	Message
TANCM0393I	Info	svc-mds1	May 16, 2003 2:08:01 AM	ALERT: The server state has changed from Initializing(2) to Joining(5).
TANCM0393I	Info	svc-mds1	May 16, 2003 2:08:13 AM	ALERT: The server state has changed from Joining(5) to Online(10).
TANCM0389I	Info	svc-mds1	May 16, 2003 2:08:13 AM	ALERT: The cluster state has changed from Forming(6) to Online(10).
TANCM0393I	Info	svc-mds2	May 16, 2003 2:09:22 AM	ALERT: The server state has changed from NotAdded(4) to Joining(5)

Related topics:

- “Logs” on page 30

- “Viewing logs” on page 71
- “catlog” on page 154

Security log

The security log displays the administrative user login activity for the Administrative server. If you display the security log from either the Administrative command-line interface or the SAN File System console, all security logs on all engines in the cluster are consolidated into a single view.

Fields:

Log entries contain the following fields:

Message ID

A unique identifier for the message.

Severity level

Indicates whether the entry is an informational, warning, or error message.

Administrative server ID

A unique identifier for the Administrative server on which the command was issued.

Message

A textual explanation of the message.

Example:

The following example illustrates the administrative log displayed through the Administrative command-line interface:

```
MMOM0302I Info mds_engine_0 May 19, 2003 9:21:17 AM
    User respey on client {1} could not be authenticated
MMOM0302I Info mds_engine_0 Jun 13, 2003 1:51:40 PM
    User jkaminski on client {1} could not be authenticated
MMOM0302I Info mds_engine_0 Jun 20, 2003 5:41:36 PM
    User fstock on client {1} could not be authenticated
```

Related topics:

- “Logs” on page 30
- “Viewing logs” on page 71
- “catlog” on page 154

Server log

The server log contains operational information. Each Metadata server in the SAN File System cluster has its own log. If you display the server log from either the Administrative command-line interface or the SAN File System console, all server logs on all engines in the cluster are consolidated into a single view. The consolidated view of the server message log is called the cluster log.

Fields:

Log entries contain the following fields:

Timestamp

A date followed by a local time.

Severity level

Indicates whether the entry is an informational, warning, error, or severe message.

Message ID

A unique identifier for the message.

Message type

Specifies whether the message is a normal log entry or one that was generated as a result of an event on the Metadata server.

Metadata server ID

A unique identifier for the Metadata server on which the command was issued.

Message

A textual explanation of the message.

Example:

The following example shows a message from the cluster log that is displayed through the Administrative command-line interface:

```
2003-04-16 12:55:50 INFORMATIONAL HSTPG0009I N
msd_engine_0 Using IP 9.38.203.26 port 10192
for Group Services messages.
```

Related topics:

- “Logs” on page 30
- “Viewing logs” on page 71
- “catlog” on page 154

Appendix G. SAN File System console help panels

Related topics:

The help panels appear in the SAN File System console Help Assistant frame. These panels provide field-level help information for items contained in the work area portion of the SAN File System console.

- “Using the Help Assistant” on page 433
- “Using the SAN File System console”

Using the SAN File System console

The SAN File System console is the Web-based user interface for administering SAN File System. You can also use client-specific tools such as the Windows Explorer or a shell on AIX or Linux to interact with the file system itself, however, you cannot administer SAN File System from the client-side.

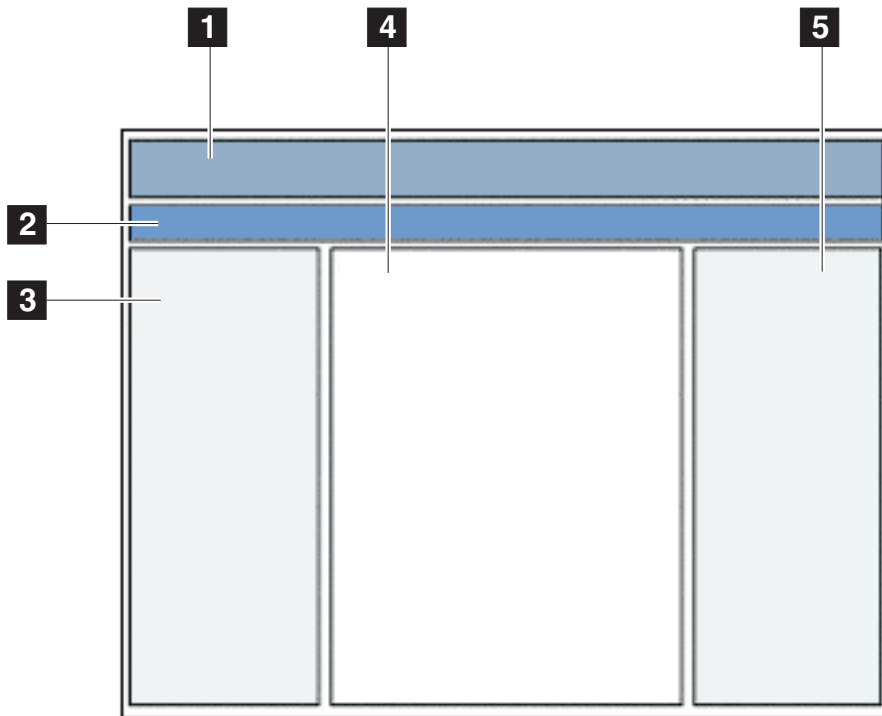
The SAN File System console panels offer easy control of main and sub tasks. Several main panels also contain sorting and filtering tables so that you can view only the details that interest you. In addition, there are status indicators, inline messages and user assistance.

Related topics:

- “Supported browsers” on page 437
- “Using the Help Assistant” on page 433
- “Overview”
- “Tasks and panels” on page 336
- “Status, help, and tables” on page 340
- “User interfaces” on page 47

Overview

SAN File System console overview: The SAN File System console includes a title bar, banner, task bar, navigation, and a work area. Several user assistance resources are also available, including an embedded Help Assistant for panel help information, as well as a more comprehensive SAN File System Information Center.



- 1** Banner
- 2** Task bar
- 3** My Work frame
- 4** Work area
- 5** Help Assistant

Title bar:

The title bar is actually part of the Web browser — either Internet Explorer or Netscape. The title bar will display “IBM TotalStorage SAN File System console.”

Banner:

The banner area appears between the title bar and task bar sections of the SAN File System console. You can show or hide the banner using the Show/Hide

banner toggle icon ().








Task bar and task management:

You can perform the following actions using the task bar:

- Access the Task Manager
- View or access currently open tasks
- Close or open the banner
- Sign off from the SAN File System console
- Obtain help information

The task bar consists of two segments. On the left side is the task segment, and on the right is the tool segment.

Table 11. SAN File System task bar buttons

Task segment	Tool segment
<p>The task segment presents the task-related functions and includes the following task buttons:</p> <p>Task Manager (“All Tasks”) A single button that brings up a full panel listing each task that is currently running. It allows you to end a task or to surface one that is running.</p> <p>Current tasks Provides links to all the tasks that are currently running. Each task is presented as a link with a small icon followed by text. The task segment also keeps track of all opened main tasks, and allows you to quickly jump back and forth between them. Main tasks are those tasks listed in the My Work frame. When a task is opened from the My Work frame, the task bar is populated with that task heading. Only one task can be active at a time (displays in reverse emphasis).</p>	<p>The tool segment provides a location for the following tool buttons:</p> <p>Toggle Banner Allows you to open or close the banner to free up screen space as required. This button has two states:</p> <ul style="list-style-type: none">  “Show Banner” — icon points down to reflect the opening action  “Hide Banner” — icon points up to reflect the closing action <p>Sign off icon  Signs you off of the SAN File System console.</p> <p>Toggle Help Assistant This button has two states:</p> <ul style="list-style-type: none">  “Show Help Assistant” — icon points left to reflect the opening action  “Hide Help Assistant” — icon points right to reflect the closing action <p>Note: You can also close the Help Assistant from the  close button on its title bar.</p> <p>Launch the Information Center  This button launches the SAN File System Information Center in another window and makes that window active. If the SAN File System Information Center is already open, it resurfaces the existing one with information related to the currently active panel.</p>

Related topics:

- “Supported browsers” on page 437
- “Using the Help Assistant” on page 433
- “Using the SAN File System console” on page 333
- “Status, help, and tables” on page 340
- “Tasks and panels” on page 336
- “User interfaces” on page 47

Tasks and panels

The SAN File System console displays different panel types and task levels.

Tasks and panels:

Each new task opened from the **My Work** frame will create an instance in the task bar. When a new task is opened it becomes the in-focus task, and is populated to the right of any task bar instances. Only one instance of a task can be listed in the task bar. If you select a task from the **My Work** frame that is already opened and represented in the task bar, the previous instance of that task in the task bar becomes selected and will be the in-focus task.

Task levels are divided into main tasks and sub tasks. Main tasks are primary tasks that display on the task bar and are incapable of multiple instances; a main task cannot be opened twice simultaneously. Main tasks close when the task is completed. Sub tasks are secondary to main tasks and are also incapable of multiple instances. The following tables summarize SAN File System task types and behaviors.

Table 12. SAN File System task levels


Main task	Sub task
Primary tasks that are directly accessible from the navigation frame (called My Work). In addition, main tasks: <ul style="list-style-type: none">• display on the task bar.• can only have one instance open at a time. For example, if the main "Storage Pool" task is already open, another instance of this main task cannot simultaneously be opened. Instead, the original instance will be resurfaced.• can only be re-accessed by closing any associated sub task.	Tasks that are secondary to a main task or another sub task. In addition, sub tasks: <ul style="list-style-type: none">• display on the task bar.• can only have one instance open at a time.

Table 13. SAN File System task types

Resource task	Action task
<p>Most of the main tasks in SAN File System are resource-oriented. Typically, condensed, data-rich displays in the form of tables are shown and you are allowed to perform many action-oriented sub tasks from the main task (for example, add, change, remove).</p>	<p>Action tasks perform a specific function and can range from complex actions that require multiple steps (such as wizards), to simpler actions that might involve only one step. Action tasks run to completion and then automatically close, resurfacing the calling task if there was one. The following SAN File System tasks are all action-oriented:</p> <p>Wizard Multi-step tasks that consist of several panels. You will be automatically guided through the wizard with buttons for Next, Back, and so forth.</p> <p>Single step Tasks that only consist of a single step and panel with the appropriate fields or objects for completing the task.</p> <p>Immediate Tasks that consist of no user (only system) steps. These tasks begin immediately and may or may not lead to subsequent panels. Immediate tasks are not displayed on the task bar. There are two types of immediate tasks:</p> <p>Status Contain no user settings, but the status is shown until the task completes.</p> <p>No status Contain no user settings, nor status indicators.</p>

My Work frame:

The SAN File System navigation frame is titled **My Work**. The **My Work** frame allows you to launch tasks by selecting links that open panels in the work area. By default, the **My Work** frame is open when you first sign on, but you can close it by

selecting the  icon. This icon functions as a toggle, allowing you to open and close this frame as preferred.

The **My Work** navigation frame contains task categories with associated main tasks listed under them. Different main tasks may use different panel types depending on the associated information and actions. More information on panel types is presented later.

Table 14. SAN File System navigation structure

My Work navigation task category	Main tasks	Panel Type
Welcome	—	splash with links

Table 14. SAN File System navigation structure (continued)

My Work navigation task category	Main tasks	Panel Type
Monitor System	System Overview	tabular
	Statistics	notebook tab with tables
	Processes	tabular
	SNMP Properties	notebook tab
	Cluster Log	tabular
	Administrative Log	tabular
	Audit Log	tabular
	Security Log	tabular
Administer Access	User Roles	tabular
	Users	selection table
Manage Servers and Clients	Cluster	list
	Servers	selection table
	Client Sessions	tabular
	Client Properties	form
Manage Filing	Policies	selection table
	Create a Policy	wizard
	Filesets (Containers)	selection table
	Create a Fileset	form
Manage Storage	Storage Pools	selection table
	Create a Storage Pool	wizard
	Volumes	selection table
	Add Volumes	wizard
	LUNs	tabular
	Available LUNs	selection table
Maintain System	Engines	selection table
	FlashCopy Images	selection table
	Create FlashCopy Images	wizard
	Server Restart Service	selection table
	Check Metadata	form and selection table
	Collect Diagnostic Data	form and selection table
	Disaster Recovery	selection table
	Download Client Software	tabular

Work area:

All user input and product content is displayed through panels in the work area. Panels contain static information and interface elements that you interact with, such as text fields and tables.

Note: If you refresh the panel, any information entered or selected in the work area will be erased.

There are two types of panels that can be displayed in the work area, main task panels and sub task panels. Both main task and sub task panels can also display in-page messages that provide information and warnings. The in-page messages are detailed in later sections of this topic. Panel types are described below.

Table 15. SAN File System panel types

Panel type	Description
Tabular	<p>Panels containing a table. The tables may be:</p> <ul style="list-style-type: none"> • multi-selection with check box selectors for each row • single-selection with radio button selectors for each row • non-selection with no row selection mechanism
List	<p>Panels with a simple two-column list. List panels are used for showing “Detail” views, which are typically accessed by selecting a link from a row on tabular panel.</p>
Form	<p>Panels containing text entry fields, radio buttons, check boxes, drop-down lists, or any combination. Form panels are most commonly used in SAN File System wizards.</p>
Notebook tab	<p>Panels consisting of a list of rows, or “notebook tabs” that you can select for viewing additional details relating to the resource-task from which they originate. For example, if you select the Storage Pools task, then from that main task panel, select a particular storage pool on which to view properties, you will go to a “notebook tab” panel with individual “tabs” for General Settings, Current Volumes, Available LUNs, and Details relating to that storage pool.</p> <p>Note: Additional information about notebook tab panels is provided later in the “Property notebook” section.</p>
Confirmation	<p>Panels consisting of only a simple message box to verify the action that you just selected and its possible results. There are two types of confirmation panels:</p> <p>Warning Provides details of the potentially negative consequences of certain actions, such as losing data.</p> <p>Informational Provides task implications that do not involve potentially negative consequences.</p>

Related topics:

- “Supported browsers” on page 437
- “Using the Help Assistant” on page 433
- “Using the SAN File System console” on page 333

- “Overview” on page 333
- “Status, help, and tables”
- “User interfaces” on page 47

Status, help, and tables

The SAN File System console offers status indicators, user assistance, and table sorting and filtering.

Status and progress indicators:

When a task takes more than a few seconds to complete, a progress indicator displays. The indicator shows that activity is taking place, but not how long the activity will take to complete. You can wait until a task completes, or use the task bar, or **My Work** frame to start another task before it completes. The **Close** button will close the panel without stopping the underlying action.

User assistance:

You can access both the Help Assistant and the SAN File System Information Center through the SAN File System console task bar.

Help Assistant:

The Help Assistant provides embedded field-level descriptions corresponding to the items contained on the panels of the SAN File System console. Because the panel help is embedded into the overall console, you can obtain context-sensitive information without interrupting your current activity. Each time you select a different panel, the information in the Help Assistant changes accordingly.

You can launch the Help Assistant by selecting the **Toggle Help Assistant** icon (




) from the right side of the task bar. Because this button performs in a toggle

More details about the SAN File System Help Assistant can be found by following the “Related topics” link below.

Information Center:

An Information Center is a searchable, online, topic-based help system that can be launched from a Web browser. The SAN File System Information Center provides additional SAN File System product documentation.


You can open the Information Center from the SAN File System console by

selecting the **Launch Information Center** icon () from the right side of the task bar. Alternately, many of the related topics provided in the Help Assistant frame will link you directly to the Information Center.

More details about the SAN File System Information Center can be found by following the “Related topics” link below.

Sorting and filtering tables:


The sortable and filterable tables in the SAN File System work area display object attributes in table columns. The data is chunked into pages as needed to reduce scrolling. The cell at the bottom of the table displays the number of pages of data and the number of table objects, including the number filtered, displayed and

selected. You can page through the objects by clicking the  icon at the bottom left of the table, or jump to pages by entering a page number and pressing the **Go** button.

You can view the details of an object by clicking on one of its attributes, indicated by an underlined link. You can click on the name or ID, which links to additional information about the associated object. The object “Details” view is opened as a sub task panel.

The tables provide sort and filter functions through icons in the top left corner. These functions provide an inline way for you to quickly search, find, select, and use resources within SAN File System tables. For example, sorting can help you quickly determine which filesets or storage pools are approaching quota limits.


You can sort, not just on single columns, but on multiple ones. Perform the following steps to sort on table attributes:

1. Select the **Edit Sort** icon () from the table you want to sort.
2. Choose the attribute column or columns that you wish to sort on from the drop-down boxes.

Note: You can sort on up to three columns.

3. Choose a preferred sort order (ascending or descending).
4. Click **OK** to confirm.

Table filtering offers a powerful data-reduction technique so that you are presented with only the information you need. Perform the following steps to place a filter on a table column:

1. Select the **Show Filter Row** icon () from the table you want to filter.
2. Click the filter name link on the column you want to filter.

Note: If the column currently has no filtering applied to it, the name link says “None.”

3. Apply the preferred filtering.
4. Click **OK** to confirm.

The sorting and filtering icons are as follows:

Table 16. SAN File System console table sorting and filtering icons








Icon	Description
	The Select All icon selects check boxes for all rows.
	The Deselect All icon clears check boxes for all rows.

Table 16. SAN File System console table sorting and filtering icons (continued)

Icon	Description
	The Show/Hide Filter Row icon toggles the filter row on and off.
	The Clear All Filters icon clears any previously set filters for all columns.
	The Edit Sort icon allows advanced sorts such as column sort order.
	The Clear All Sorts icon clears all table sort actions.
	The Collapse/Expand Table icon toggles between show all and hide all table rows.

The check boxes in the first column of the table allow you to select one or more objects on which to perform actions. Actions specific to the table objects are provided in the drop-down box at the top of each table. The drop-down box provides an additional way to access the sorting and filtering functions. An example of the contents of the drop-down box follows:

Note: The first three functions listed may not be applicable for every table, but serve here as a sample only.

- Add...
- Properties...
- Delete...
- Other panel-specific table actions...
- Table Actions (available for every table as described previously)
 - Select All
 - Deselect All
 - Show/Hide Filter Row
 - Clear All Filters
 - Edit Sort
 - Clear All Sorts
 - Collapse/Expand Table

The **Go** button to the right of the drop-down box submits the selected action. Some actions depend on one or more table items being selected; whereas others can only be performed against one selected item. When you can only select one item to complete an action, that action will have the number one in parentheses listed beside it as in this example: “Properties (1).”

When an action is submitted, the system will verify that a proper selection has been made within the table. If a proper selection has not been made, the page will display a message with instructions of how to correct your selection and complete the action.

Messages:

In-page messages are embedded within the work content area of the panels and provide information or warnings. They appear above the content of the panel, but below the panel description and instructions.

A common use of in-page messages is to inform you that an action needs to take place. For example, if you select the delete action from a sortable, filterable table, but do not select an item in the table, a message will be sent informing you that you need to select an item for the action to take place. In-page messages are also used to inform you of any invalid entries when submitting an input form.

If a message ID is available it will be presented as a link to access help information pertaining to that message. The message help will appear in the Help Assistant frame of the interface. To dismiss the message, click **Close Message** at the bottom of the message box. However, you can continue with the task without dismissing the message box.

Note: Both displaying message help and dismissing the message will force a refresh of the panel.

Entry fields:

Entry fields will dynamically indicate whether or not a field is required and whether or not it is in error. Required entry fields are shaded yellow and have a star-like icon by the field name. When an error has been made in an entry field, it will display a red outline, as well as an “x” icon beside the field name.

Property notebook:

In the property notebook, tabs appear on the left side of the panel. By default the last tab is always selected when the panel opens. As necessary, the tabbed panels may contain a description and instructions, as well as applicable fields, selection boxes, tables and in-page buttons.

The **OK**, **Apply**, and **Cancel** buttons are located at the bottom and apply to the entire panel (that is, they act upon all tabs). The **OK** button closes and submits any actions, while **Cancel** closes and dismisses all actions. The **Apply** button submits any actions, but keeps the panel open.

Note: Some notebook panels only have a **Close** button (not **OK** and **Apply**).

System overview:

The **System Overview** panel enables you to view general system health by providing all the key statistics in one place. Using this overview prevents you from needing to open several different panels to piece together the same information. From the overview you can periodically monitor major components of the system, including cluster state, clients, filesets, Metadata servers, recent events, and so forth. This is the highest-level monitoring panel in SAN File System.

The overview not only provides statistics on key overall values, status conditions, and some drill downs, but also uses aggregation techniques to summarize conditions.

To access the SAN File System overview, go to the **My Work** frame, and under **Monitor System**, click **System Overview**.

Related topics:

- “Supported browsers” on page 437
- “Using the Help Assistant” on page 433
- “Using the SAN File System console” on page 333
- “Overview” on page 333
- “Tasks and panels” on page 336
- “User interfaces” on page 47

Add Volumes to a Storage Pool- Add Volumes

This panel (second step in the wizard) allows you to add available LUNs as volumes to the selected storage pool.

Attributes:

Volume Name Prefix	(Required entry field.) The prefix entered here is added to the LUN ID, which is then combined with a sequence number to each volume. For example, if you enter the prefix “ABC,” your volumes may become ABC4.1, ABC4.2, and so forth. You should use a prefix that will correspond to the storage device of the LUN for easy association. Note: You can later rename the volumes if you want.
Initially activate volumes	Selecting this option will add the volumes and activate them when they are added. Otherwise, the volumes will be added in a suspended state, and data cannot be written to them.
Forced Addition	When not selected, this setting will ensure that volumes are not currently assigned in the associated cluster. When it is selected, it will take over the LUN, regardless of whether or not it is assigned to a storage pool. Note: The volume name follows the formula: “Prefix-LUN ID.” Another sequence number is not added. So a prefix of “disk” and a LUN ID of 3 would yield “disk-3” as the volume name.

Attributes:

The **Available LUNs** table displays the available LUNs from which you select to write to the storage pool as volumes. It displays the size of the LUN in megabytes, as well as the following:

LUN ID	Links to additional details about the selected LUN. Note: If you enter the Add Volumes wizard from the Available LUNs panel, your previously chosen LUN or LUNs will also be preselected.
--------	---

Vendor	The vendor of the storage device associated with the LUN. Note: The system displays “-” if the vendor cannot be determined.
Product	The product name of the storage device associated with the LUN. Note: The system displays “-” if the product cannot be determined.
Version	The version of the storage device, for example: “4.1.” Note: The system will displays “-” if the version cannot be determined.
Engine WWN	Uniquely identifies the storage device by its World Wide Name (WWN). This information is helpful if you want to ensure that all LUNs in a storage pool are from the same storage device.

Related topics:

- “Managing volumes and LUNs” on page 128
- “Storage pools” on page 37
- “Storage devices” on page 46
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Add Volumes to a Storage Pool- Introduction

Use this wizard to create new volumes from available LUNs and add them to a SAN File System storage pool. This three-step wizard will allow you to select the storage pool, add the volume or volumes, and verify your settings before they are added.

Note: The **Next** button in the wizard will be disabled if any failures occur during the add volume process.

Related topics:

- “Managing volumes and LUNs” on page 128
- “Storage pools” on page 37
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Add Volumes to a Storage Pool- Select Storage pool

This panel (first step in the wizard) allows you to select a storage pool to which to add available LUNs as volumes.

Attributes:

In addition to its description, the selection table provides the following information about each storage pool:

Name	Links to additional details about the associated storage pool.
------	--

Type	<p>One of three storage pool types are possible:</p> <p>User storage pools Data storage pools in which user data gets stored in blocks.</p> <p>User Default storage pool The user storage pool that data gets assigned to, if not picked up by a policy rule.</p> <p>System storage pool The storage pool that stores the metadata (regarding filesets, policies, and so forth). Note: It is important to ensure that this storage pool never runs out of space.</p>
Size (MB)	Total space available in the associated storage pool.
Used (MB)	Amount of space currently used by the storage pool.
Used (%)	Percentage of the total storage pool size that is currently being used.
Alert (%)	Storage pool usage percentage threshold. When the Used (%) reaches this threshold, SAN File System will send an alert.
Volumes	Number of volumes in the associated storage pool.

Related topics:

- “Managing volumes and LUNs” on page 128
- “Storage pools” on page 37
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Add Volumes to a Storage Pool- Verify Settings

This panel (final step in the wizard) lets you verify your selections before the volumes are added. All of the displayed fields are read-only. To make changes, return to any of the previous panels in the wizard.

Attributes:

The **New Volumes** table lists the size of each volume in megabytes, the LUN ID associated with the volume, as well as the following information:

Volume Name	The name that will applied to the new volume.
Initial Status	<p>The status of the volume can be either Active or Suspended. Note: If volumes are in a suspended state, data cannot be written to them. You can change a volume to the active state by going back to the previous step of the wizard (Add Volumes) and select the checkbox to Initially activate volumes.</p>

Related topics:

- “Managing volumes and LUNs” on page 128
- “Storage pools” on page 37
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Administrative Log

This panel enables you to monitor SAN File System administrative processes related to the Administrative agent.

Actions:

You have the option to set certain parameters regarding which messages the administrative log retrieves.

Messages to Retrieve

You have three choices regarding which type of messages to retrieve:

Most Recent (default)

Always gets the most recent rows in the log.

Note: This choice is not relative to the current log view, unlike the other two choices.

Previous Messages

Selects rows down the log relative to the current view.

Next Messages

Selects rows up the log relative to the current view.

Number of Messages

Determines how many messages display at one time. You can choose from 10, 25, 50, or 100.

Refresh

Reloads the log based on the adjacent log-retrieval settings.

Fields:

The **Administrative Log** table lists the following details:

Message ID

Follows this convention:





“XXX” Component

“YY” Subcomponent

“nnnn”

Message code

“Z” Severity (Severe, Error, Warning, or Information level)

Level	Based on the severity level of the Message ID. Four possibilities exist: <ul style="list-style-type: none">  Severe  Error  Warning  Information
Server	The Metadata server name that is the source of the message.
Date and Time	System (not local) time that the message was issued.
Message	Text of the message.

Related topics:

- “Alerts” on page 4
- “Common Information Model” on page 10
- “Metadata server” on page 33
- Chapter 3, “Monitoring”, on page 57
- “Using the SAN File System console” on page 333

Attach Fileset

Use this panel to attach a fileset (container) using a different directory name.

Actions:

Attach Point Path and directory location of the fileset within the global namespace.

Fields:

Existing Directory Path Specifies the existing directory path (relative to the root of the global namespace and without the / prefix) at which to attach the fileset. This directory must already exist on the client side. The root of the global namespace must be included in the directory path. Use only forward slashes (/) in the directory path for delimiters.

New Directory Name Specifies the directory name for the root of the fileset directory tree. The directory name must not already exist on the client side. This directory can be up to 256 characters in length and must not contain backslash (\) or forward slash (/) characters as delimiters. **Note:** You must enter the new directory name when the fileset is created, moved or reattached.

Related topics:

- “Detaching a fileset” on page 105

- “Filesets” on page 24
- “Global namespace” on page 29
- “Using the SAN File System console” on page 333

Audit Log

This panel enables you to monitor SAN File System status.

Actions:

You have the option to set certain parameters regarding which messages the audit log retrieves.

Messages to Retrieve

You have three choices regarding which type of messages to retrieve:

Most Recent (default)

Always gets the most recent rows in the log.

Note: This choice is not relative to the current log view, unlike the other two choices.

Previous Messages

Selects rows down the log relative to its current view.

Next Messages

Selects rows up the log relative to its current view.

Number of Messages

Determines how many messages display at one time. You can choose from 10, 25, 50, or 100.

Refresh

Reloads the log based on the adjacent log-retrieval settings.

Clear Log

Deletes all messages in the log.

Fields:

The **Audit Log** table lists the following details:

Message ID





Follows this convention:

“XXX” Component

“YY” Subcomponent

“nnnn”
Message code

“Z” Severity (Severe, Error, Warning, or Information level)

Level	Based on the severity level of the Message ID. Four possibilities exist: <ul style="list-style-type: none">  Severe  Error  Warning  Information
Server	The Metadata server name that is the source of the message.
Date and Time	System (not local) time that the message was issued.
Message	Text of the message.

Related topics:

- “Metadata server” on page 33
- Chapter 3, “Monitoring”, on page 57
- “Using the SAN File System console” on page 333

Available LUNs

Use this panel to view and perform actions on the LUNs designated for use by SAN File System. Available LUNs are the writable LUNs that can be seen by SAN File System, but have not yet been assigned to a storage pool.

Note: To view all LUNs that SAN File System can access, whether or not they have been assigned to a storage pool, select the **LUNs** panel.

Actions:

Add to a Storage Pool	Takes you to a wizard for adding a volume to a storage pool. Note: When entering the Add Volumes wizard from the Available LUNs panel, your chosen LUNs will also be preselected in the wizard.
Details	Displays additional LUN details, such as the Volume Name, State, Port WWN, OS Device Path, and Sector Size.

Fields:

The **Available LUNs** table allows you to select one or more LUNs to add as volumes to a SAN File System storage pool. In addition, the table displays the following information:

LUN ID	Links to additional details about the associated LUN.
Size (MB)	Total size of the LUN in megabytes.
Vendor	The vendor of the storage device associated with the LUN. Note: The system displays “-” if the vendor cannot be determined.

Product	The product name of the storage device associated with the LUN. The product name of the storage device associated with the LUN. The system displays the following IBM products in addition to other products:												
	<table> <tr> <td>2105F20</td> <td>ESS F20</td> </tr> <tr> <td>2105800</td> <td>ESS 800</td> </tr> <tr> <td>2145</td> <td>SAN Volume Controller and SAN Integration Server</td> </tr> <tr> <td>3552</td> <td>FAStT 5000</td> </tr> <tr> <td>1722</td> <td>FAStT 600</td> </tr> <tr> <td>1742</td> <td>FAStT 700 and 900</td> </tr> </table> <p>Note: The system displays “-” if the product cannot be determined.</p>	2105F20	ESS F20	2105800	ESS 800	2145	SAN Volume Controller and SAN Integration Server	3552	FAStT 5000	1722	FAStT 600	1742	FAStT 700 and 900
2105F20	ESS F20												
2105800	ESS 800												
2145	SAN Volume Controller and SAN Integration Server												
3552	FAStT 5000												
1722	FAStT 600												
1742	FAStT 700 and 900												
Storage Device WWNN	The world-wide node name of the hosting storage device (for example, ESS or FAStT). This information is helpful if you want to ensure that all LUNs in a storage pool are from the same storage device.												

Related topics:

- “Managing volumes and LUNs” on page 128
- “Storage pools” on page 37
- “Storage devices” on page 46
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Change Power State of Engine

Use this panel to verify the engine or engines selected for a power state change, and then choose a change method.

Attributes:

Engines Selected for Power State Change	Read-only list of the selected engines that are now targeted for power state change.
---	--

Actions:

Power State Change Method	Select from one of the five listed methods for changing the engine’s power state. Note: For “ Schedule Power On Or Off ,” select the radio button next to that method, and then select either On or Off .
Time settings	Select the year, month, day, and hour when you want the power state change to take effect. Note: The Current ASM field is a read-only field. This is the current time on the local clock for the Advanced Systems Management device’s local clock. This time reference must be used to schedule the “power-off,” operation and is independent of the time on the engine.

Related topics:

- “Metadata server” on page 33
- “Engines” on page 23
- “Using the SAN File System console” on page 333

Change State of Cluster

Use this panel to change the state of the cluster in one of four ways: bring online, fully or partly quiesce, or take offline. Cluster state change prepares both clients and servers for volume-level backup.

Fields:

Resume to Online	Resumes the cluster if it is in a quiescent or offline state. (Default choice if the cluster is fully quiescent, partly quiescent, or offline.)
Partly Quiesce	Only quiesces the metadata activity. Allows existing client file input/output (I/O) activity to continue (used for “dirty backups”). (Default choice if the cluster is online.)
Fully Quiesce	Quiesces all metadata and client activity (used for “clean backups”). All data and metadata is flushed to disk.
Take Offline	<p>Attention: Moving to this state could cause a major delay to users that have current client sessions.</p> <p>Suspends all client sessions.</p> <p>Attention: Moving to this state could cause a major delay to users that have current client sessions.</p>

Related topics:

- “Cluster” on page 16
- “Cluster states” on page 19
- “Managing the cluster” on page 82
- “Clients” on page 11
- “Using the SAN File System console” on page 333

Check Metadata

Use this panel to check the integrity of SAN File System metadata. You also have the option of repairing the metadata.

Attributes:

Repair as necessary	Enables the check procedure to repair metadata as needed. Note: The cluster may need to be taken offline and require manual intervention to repair metadata.
---------------------	--

Metadata Type	Specify which type of metadata to check, repair, or both: <ul style="list-style-type: none"> • Content and Structure (default enabled) • Content Only • Structure Only
Metadata Location	Select which storage pool and fileset (container) combination you want the check to run against: <ul style="list-style-type: none"> • System Pool and All Filesets (Container) (default enabled) • System Pool Only • All Filesets (Containers) • Specific Filesets (Containers)

Fields:

If you choose to run the metadata check only against specific filesets (as specified in the **Metadata Location** setting), you need to select one or more filesets from the Filesets (Containers) table. The table lists the following details:

Name	Name of the associated fileset.
State	Two fileset states are possible: <ul style="list-style-type: none"> • Attached • Detached
Used (MB)	The number of megabytes used by the fileset.
Attach Point	Path and directory location of the fileset within the global namespace. (Will show "-" if detached.)
Children	Number of subfilesets. (If there are no subfilesets, "-" displays.)

Related topics:

- "Attaching a fileset" on page 103
- "Checking metadata" on page 112
- "Detaching a fileset" on page 105
- "File placement" on page 43
- "Filesets" on page 24
- "Global namespace" on page 29
- "Storage pools" on page 37
- "Using the SAN File System console" on page 333

Check Metadata Progress

Use this panel to view the progress of the metadata check and repair task. You can also stop the check and repair process from this panel, since it can be potentially time-consuming.

Fields:

Progress Status Area

Progress text	Displays one of the following, depending on the check and repair task status: <ul style="list-style-type: none">• “The metadata check is in progress.”• “The metadata check completed <time stamp>.”• “The metadata check was stopped <time stamp>.”
Progress bar	Animated graphic that continues until the metadata check completes, or is stopped.

Attributes:

Metadata Check Settings

Repair as necessary	Indicates if the repair operation is enabled. If so, the overall “check metadata” task will repair metadata as needed.
Metadata Type	Indicates which type of metadata is being checked, repaired, or both (if the repair setting is enabled).
Metadata Location	Indicates which storage pool, fileset, or combination the check is running against.

Fields:

Pushbuttons.

Stop	Stops the check. Does not close the panel. Note: This pushbutton becomes disabled when the operation is complete.
Close	Closes the panel.

Notes:

1. The close action does not stop the underlying check if it is currently running.
2. If you close this panel before the check completes, you cannot stop the check process from the SAN File System console. However, you can stop the process from the command line interface (CLI) using the **stopmetadatacheck** command.

Related topics:

- “Checking metadata” on page 112
- “File placement” on page 43
- “Filesets” on page 24
- “Storage pools” on page 37
- “stopmetadatacheck” on page 282
- “Using the SAN File System console” on page 333

Client Privileges

Use this panel to view, revoke, or grant root privileges for SAN File System clients. There two client privilege (that is, access) levels:

Root Privileged client access level.

Standard

Normal (and default) client access level.

Actions:

Revoke Clients Root Privileges

Removes unrestricted access from the selected clients.

Note: You can also perform this action from the **Client Sessions** panel.

Fields:

You can select a client for which to revoke root privileges. The **Client with Root Privileges** table lists the following information:

Client Name

Name of the client. This name is given on the client side.

Note: This is not the IP host name.

Client IP

IP address of the client.

Actions:

Grant Root Privilege

Client Name

Type a client name for which to grant unrestricted access.

Note: You can also perform this action from the **Client Sessions** panel.

Add pushbutton

Adds the client to the list of those with root privileges.

Related topics:

- “Clients” on page 11
- “Locks and leases” on page 30
- “Metadata server” on page 33
- “Using the SAN File System console” on page 333

Client Sessions

Use this panel to grant or revoke root privileges for SAN File System clients, or to view client session properties. There are two client privilege (that is, access) levels:

Root Privileged client-access level.

Standard

Normal (and default) client-access level.

Actions:

Grant Clients Root Privileges

Allows you to grant unrestricted access to the selected clients.

Note: You can also perform this action from the **Client Privileges** panel.

Revoke Clients Root Privileges	Removes unrestricted access from the selected clients. Note: You can also perform this action from the Client Privileges panel.
Details	Displays additional detailed properties of the selected client, including transactions and locks

Fields:

You can select a client for which to grant root privileges, revoke root privileges, or view details. The **Client Sessions** table lists the following information:

Client	Identifies each client by its name. This name is given on the client side. Note: It is not an IP host name.
Session ID	Links to additional details about the selected client. Note: A single client may have multiple, simultaneous sessions — one per Metadata server.
State	Two client session states are possible: Expired Lease is expired. Current Lease is currently valid. Note: A client might also not be connected, in which case, neither state applies.
Server	Name of the Metadata server that is part of the client session.
Renewals	A count of session renewals that increments once per renewal. The renewal gives you an indication of the length of each client session.
Client Privilege	Refers to the client root privileges, which is one of these options: <ul style="list-style-type: none"> • Root (privileged client) • Standard (normal client)

Related topics:

- “Clients” on page 11
- “Locks and leases” on page 30
- “Metadata server” on page 33
- “Using the SAN File System console” on page 333

Cluster

Use this panel to perform an action on the SAN File System cluster, or to view cluster properties.

Actions:

Properties	Displays detailed properties of the SAN File System cluster and lets you view tuning settings. This panel also lets you modify the storage pool-space-reclamation interval.
Start	Starts the cluster. Note: Starting the cluster could take some time.
Change State	Lets you change the state of the cluster (bring online, fully or partly quiesce, or take offline). Cluster state change prepares both clients and Metadata servers for volume-level backup.
Stop	Lets you gracefully stop the Metadata servers in the cluster. The Metadata server and their associated states, roles, and filesets are displayed for review and confirmation before the stop operation begins.
Commit Software Upgrade	Upgrades the currently operational software version on the cluster. Notes: <ol style="list-style-type: none">1. Cluster upgrade can occur only after each individual Metadata server has been upgraded to the same new software version. Individual Metadata server upgrades must be done at the Administrative command-line interface.2. The cluster upgrade can only be run on the master Metadata server.3. The cluster does not need to be stopped for the upgrade.

The cluster table lists these cluster properties: name; system identification number; current state; and the date and time of the last state and target state changes. In addition, the table provides the following:

Attributes:

Target State	State to which the cluster is trying to progress. For example, if the system is booting up, the pending target state might be "Online."
Servers	Total number of Metadata servers in the cluster.
Active Servers	Number of Metadata servers in an active state (online, offline or quiesced).
Privileged Clients	Number of clients that have been granted root privileges.
Software Version	Current software version on the cluster. It may or may not have yet been committed.

Committed Software Version	Currently committed software version on the cluster. This attribute does not indicate whether or not the commit has finished, because it only updates the metadata structures on demand. Note: Typically, if an upgrade is in progress, the committed software version will be an earlier version than the pending cluster software version.
Last Software Commit	Date and time that last software commit was initiated.
Software Commit Status	Indicates if an upgrade is: In progress The commit is ongoing. A commit can take a long time to complete because it is done on demand. Note: Future upgrades will not be possible if the software commit status is in progress. Not in progress Either a commit has never been done, or one has completed.
Installation Date	Date and time that the global fileset was created (that is, when SAN File System was installed).

Related topics:

- “Backup and restore” on page 5
- “Cluster” on page 16
- “Cluster states” on page 19
- “Managing the cluster” on page 82
- “Metadata server” on page 33
- “Metadata server states” on page 35
- “Using the SAN File System console” on page 333

Cluster Log

This panel enables you to monitor the servers in the SAN File System cluster.

Actions:

You have the option to set certain parameters to limit which messages the cluster log retrieves.

Messages to Retrieve

You have three choices regarding which type of messages to retrieve:

Most Recent (default)

Retrieves the most recent rows in the log.

Note: This choice is not relative to the current log view, unlike the other two choices.

Previous Messages

Retrieves rows down the log relative to the current view.

Next Messages

Retrieves rows up the log relative to the current view.

Number of Messages

Determines how many messages display at one time. You can choose from 10, 25, 50, or 100.

Message Types

You can choose to filter from All Messages or Events Only.

Note: An event is a significant or major message that will often, but not necessarily, trigger an alert.

Refresh

Reloads the log based on the adjacent log-retrieval settings.

Clear Log

Deletes all messages in the log.

Fields:

The **Cluster Log** table lists the following details:

Message ID

Follows this convention:

“XXX” Component

“YY” Subcomponent

“nnnn”

Message code

“Z” Severity (Severe, Error, Warning, or Information level)

Level

Based on the severity level of the Message ID. Four possibilities exist:

-  Severe
-  Error
-  Warning
-  Information

Type	Message types fall into one of three categories: Normal Messages that are not events. Event A significant or major message will often, but not necessarily, trigger an alert. Note Other message types such as "Audit" or "Trace."
Server	The Metadata server name that is the source of the message.
Date and Time	System (not local) time that the message was issued.
Message	Text of the message.

Related topics:

- "Alerts" on page 4
- "Cluster" on page 16
- "Cluster states" on page 19
- "Metadata server" on page 33
- "Metadata server states" on page 35
- Chapter 3, "Monitoring", on page 57
- "Using the SAN File System console" on page 333

Cluster Properties - Details

Use this panel to view information about the SAN File System cluster.

The cluster **Details** table lists these cluster properties: name; system identification number; current state; and the date and time of the last state and target state changes. In addition, the table provides the following:

Attributes:

Target State	State to which the cluster is trying to progress. For example, if the system is booting up, the pending target state might be "online."
Servers	Total number of servers in the cluster.
Active Servers	Number of servers in an active state (online, offline or quiesced).
Privileged Clients	Number of clients that have been granted root privileges.
Software Version	The next software version to which the cluster is set to move.
Committed Software Version	The current software version on the cluster. Note: If an upgrade is in progress, the Operational Software Version will be an earlier version than the Pending Software Version .
Last Software Commit	Date and time of the last upgrade to the cluster software.

Software Commit Status

Indicates if an upgrade is “In progress,” or “Not in progress.”

Installation Date

Date and time that the Global (Root) Fileset was created (that is, when SAN File System was installed).

Related topics:

- “Cluster” on page 16
- “Cluster states” on page 19
- “Managing the cluster” on page 82
- “Metadata server” on page 33
- “Metadata server states” on page 35
- “Using the SAN File System console” on page 333

Cluster Properties - Tuning

Use this panel to set the storage pool-space-reclamation-interval for SAN File System.

Fields:

Storage Pool Space Reclamation Interval checkbox

Select the checkbox if you want would like to set how often SAN File System reclaims disk space allocated to storage pools.

Minutes fields

Specify the interval, in minutes, if the setting is enabled. (Range 1 to 1 440; Default = 60)

Related topics:

- “Cluster” on page 16
- “Managing the cluster” on page 82
- “Storage pools” on page 37
- “Using the SAN File System console” on page 333

Cluster Properties - Tuning Details

Use this panel to view additional SAN File System cluster tuning properties.

Attributes:

The **Cluster Tuning Details** table lists the following information.

Note: Except for the **Storage Pool Space Reclamation Interval**, the remaining parameters cannot be reconfigured using the SAN File System console. They are set to default values at installation and must be changed at the Administrative command-line interface.

Storage Pool Space Reclamation Interval (minutes)

How often SAN File System reclaims disk space allocated to storage pools. You can reset this parameter on the **Cluster Properties - Tuning** panel.

Client Lease Period (seconds)	The amount of time a Metadata server grants a client access to a lock. (Range = 10 to 120 seconds.) Note: The Metadata server also applies the multiplier to this value before the lease expires.
Client Lease Period Multiplier	Directly interacts with the lease period to accommodate possible networking delays between when a message gets sent from the client and when it is received by the Metadata server. (Range 0–4; with 0 meaning “disabled.”)
Master Server Buffer Size (pages)	Additional pages to serve as a space buffer on the master Metadata server. (Range is 2 048 to 8 192 pages. Installation default is 2 048.)
Subordinate Server Buffer Size (pages)	Additional pages to serve as a space buffer on the subordinate Metadata servers in the cluster. (Range is 30 000 to 250 000 pages. Installation default is 30 000.)
Maximum Network Heartbeats	The maximum number of heartbeats that can be missed before declaring a path failure on a subsequent missed heartbeat. A Metadata server will be ejected from the cluster if it exceeds any of the heartbeat thresholds. (Range is 1 to 100.) Note: When the maximum setting is reached, the Metadata server will go into a “Not running” state, if it was already up and running, and will remain that way until repaired. Otherwise, the ejected Metadata server will restart and rejoin the cluster.
Network Heartbeat Interval (msec)	The interval between sending heartbeats. (Range is 20 to 10 000 msec.)
Within-Cluster Timeout (msec)	The timeout period for intra-cluster communications. (Range 500 000 to 10 000 000 msec.)
Retries to Client	Maximum times the Metadata server will attempt to contact the client before declaring the client’s session lease to be expired. (Range is 1 to 100. Installation default is 5.)
Retries-to-Client Interval (msec)	The interval between each attempt by the Metadata server to contact the client before declaring the client’s session lease to be expired.

Related topics:

- “Backup and restore” on page 5
- “Cluster” on page 16
- “Cluster states” on page 19
- “Locks and leases” on page 30
- “Managing the cluster” on page 82
- “Metadata server” on page 33
- “Metadata server states” on page 35
- “Storage pools” on page 37
- “Clients” on page 11

- “Using the SAN File System console” on page 333
- “Viewing cluster tuning details” on page 85


Collect Diagnostic Data

Use this panel to view detailed information on engines to assist with monitoring and troubleshooting.

Note: The diagnostic data collection might take several minutes to complete.

Fields:

You can select one or more engines on which to view additional details. The **Engines** table lists the following information for each engine:

Engine IP	The IP address of the associated engine, which links to its summary notebook panel. The engine summary panel also provides status on temperature, voltage and fans.
Server	The Metadata server residing on the associated engine. This column also identifies the master-subordinate relationship among the Metadata servers. Note: The master Metadata server will have the  icon next to it.
Server State	The state of the engine’s associated Metadata server will be one of the following: <ul style="list-style-type: none"> • Not running • Failed initialization • Initializing • Not added • Joining • Offline • Fully quiescent • Partly quiescent • Online • Unknown

Related topics:

- “Changing the master Metadata server” on page 110
- “Engines” on page 23
- “Managing engines” on page 93
- “Managing Metadata servers” on page 109
- “Metadata server” on page 33
- “Metadata server states” on page 35
- Chapter 3, “Monitoring”, on page 57
- “Using the SAN File System console” on page 333

Collect Diagnostic Data Progress

Use this panel to view the progress of the engine diagnostic data collection task.

Fields:

Progress Status Area

Progress text

Displays one of the following, depending on the data collection task status:

- “The diagnostic data collection is in progress.”
- “The diagnostic data collection completed <time stamp>.”
- “The diagnostic data collection ended at <time stamp> and did not complete.”

Note: This last message appears if an error is encountered.

Progress bar

Animated graphic that continues until data collection completes for all engines.

Attributes:

The **Collections on Each Engine** table specifies where the data collections for the listed engines reside.

Diagnostic Data

Name of the output directory where the data is located.

Status Log

Name of the output directory where errors associated with the diagnostic data collection task are logged.

Attributes:

The **Collection Status** table lists the engines on which the data was collected, and the collection status for each.

Engine IP

IP address of the associated engine.

Server

The Metadata server residing on the associated engine.

Status

The data collection status for the associated engine. Four values are possible:

- Not Started
- In Progress
- Completed
- Failed

Fields:

Pushbuttons.

Close

Closes the panel.

Note: The close action does not stop the underlying data collection if it is currently running.

Related topics:

- “Engines” on page 23
- “Managing engines” on page 93
- “Managing Metadata servers” on page 109
- “Metadata server” on page 33
- “Using the SAN File System console” on page 333

Create a Fileset

Use this panel to create a new fileset (container) in SAN File System.

Attributes:

Name	Enter a name for the fileset.
Description	Enter a description for the fileset.
Server	<p>Assigns the fileset to a Metadata server. Use the drop-down list to view all of the servers known to SAN File System, regardless of their current state.</p> <p>Note: The Metadata server that you select to run the associated fileset must meet the following conditions:</p> <ul style="list-style-type: none"> • The target Metadata server must be part of the cluster. • The master Metadata server must not be down. • The global fileset must always be bound to the master Metadata server.
Quota Options	Set your preferred choices for the quota type, size, and alert triggers.
Quota Type	<p>Select one of two quota types:</p> <p>Soft quota Only issues an alert (warning level); it does not cut off additional storage use when reached. (Default setting)</p> <p>Hard quota Issues an alert (error level); and also cuts off additional storage use when reached.</p>
Quota Size	<p>You must choose a quota size and unit of measurement (MB, GB, TB, or PB) for the fileset.</p> <p>Note: The size cannot exceed 1 PB.</p>
Enable usage alerts checkbox	<p>Select to enable, or clear the checkbox to disable quota usage alerts. If enabled, usage-based alerts (warning level) will be generated per the usage threshold setting.</p>
Usage Threshold	<p>(Required if usage alerts are enabled.) Select a chosen used percentage of the fileset quota that when reached, will issue an alert.</p>
Attach Point	Path and directory location of the fileset within the global namespace.

Existing Directory Path	Specifies the existing directory path (relative to the root of the global namespace and without the / prefix) at which to attach the fileset. This directory must already exist on the client side. The root of the global namespace must be included in the directory path. Use only forward slashes (/) in the directory path for delimiters.
New Directory Name	Specifies the directory name for the root of the fileset directory tree. The directory name must not already exist on the client side. This directory can be up to 256 characters in length and must not contain backslash (\) or forward slash (/) characters as delimiters. Note: You must enter the new directory name when the fileset is created, moved or reattached.

Related topics:

- “Attaching a fileset” on page 103
- “Detaching a fileset” on page 105
- “Filesets” on page 24
- “Global namespace” on page 29
- “Using the SAN File System console” on page 333

Create FlashCopy Images of Filesets- Introduction

Use this wizard to create new SAN File System FlashCopy images. This three-step wizard will allow you to select the fileset (container) from which to create the FlashCopy image, set properties for the image, and verify your settings before it is created.

Notes:

1. After it is created, you cannot change the name, description, or the directory name of a FlashCopy image.
2. There is a limit of 32 FlashCopy images per fileset.
3. A fileset may be attached or detached when the FlashCopy image is created.
4. The **Next** button in the wizard will be disabled if any failures occur during the image creation process.

Related topics:

- “Attaching a fileset” on page 103
- “Filesets” on page 24
- “Creating a FlashCopy image” on page 107
- “Detaching a fileset” on page 105
- “FlashCopy images” on page 26
- “FlashCopy Images” on page 395
- “Reverting to a previous FlashCopy image” on page 108
- “Using the SAN File System console” on page 333

Create FlashCopy Images of Filesets- Select Filesets

This panel (first step in the wizard) allows you to choose one or more filesets (containers) for which to create the FlashCopy image.

Note: If you entered the wizard from the Filesets panel, the fileset or filesets that you chose before will be preselected.

Fields:

In addition to its description, the table provides the following information about each fileset:

Name	Links to additional details about the selected fileset.
State	Two fileset states are possible: <ul style="list-style-type: none">• Attached• Detached
Server	The Metadata server that is manually assigned to the fileset.
Quota (MB)	The quota for the fileset, displayed in megabytes.
Used (MB)	The number of megabytes used by the fileset.
Used (%)	System percentage used by the fileset.
Threshold (%)	Percentage of the quota that the fileset must reach before an alert will be issued. (Established when you create the fileset.) Note: For a soft quota, SAN File System will only issue an alert, not cut off additional storage use.
Most Recent Image	Date and time that the last FlashCopy image was taken for the associated fileset.

Related topics:

- “Attaching a fileset” on page 103
- “Filesets” on page 24
- “Creating a FlashCopy image” on page 107
- “Detaching a fileset” on page 105
- “FlashCopy images” on page 26
- “FlashCopy Images” on page 395
- “Reverting to a previous FlashCopy image” on page 108
- “Using the SAN File System console” on page 333

Create FlashCopy Images of Filesets- Set Properties

This panel (second step in the wizard) allows you to add detailed information about the FlashCopy image, including the image name, directory name, and description.

Attributes:

Force Image Creation

When selected, this setting deletes the oldest image to make room for a new one, if necessary.

Note: There is a limit of 32 FlashCopy images per fileset.

Attributes:

In addition to allowing you to enter an optional detailed description for the FlashCopy image, the **FlashCopy Images Settings per Fileset (Container)** section requires you to enter the following:

FlashCopy Image Name

Will be prefilled with the same name used in the Directory Name field. However, you can change the name if you want.

Directory Name

Will be prefilled with a name according to the following convention: "Image-<sequence number>." However, you can change the name if you want.

Related topics:

- "Attaching a fileset" on page 103
- "Filesets" on page 24
- "Creating a FlashCopy image" on page 107
- "Detaching a fileset" on page 105
- "FlashCopy images" on page 26
- "FlashCopy Images" on page 395
- "Reverting to a previous FlashCopy image" on page 108
- "Using the SAN File System console" on page 333

Create FlashCopy Images of Filesets- Verify Settings

This panel (final step in the wizard) allows you to confirm your selected configuration for the new FlashCopy image before it is created. All of the displayed fields are read-only. To make changes, return to any of the previous panels in the wizard.

Attributes:

The **FlashCopy Images to Be Created** table lists the fileset, directory name, and description (if entered) for the FlashCopy image. It also provides the following:

Name

This name will be applied to the new image that you are creating.

Related topics:

- "Attaching a fileset" on page 103
- "Filesets" on page 24
- "Creating a FlashCopy image" on page 107
- "Detaching a fileset" on page 105

- “FlashCopy images” on page 26
- “FlashCopy Images” on page 395
- “Reverting to a previous FlashCopy image” on page 108
- “Using the SAN File System console” on page 333

Create a Policy - Add Rules

This panel (second step in the wizard) allows you to add rules to a newly-created policy. File-placement rules are lists of conditions on file attributes. The ordered list of file-placement rules composes the policy. You can add any number of file placement rules to the policy.

Note: All date and time attributes in these rules are based in coordinated universal time (UTC). Therefore, any time and date entry is assumed to be in universal time. Also, functions (for example, “dayofweek”) are based on UTC.

Fields:

Rules Description

This optional description is added as a comment statement before each rule.

Action: Storage Pool Assignment

Assign the policy rule to a storage pool. The drop-down list presents all of the existing storage pools, except the system storage pool, which is reserved for system data.

Conditions

After the policy is activated, if one or more of the enabled conditions matches a new file being stored, it will be assigned to the storage pool that was selected from the **Storage Pool Assignment** drop-down list. Select one or more of the following conditions for the file placement rule:

File name

Places a condition on the file name. You can place a file based on a name with specified characters appearing at the beginning (“starts with”), end (“ends with”), or anywhere within it, including the beginning or the end (“contains”). You may also select the “is” option if you want to specify an exact file name, or “is not” if you are sure of what you do not want the name to contain for files assigned to the associated storage pool.

Notes:

1. You can use a % wildcard to represent one or more characters and use the _ wildcard to represent a single character.
2. Allowed characters include letters, numerals, %.
3. You can only specify one file name type per rule.
4. Multiple listings are not allowed.

File created

Places a condition on the year, month, day and hour the file was created.

Fileset (Container)

Places a condition based on the existing fileset to which the file is assigned.

Group ID (UNIX only)

For UNIX-based files. This condition only takes a numeric value. Valid range is 0 – 4 294 967 295 (10–digit maximum).

User ID (UNIX only)

For UNIX-based files. This condition only takes a numeric value. Valid range is 0 – 4 294 967 295 (10–digit maximum).

New Rule pushbutton

Clicking **New Rule** saves the rule that you just created and clears the panel to build another rule.

Notes:

1. You can click **Back** to go back to previous steps, but you cannot go back to previous rules.
2. You can advance to the next step (the rules editor), and bypass the **Add Rules** panel by clicking **Next**, regardless of whether or not any rules have been set. The rules editor allows you to perform alternate actions, such as pasting in a rule set.

Related topics:

- “File placement” on page 43
- “Policies and rules” on page 44
- “Using the SAN File System console” on page 333

Create a Policy - Edit Rules

This panel (final step in the wizard) allows you to edit rules of a newly created policy or cloned policy.

Note: All date and time attributes in these rules are based in coordinated universal time (UTC). Therefore, any time and date entry is assumed to be in universal time. Also, functions (for example, “dayofweek”) are based on UTC.

Attributes:

Policy Settings

Read-only fields indicating the assigned **Name** and **Description** of the policy.

Attributes:

Rules

Editable text area, for making changes to the rules. You can perform basic editing tasks, such as cut, copy, and paste.

Related topics:

- “File placement” on page 43
- “Policies and rules” on page 44
- “Using the SAN File System console” on page 333

Create a Policy - High-Level Settings

This panel (first step in the wizard) allows you to enter a name and description for the policy, as well as select a policy creation method.

Attributes:

Name

Enter a name for the policy (required).

Description

Enter an optional detailed description for the policy .

Attributes:

Policy Creation Method

New Policy

If you choose to create a new policy, you will be taken to another panel for specifying the policy rules.

Clone Policy

If you choose to clone a policy, you must also choose an **Existing Policy** from the drop-down list.

Related topics:

- “File placement” on page 43
- “Policies and rules” on page 44
- “Using the SAN File System console” on page 333

Create a Policy - Introduction

Use this wizard to create new file-placement policies, or clone existing policies in SAN File System. This three-step wizard (two steps if you are cloning) will allow you to add and edit rules to define the policy.

Note: The **Next** button in the wizard will be disabled if any failures occur during the policy creation process.

Related topics:

- “File placement” on page 43
- “Policies and rules” on page 44
- “Using the SAN File System console” on page 333

Create Recovery File

Use this panel to create a new disaster recovery file, or to overwrite an existing one. The recovery file will contain information that can aid in restoring the metadata (that is, the storage pools, volumes, filesets and policies).

Actions:

Creation Method

Create

You can choose to create a new recovery file and type the new file name.

Forced Create

Select an existing file from the drop-down list to overwrite.

Related topics:

- “Creating a recovery file” on page 87
- “Checking metadata” on page 112
- “Deleting a recovery file” on page 87
- “Using the SAN File System console” on page 333

Create a Storage Pool - Add Volumes

This panel (second step in the wizard) allows you to add volumes to a newly-created storage pool. You can add volumes to the storage pool by selecting from the available LUNs.

Attributes:

Volume Name Prefix	(Required entry field if any LUNs are selected.) The prefix entered here is added to the LUN ID. For example, if you enter the prefix "ABC," your volumes may become ABC-4, ABC-4, and so forth. You should use a prefix that will correspond to the storage device assigned to the LUN for easy association.
Initially activate volumes	Note: You can later rename the volumes. Selecting this option will add the volumes and activate them when added. Otherwise, the volumes will be added in a suspended state, and data cannot be written to them.
Forced Addition	When not selected, this setting will ensure that the volumes are not currently assigned in the associated cluster.

Attributes:

Available LUNs table

This table displays the available LUNs from which you select to write to the storage pool as volumes. It displays the size of the LUN in megabytes, as well as the following:

LUN ID	Links to additional details about the selected LUN.
Vendor	The vendor of the storage device associated with the LUN. Note: The system displays "-" if the vendor cannot be determined.
Product	The product name of the storage device associated with the LUN. Note: The system displays "-" if the product cannot be determined.
Storage Device WWN	Uniquely identifies the storage device by its World Wide Name (WWN). This information is helpful if you want to ensure that all LUNs in a storage pool are from the same storage device.

Related topics:

- "Storage pools" on page 37
- "Storage devices" on page 46
- "Volumes" on page 49
- "Using the SAN File System console" on page 333

Create a Storage Pool - Introduction

Use this wizard to create new user storage pools in SAN File System. This three-step wizard will allow you to set the properties of the new storage pool and add volumes to it.

Related topics:

- “Storage pools” on page 37
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Create a Storage Pool - Set Properties

This panel (first step in the wizard) allows you to specify a name (required), description (optional) and partition size for the storage pool, as well as a percentage-used-threshold alert.

Actions:

Name	Enter a name for the storage pool (required).
Description	Enter an optional detailed description for the storage pool.
Partition size	Select from 16, 64, or 256 MB as the partition size of the new storage pool. (Default = 16 MB) Note: You cannot change the partition size after it is set.
Allocation size	Select from “System Automated,” “4 KB Fixed,” or “128 KB Fixed.” Note: You cannot change the allocation size after it is set.
Enable usage alerts checkbox	You can optionally select to enable an alert when the storage pool partition reaches a particular percentage used.
Usage Threshold	The storage pool usage threshold at which point an alert will be issued if you enabled usage alerts. (Default = 80%)

Related topics:

- “Storage pools” on page 37
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Create a Storage Pool - Verify Settings

This panel (final step in the wizard) allows you to verify your selected configuration for a newly-created storage pool. All of the displayed fields are read-only. To make changes, return to any of the previous panels in the wizard.

Attributes:

The **Storage Pool Properties** table lists the name, description, and partition size of the new storage pool, as well as the following:

Allocation Size (KB)	The storage pool allocation size that you set in the previous panel of the wizard. (The choices were "System Automated," "4 KB Fixed," or "128 KB Fixed.") Note: You cannot change the allocation size after it is set. If you want to change the setting, then return to the previous step in the wizard and modify it before clicking Finish .
Usage Threshold (%)	(If enabled in the Set Properties portion of the wizard.) When the amount used of the storage pool reaches this percentage of the total allotted to its partition, the system will issue an alert. (Default = 80%)

Attributes:

For each volume associated with the newly created storage pool, the **New Volumes** table lists the size of the volume in megabytes, as well as the LUN ID associated with the volume. The table also lists the following information:

Volume Name	The name that will applied to the new volume.
Initial Status	The status can be either Active or Suspended . Note: If volumes are in a suspended state, data cannot be written to them. You can change a volume to the active state by going back to the previous step of the wizard (Add Volumes) and selecting the Initially activate volumes checkbox.

Related topics:

- "Storage pools" on page 37
- "Volumes" on page 49
- "Using the SAN File System console" on page 333

Delete Filesets

Use this panel to verify the removal of filesets from SAN File System, and choose a method of deletion (regular or forced).

Fields:

The **Filesets Selected for Deletion** table displays the name and attach point for the associated fileset or filesets. It also displays the following:

State	One of two fileset states are possible: <ul style="list-style-type: none"> • Attached • Detached
Used (MB)	The number of megabytes used by the fileset.
Children	Number of subfilesets. (If there are no subfilesets, "-" is displayed.)

Related topics:

- “Attaching a fileset” on page 103
- “Detaching a fileset” on page 105
- “Filesets” on page 24
- “Using the SAN File System console” on page 333

Delete FlashCopy Images

Use this panel to review and verify the image or images that you have selected for deletion. You can also select a deletion method (regular or forced).

Note: You cannot delete a FlashCopy image that has client activity (session locks open) unless it is a forced delete operation.

Attributes:

The **Images Selected for Deletion** table displays the name and description of each FlashCopy image so that you can verify your choices before they are deleted. It also lists the following detail for each selected image:

Name	Name of the FlashCopy image of the associated fileset.
Fileset (Container)	Fileset that the image represents.
Directory Name	Directory name where the image is located. Note: The FlashCopy image directory is the one below the “.flashcopy” directory of the fileset.
Date and Time	Indicates when the FlashCopy image was taken.

Related topics:

- “Attaching a fileset” on page 103
- “Filesets” on page 24
- “Creating a FlashCopy image” on page 107
- “Detaching a fileset” on page 105
- “Locks and leases” on page 30
- “FlashCopy images” on page 26
- “FlashCopy Images” on page 395
- “Reverting to a previous FlashCopy image” on page 108
- “Using the SAN File System console” on page 333

Delete Storage Pools

Use this panel to review and verify the storage pool or storage pools that you have selected for deletion.

Attributes:

The **Storage Pools Selected for Deletion** table displays the name and description of each storage pool so that you can verify your choices before they are deleted. It also lists the following detail for each selected storage pool:

Type	Displays the storage pool type or types that you selected for deletion. Note: You can only delete user storage pools. If you attempt to delete the user default or system storage pool, you will get an error message.
Size (MB)	The total space available in the associated storage pool.
Used (MB)	The amount of space being used in the storage pool.
Used (%)	The percentage of the total storage pool size that is currently being used.
Volumes	Number of volumes in the associated storage pool.

Related topics:

- “Storage pools” on page 37
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Detach Filesets

Use this panel to verify the filesets (containers) to detach, as well as a method of detachment (regular or forced).

Fields:

In addition to the system name and amount used for the fileset or filesets to be detached, the **Detach Fileset (Containers)** table lists the following:

State	The fileset will be in one of two states: <ul style="list-style-type: none"> • Attached • Detached
Children	The number of subfilesets, if any. (If there are no subfilesets, “-” is displayed.)
Attach Point	Path and directory location of the fileset in the global namespace.

Related topics:

- “Attaching a fileset” on page 103
- “Filesets” on page 24
- “Global namespace” on page 29
- “Using the SAN File System console” on page 333

Details of Client Session

Use this panel to view additional information about client sessions, including transactions and locks.

Attributes:

The **Details** table lists the client IP address, port, OS Platform, and the following information:

Client	Identifies each client by its name. This name is given on the client side. Note: It is not an IP host name.
Session ID	Unique identifier for the client session. Note: A single client may have multiple, simultaneous sessions – one per Metadata server.
State	Two client session states are possible: Expired Lease is expired. Current Lease is currently valid. Note: A client might also not be connected, in which case, neither state applies.
Server Name	Name of the Metadata server that is part of the client session.
Renewals	A count of session renewals that increases once per renewal. The renewal gives you an indication of the length of each client session.
Last Renewal	Date and time (system, not local) of the last renewal.
Next Renewal (msec)	Milliseconds until next session renewal is to occur.
Client Privilege	Refers to the client root privileges, which can be: <ul style="list-style-type: none"> • Root (privileged client) • Standard (normal client)
Client File System Version	The version includes the client level and the SAN File System Protocol level in parentheses.
Transactions Started	Transactions started during the client session.
Transactions Completed	Transactions completed per the client session.
Session Locks	The current number of session locks for the client session. Clients must acquire session locks to do any operation with a filename.
Data Locks	The current number of data locks held for the client session. Clients must hold data locks in order to cache data pages, attributes of files, and to cache read-only attributes and contents of directories and links.
Byte Range Locks	The current number of byte range locks held for the client session. These locks are used to implement lock system calls and have no direct effect on the contents or attributes of SAN File System objects or other file system operations.

Related topics:

- “Clients” on page 11
- “Locks and leases” on page 30
- “Metadata server” on page 33
- “Using the SAN File System console” on page 333

Details of Image

Use this panel to view additional FlashCopy image attributes.

Attributes:

The **Details of Image** table displays the description of the FlashCopy image, as well as the following information:

Name	Name of the FlashCopy image of the associated fileset.
Fileset (Container)	Fileset that the image represents.
Fileset (Container) state	Two fileset states are possible: <ul style="list-style-type: none">• Attached• Detached
Directory Name	Directory name where the image is located. Note: The FlashCopy image directory is the one below the “.flashcopy” directory of the fileset.
Directory Path	Path where the FlashCopy image directory is located, using the following naming convention: <ul style="list-style-type: none">• the fileset’s attach point• “/.flashcopy”• /Directory name
Date and Time	Indicates when the FlashCopy image was taken.

Related topics:

- “Attaching a fileset” on page 103
- “Filesets” on page 24
- “Creating a FlashCopy image” on page 107
- “Detaching a fileset” on page 105
- “FlashCopy images” on page 26
- “Reverting to a previous FlashCopy image” on page 108
- “Using the SAN File System console” on page 333

Details of LUN

Use this panel to view additional LUN attributes.

Attributes:

The **Details of LUN** table displays the following information about the selected LUN:

LUN ID	Links to additional details about the associated LUN.
Volume Name	If the LUN is a volume assigned by SAN File System, the volume name will be listed here. Note: The system displays “-” if the LUN has not been assigned as a volume.

State	<p>Five LUN states are possible:</p> <p>Assigned Already assigned to a SAN File System storage pool as a usable volume.</p> <p>Available Usable LUNs that can be seen by SAN File System, but are not currently assigned to a storage pool.</p> <p>Error Indicates that there was an error in determining the properties of the LUN.</p> <p>Unknown Indicates that the Metadata server is not running, so a LUN state could not be determined (that is, the property could not be queried).</p> <p>Unusable LUNs that are unsuitable to be a SAN File System volume. Possible reasons that a LUN might be unusable include:</p> <ul style="list-style-type: none"> • The (inherited) access property shows that the LUN does not support read or write operations. • Inconsistent availability of a LUN from all servers in the cluster. • Disk or driver configuration errors. • A LUN type that is not recognizable to the system. <p>Note: The unusable state takes precedence over the either the assigned or available states.</p>												
Size (MB)	Total size of the LUN in megabytes.												
Vendor	The vendor of the storage device associated with the LUN. Note: The system displays “-” if the vendor cannot be determined.												
Product	<p>The product name of the storage device associated with the LUN. The system displays the following IBM products in addition to other products:</p> <table border="0"> <tr> <td>2105F20</td> <td>ESS F20</td> </tr> <tr> <td>2105800</td> <td>ESS 800</td> </tr> <tr> <td>2145</td> <td>SAN Volume Controller and SAN Integration Server</td> </tr> <tr> <td>3552</td> <td>FAStT 5000</td> </tr> <tr> <td>1722</td> <td>FAStT 600</td> </tr> <tr> <td>1742</td> <td>FAStT 700 and 900</td> </tr> </table> <p>Note: The system displays “-” if the product cannot be determined.</p>	2105F20	ESS F20	2105800	ESS 800	2145	SAN Volume Controller and SAN Integration Server	3552	FAStT 5000	1722	FAStT 600	1742	FAStT 700 and 900
2105F20	ESS F20												
2105800	ESS 800												
2145	SAN Volume Controller and SAN Integration Server												
3552	FAStT 5000												
1722	FAStT 600												
1742	FAStT 700 and 900												
Storage Device WWN	Uniquely identifies the storage device by its World Wide Name (WWN). This information is helpful if you want to ensure that all LUNs in a storage pool are from the same storage device.												
OS Device Path	The path that points to the SAN Adapter driver. Note: This driver is local to the SAN File System cluster and is not on the LUN storage device.												

Sector Size (Bytes)

Size of the areas, given in bytes, where blocks of data are physically stored on the LUN's associated storage device.

Related topics:

- "Managing volumes and LUNs" on page 128
- "Storage pools" on page 37
- "Storage devices" on page 46
- "Viewing available LUN details" on page 134
- "Viewing LUN details" on page 134
- "Volumes" on page 49
- "Using the SAN File System console" on page 333

Disaster Recovery

Use this panel to create or delete a disaster recovery file. The recovery file will contain information that can aid in restoring the metadata (that is, the storage pools, volumes, filesets, and policies).

Attributes:

Recovery Directory

Directory on each Metadata server where the recovery files are located. This directory is the same for all servers.

Build Script

The script used to generate command files from the metadata dump file.

Actions:

Create

Creates a new recovery file or overwrites an existing one.

Delete

Deletes one or more selected recovery files.

Fields:

You can select an existing recovery file to delete from the **Recovery Files** table. The table lists the following details:

Name

Name of the associated recovery file.

Date and Time

Indicates when the file was created.

Size (KB)

The size of the associated recovery file, in kilobytes.

Related topics:

- "Creating a recovery file" on page 87
- "Checking metadata" on page 112
- "Deleting a recovery file" on page 87
- "Using the SAN File System console" on page 333

Download Client Software

Use this panel to download and update your local SAN File System client software for Windows® or AIX® operating systems.

Actions:

Windows® Client Software

Name	Alphabetical listing of the Windows client file names, which are linked to the master Metadata server's corresponding file. The browser will let you download the file to your local machine.
Size	Size of the associated file, in kilobytes.

Actions:

AIX Client Software

Name	Alphabetical listing of the AIX client file names, which are linked to the master Metadata server's corresponding file. The browser will let you download the file to your local machine.
Size	Size of the associated file, in kilobytes.

Related topics:

- "Clients" on page 11
- "Using the SAN File System console" on page 333


Engines

Use this panel to view engine properties or change the state of an engine.

Actions:

Properties	Displays additional detailed properties of the selected engine, including information about temperatures, fans, voltages, timeouts, vital engine data, power, and an engine summary.
Change Power State	Allows you to change the power state of the selected engine or engines, as well as choose a change method.

Fields:

Engine IP	IP address of the engine, which also links to the engine's Summary panel.
Server	Metadata server associated with the engine. The master Metadata server is indicated by the  icon.

Boot State

Eight engine boot states are possible:

- “Unknown / Power Off”
- “Before POST” (Engine is powered on, but has not started POST.)
- “In POST” (Engine is in the Power On Self Test)
- “Stopped in POST” (Error detected.)
- “Booted Flash”(Engine booted from flash memory.)
- “Booting OS”
- “In OS” (The normal state.)
- “CPUs Held in Reset”

Temperature

Six engine temperature values are possible:

“Normal”

The temperature for all engine components that have thresholds are below the warning-level threshold.

Note: If no thresholds have been set, then “-” is displayed.

“Unknown”

The temperature state cannot be determined. Potential reasons include; failure; misconfiguration; or the absence of the RSA card.

Undefined

Undefined values were returned by the RSA card provider.

“Warning”

The temperature of one or more engine components is between the warning level threshold and the soft shutdown level threshold.

“Error” The temperature of one or more engine components is above the soft shutdown level threshold.

Fans

Four engine fan values are possible:

“Normal”

All fans are running in the normal range (15 % or greater speed).

“Unknown”

The fan state cannot be determined. Potential reasons include: failure; misconfiguration; or the absence of the RSA card.

Undefined

Undefined values were returned by the RSA card provider.

“Warning”

One or more fans is running below 15% of potential speed.

Voltage

Five engine voltage values are possible:

“Normal”

The voltages for all engine components that have thresholds are below the warning-level threshold.

Note: If no thresholds have been set, then “-.” is displayed.

“Unknown”

The voltage state cannot be determined. Potential reasons include: failure; misconfiguration; or the absence of the RSA card.

Undefined

Undefined values were returned by the RSA card provider.

“Warning”

The voltage for one or more engine components is either below the low voltage warning threshold, or is above the high voltage warning threshold

Related topics:

- “Alerts” on page 4
- “Metadata server” on page 33
- “Engines” on page 23
- “Using the SAN File System console” on page 333

Engines Properties - Fans

Use this panel to view the state and speed status for each fan of the selected engine.

Fields:

Fan

State

Each engine has eight fans.

The current state of the associated engine fan. Four engine fan values are possible:

“Normal”

All fans are running in the normal range (15 % or greater speed).

“Unknown”

The fan state cannot be determined. Potential reasons include: failure; misconfiguration; or the absence of the RSA card.

Undefined

Undefined values were returned by the RSA card provider.

“Warning”

One or more fans is running below 15% of potential speed.

Speed (%)

Fan speed in relationship to its expected speed. Absolute fan speed (RPMs) will vary by engine depending on the current environmental situation. For example, if one fan goes out, the remaining fans speed up.

Related topics:

- “Alerts” on page 4
- “Engines” on page 23
- “Using the SAN File System console” on page 333

Engines Properties - Power

Use this panel to view power details, or to change the power state of the associated engine.

Attributes:

Power

Current power state of the associated engine. The power state will be either “Off” or “On.”

Power-on Hours

Number of hours the engine has been powered on.

Restart Count

Number of times the engine has been restarted.

Current ASM Time

Note: This counter is cleared when the Advanced Systems Management (ASM) subsystem is cleared to factory defaults.

This is the current time on the local clock for the Advanced Systems Management device.

This time reference is used to schedule a “power-off.” This time is independent of the time on the engine.

Actions:

Change Power State pushbutton

Takes you to the **Change Power State of Engine** panel for the single engine in context. This panel enables you to verify the engine selected for a power state change and choose a change method.

Related topics:

- “Engines” on page 23
- “Using the SAN File System console” on page 333


Engines Properties - Summary

Use this panel to view high-level information about the selected SAN File System engine, including the associated Metadata server, IP address, and state.

Attributes:

Engine IP

IP address of the engine.

Server	Metadata server associated with the engine. The master Metadata server is indicated by the  icon.
Boot State	<p>Eight engine boot states are possible:</p> <ul style="list-style-type: none"> • “Unknown / Power Off” • “Before POST” (Engine is powered on, but has not started POST.) • “In POST” (Engine is in the Power On Self Test) • “Stopped in POST” (Error detected.) • “Booted Flash” (Engine booted from flash memory.) • “Booting OS” • “In OS” (The normal state.) • “CPUs Held in Reset”
Temperature	<p>Six engine temperature values are possible:</p> <p>“Normal” The temperature for all engine components that have thresholds are below the warning-level threshold. Note: If no thresholds have been set, then “-” is displayed.</p> <p>“Unknown” The temperature state cannot be determined. Potential reasons include; failure; misconfiguration; or the absence of the RSA card.</p> <p>Undefined Undefined values were returned by the RSA card provider.</p> <p>“Warning” The temperature of one or more engine components is between the warning level threshold and the soft shutdown level threshold.</p> <p>“Error” The temperature of one or more engine components is above the soft shutdown level threshold.</p>

Voltage

Five engine voltage values are possible:

“Normal”

The voltages for all engine components that have thresholds are below the warning-level threshold.

Note: If no thresholds have been set, then “-.” is displayed.

“Unknown”

The voltage state cannot be determined. Potential reasons include: failure; misconfiguration; or the absence of the RSA card.

Undefined

Undefined values were returned by the RSA card provider.

“Warning”

The voltage for one or more engine components is either below the low voltage warning threshold, or is above the high voltage warning threshold

Fans

Four engine fan values are possible:

“Normal”

All fans are running in the normal range (15 % or greater speed).

“Unknown”

The fan state cannot be determined. Potential reasons include: failure; misconfiguration; or the absence of the RSA card.

Undefined

Undefined values were returned by the RSA card provider.

“Warning”

One or more fans is running below 15% of potential speed.

Related topics:

- “Alerts” on page 4
- “Metadata server” on page 33
- “Engines” on page 23
- “Using the SAN File System console” on page 333

Engines Properties - Temperatures

Use this panel to view component temperatures and temperature thresholds for the selected engine.

Fields:

The **Component Temperatures** table displays the following information:

Component	<p>There are five engine temperature components:</p> <ul style="list-style-type: none">• CPU 1• CPU 2• DASD 1 – direct access storage device (first hard drive)• DASD 2 – direct access storage device (second hard drive)• Ambient – Overall temperature
State	<p>The temperature state of the associated engine component. Six engine temperature values are possible:</p> <p>“Normal” The temperature for all engine components that have thresholds are below the warning-level threshold. Note: If no thresholds have been set, then “-” is displayed.</p> <p>“Unknown” The temperature state cannot be determined. Potential reasons include; failure; misconfiguration; or the absence of the RSA card.</p> <p>Undefined Undefined values were returned by the RSA card provider.</p> <p>“Warning” The temperature of one or more engine components is between the warning level threshold and the soft shutdown level threshold.</p> <p>“Error” The temperature of one or more engine components is above the soft shutdown level threshold.</p>
Temperature (°C)	<p>The actual engine temperature of the associated engine component, in degrees Celsius.</p>

Fields:

The **Temperature Thresholds** table displays the following information:

Component	<p>The following four engine components with temperature thresholds are listed:</p> <ul style="list-style-type: none">• CPU 1• CPU 2• DASD 1 – direct access storage device (first hard drive)• DASD 2 – direct access storage device (second hard drive)
-----------	--

Warning Reset (°C)	Temperature threshold, for the associated engine component, at which the warning will be reset.
Warning (°C)	Temperature threshold for issuing a warning. Temperature threshold, for the associated engine component, at which a warning is issued.
Soft Shutdown (°C)	Temperature threshold, for the associated engine component, at which a soft shutdown will occur. Note: A soft, or graceful, shutdown cleanly synchronizes everything before shutting off.
Hard Shutdown (°C)	Temperature threshold, for the associated engine component, at which a hard shutdown will occur. Note: A hard, or forced, shutdown stops without a clean synchronization.

Related topics:

- “Alerts” on page 4
- “Engines” on page 23
- “Using the SAN File System console” on page 333

Engines Properties - Timeouts

Use this panel to view timeout and Metadata server restart service details for the associated engine.

Attributes:

POST Server Restart Service (seconds)	The time, in seconds, that the Metadata server restart service will wait for Power On Self Test (POST) to complete before sending an alert and auto-restarting the engine. If this feature is disabled, “-” is displayed.
Loader Server Restart Service (seconds)	The time, in seconds, that the Metadata server restart service will wait for the operating system (OS) load to complete before sending an alert and auto-restarting the engine. If this feature is disabled, “-” is displayed.
OS Server Restart Service Timeout (seconds)	The total time, in seconds, that the Metadata server restart service will wait for the OS to respond before sending an alert and auto-restarting the engine. If this feature is disabled, “-” is displayed.
OS Server Restart Service Check Interval (seconds)	The time, in seconds, between the OS Metadata server restart service timeout checks. If this feature is disabled, “-” is displayed.
Power Off Delay (seconds)	The time, in seconds, that the engine will wait for the OS to shutdown before turning off the system (default is 30 seconds). If this feature is disabled, “-” is displayed.

Related topics:

- “Alerts” on page 4

- “Metadata server” on page 33
- “Engines” on page 23
- “Using the SAN File System console” on page 333

Engines Properties - Vital Engine Data

Use this panel to view model, serial number, and firmware details for the associated engine.

Attributes:

Machine Model Number	Identifies the engine model number.
Machine Serial Number	Identifies the engine serial number.
Firmware Revision	Identifies the revision numbers and vital product data of the application firmware and the startup read-only memory (ROM) firmware.
Firmware Revision Date	The firmware revision date for the logical device.
Firmware File Name	The file names and vital product data of the application firmware and the startup ROM firmware.
Firmware Build ID	The build IDs and vital product data of the application firmware and the startup ROM firmware.
Universal Unique Identifier	Number that uniquely identifies the machine.

Related topics:

- “Engines” on page 23
- “Using the SAN File System console” on page 333

Engines Properties - Voltages

Use this panel to view component voltages and system board voltage thresholds for the selected engine.

Fields:

The **Component Voltages** table displays the following information:

Component	<p>There are seven engine voltage components:</p> <ul style="list-style-type: none"> • Voltage Regulator Module (VRM) 1 (1.5 Volt) • Voltage Regulator Module (VRM) 2 (1.5 Volt) • System Board (SB): 12 Volt • System Board (SB): 5 Volt • System Board (SB): 3.3 Volt • System Board (SB): 2.5 Volt • System Board (SB): 1.5 Volt
-----------	--

State	<p>The voltage state of the associated engine component. Five engine voltage values are possible:</p> <p>“Normal” The voltages for all engine components that have thresholds are below the warning-level threshold. Note: If no thresholds have been set, then “-.” is displayed.</p> <p>“Unknown” The voltage state cannot be determined. Potential reasons include: failure; misconfiguration; or the absence of the RSA card.</p> <p>Undefined Undefined values were returned by the RSA card provider.</p> <p>“Warning” The voltage for one or more engine components is either below the low voltage warning threshold, or is above the high voltage warning threshold</p>
Voltage (V)	The voltage of the associated engine component, listed in volts.

Fields:

The **System Board Voltage Thresholds** table displays the following information:

Component	<p>The following engine system board voltage components with thresholds are listed:</p> <ul style="list-style-type: none"> • 12 Volt • 5 Volt • 3.3 Volt • 2.5 Volt • 1.5 Volt
Warning Low	Component voltage threshold at which a low voltage warning will be issued.
Warning High	Component voltage threshold at which a high voltage warning will be issued.

Related topics:

- “Alerts” on page 4
- “Engines” on page 23
- “Using the SAN File System console” on page 333

Fileset Properties - Details

Use this panel to view additional information about the selected SAN File System fileset (container).

Attributes:

The fileset (container) **Details** table lists the fileset's name, directory name, directory path, and description. The table also displays the following information:

State	Two fileset states are possible: <ul style="list-style-type: none"> • Attached • Detached
Server	Name of the Metadata server that is running the selected fileset.
Serving State	Indicates whether or not the fileset is currently being served. Two states are possible: <ul style="list-style-type: none"> • Online <p>Note: This state means that the Metadata server is online and the fileset is attached.</p> • Offline
Quota Type	Indicates whether the quota type is soft (alert only) or hard (alert and storage cutoff).
Quota (MB)	The fileset's quota displayed in megabytes.
Used (MB)	The number of megabytes used by the fileset.
Used (%)	System percentage used by the fileset.
Threshold (%)	Percentage used of the quota that the fileset must reach before an alert will be issued.
Attach Point	Path and directory location of the fileset within the global namespace. (Will show "-" if detached.)
Parent	Name of the parent fileset. (If there is none, "-" is displayed.)
Children	Number of subfilesets. (If there are no subfilesets, "-" is displayed.)
FlashCopy Images	Number of FlashCopy images. (Range is from 0 to 32.)
Most Recent Image	Date and time that the last FlashCopy image was taken for the associated fileset.

Related topics:

- "Attaching a fileset" on page 103
- "Detaching a fileset" on page 105
- "Filesets" on page 24
- "Global namespace" on page 29
- "FlashCopy images" on page 26
- "Reverting to a previous FlashCopy image" on page 108
- "Using the SAN File System console" on page 333

Fileset Properties - General Settings

Use this panel to change the fileset (container) name, description, Metadata server, quota, and alert settings.

Attributes:

Name	Enter a name for the fileset (required).
Description	Enter a description for the fileset (optional).

Server	<p>Displays the name of the Metadata server that is currently running the selected fileset. Use the drop-down list to view all of the servers known to SAN File System, regardless of their current state.</p> <p>Note: You can also select a different Metadata server to run the associated fileset (whether attached or detached) as long as the Metadata server meets the following conditions:</p> <ul style="list-style-type: none"> • The target Metadata server must be part of the cluster. • The master Metadata server must not be down. • The global fileset must always be bound to the master Metadata server. • The current Metadata server must not be online, fully quiescent, or partly quiescent. A recommended action would be to place the Metadata server offline or turn it off.
Quota Options	Set your preferred choices for the quota type, size, and alert triggers.
Quota Type	<p>Select one of two quota types:</p> <p>Soft quota Only issues an alert (warning level); it does not cut off additional storage use when reached. (Default setting)</p> <p>Hard quota Issues an alert (error level); and also cuts off additional storage use when reached.</p>
Quota Size	<p>You must choose a quota size and unit of measurement (MB, GB, TB, or PB) for the fileset.</p> <p>Note: The size cannot exceed 1 PB.</p>
Enable usage alerts checkbox	Select to enable, or deselect to disable quota usage alerts. If enabled, usage-based alerts (warning level) will be generated per the usage threshold setting.
Usage Threshold	(Required if usage alerts are enabled.) Select a chosen used percentage of the fileset quota that, when reached, will issue an alert.

Related topics:

- “Metadata server” on page 33
- “Cluster” on page 16
- “Filesets” on page 24
- “Global namespace” on page 29
- “Metadata server states” on page 35
- “Using the SAN File System console” on page 333

Filesets

Use this panel to create, attach, detach, delete, and view properties for filesets (containers). You can also create and manage FlashCopy images of filesets.

Actions:

Create	Takes you to a one-step process to create a fileset.
Properties	Displays, and allows you to set, certain properties for the selected fileset or filesets, such as a percentage used threshold alert.
Delete	Deletes the selected fileset from SAN File System.
Attach	Attaches the fileset by allowing you to specify a path and directory location for it within the global fileset.
Detach	Verifies the fileset or filesets to be detached, and allows you to choose a detachment method (normal or forced).
Manage FlashCopy Images	Allows you to view and perform actions on the FlashCopy images of the selected fileset.
Create FlashCopy Images	Creates a FlashCopy image of the selected fileset or filesets.

Fields:

You can select a fileset for which to view additional details or on which to perform actions. In addition to its description, the table provides the following information about each fileset:

Name	Links to additional details about the selected fileset.
State	Two fileset states are possible: <ul style="list-style-type: none">• Attached• Detached
Server	The Metadata server that is manually assigned to the fileset.
Quota (MB)	The quota for the fileset in megabytes.
Used (MB)	The number of megabytes used by the fileset.
Used (%)	System percentage used by the fileset.
Threshold (%)	Percentage of the quota that the fileset must reach before an alert will be issued. (Established when you create the fileset.) Note: For a soft quota, SAN File System will only issue an alert, not cut off additional storage use.
Most Recent Image	Date and time that the last FlashCopy image was taken for the associated fileset.

Related topics:

- “Attaching a fileset” on page 103
- “Detaching a fileset” on page 105
- “Filesets” on page 24
- “Global namespace” on page 29

- “FlashCopy images” on page 26
- “Reverting to a previous FlashCopy image” on page 108
- “Using the SAN File System console” on page 333

FlashCopy Images

Use this panel to manage SAN File System FlashCopy images. You can create, delete, and view details for FlashCopy images, as well as revert a fileset (container) to a FlashCopy image.

Actions:

Create	Takes you to a wizard for creating a new FlashCopy image.
Delete	Deletes the selected FlashCopy image. Note: You cannot delete a FlashCopy image that has client activity (that is, session locks open), unless it is a forced delete.
Revert to	Reverts the associated fileset to the FlashCopy image that you select. Note: If the associated fileset has child filesets, they must first be detached before you can revert the parent fileset to the selected image.
Details	Displays additional details about the selected FlashCopy image, such as the associated fileset state and the image’s directory path.

Fields:

You can select a FlashCopy image on which to view additional details or perform actions. In addition to its description, the table provides the following information about each FlashCopy image:

Name	Name of the FlashCopy image of the associated fileset.
Filesets (Containers)	Fileset that the image represents.
Directory Name	Directory name where the image is located. Note: The FlashCopy image directory is the one below the “.flashcopy” directory of the fileset.
Date and Time	Indicates when the FlashCopy image was taken.

Related topics:

- “Attaching a fileset” on page 103
- “Filesets” on page 24
- “Creating a FlashCopy image” on page 107
- “Detaching a fileset” on page 105
- “FlashCopy images” on page 26
- “Reverting to a previous FlashCopy image” on page 108
- “Using the SAN File System console” on page 333

LUNs

Use this panel to view detailed information for all LUNs that can be seen by SAN File System. Therefore, this panel displays LUNs that are read-only, those that have been assigned as SAN File System volumes, as well as those that are unassigned, but available.

Note: To view only writable LUNs that have not yet been assigned to a SAN File System storage pool, select the **Available LUNs** panel.

Actions:

Details	Displays additional LUN details, such as the Volume Name, State, Port WWN, OS Device Path, and Sector Size.
---------	---

Fields:

LUN ID	Links to additional details about the associated LUN.
Volume Name	If the LUN is a volume assigned by SAN File System, the volume name will be listed here. Note: The system displays “-” if the LUN has not been assigned as a volume.
State	Five LUN states are possible: Assigned Already assigned to a SAN File System storage pool as a usable volume. Available Usable LUNs that can be seen by SAN File System, but are not currently assigned to a storage pool. Error Indicates that there was an error in determining the properties of the LUN. Unknown Indicates that the Metadata server is not running, so a LUN state could not be determined (that is, the property could not be queried). Unusable LUNs that are unsuitable to be a SAN File System volume. Possible reasons that a LUN might be unusable include: <ul style="list-style-type: none">• The (inherited) access property shows that the LUN does not support read or write operations.• Inconsistent availability of a LUN from all servers in the cluster.• Disk or driver configuration errors.• A LUN type that is not recognizable to the system. Note: The unusable state takes precedence over the either the assigned or available states.
Size (MB)	Total size of the LUN in megabytes.
Vendor	The vendor of the storage device associated with the LUN. Note: The system displays “-” if the vendor cannot be determined.

Product	The product name of the storage device associated with the LUN. The system displays the following IBM products in addition to other products: <table> <tr> <td>2105F20</td> <td>ESS F20</td> </tr> <tr> <td>2105800</td> <td>ESS 800</td> </tr> <tr> <td>2145</td> <td>SAN Volume Controller and SAN Integration Server</td> </tr> <tr> <td>3552</td> <td>FAStT 5000</td> </tr> <tr> <td>1722</td> <td>FAStT 600</td> </tr> <tr> <td>1742</td> <td>FAStT 700 and 900</td> </tr> </table> Note: The system displays "-" if the product cannot be determined.	2105F20	ESS F20	2105800	ESS 800	2145	SAN Volume Controller and SAN Integration Server	3552	FAStT 5000	1722	FAStT 600	1742	FAStT 700 and 900
2105F20	ESS F20												
2105800	ESS 800												
2145	SAN Volume Controller and SAN Integration Server												
3552	FAStT 5000												
1722	FAStT 600												
1742	FAStT 700 and 900												
Storage Device WWNN	The world-wide node name of the hosting storage device. This information is helpful if you want to ensure that all LUNs in a storage pool are from the same storage device.												

Related topics:

- "Managing volumes and LUNs" on page 128
- "Storage pools" on page 37
- "Storage devices" on page 46
- "Viewing available LUN details" on page 134
- "Viewing LUN details" on page 134
- "Volumes" on page 49
- "Using the SAN File System console" on page 333

Policies

Use this panel to perform actions related to SAN File System policies, or to view existing policy properties.

Actions:

Create	Takes you to a wizard for creating a new policy.
Properties	Displays additional details about the selected policy and allows you to edit policy rules. Note: Only inactive policies can be changed. Rules for active policies will be listed as read-only.
Delete	Deletes the selected, inactive policy or policies from the system. Note: No policy can be deleted when it is in an active state. You must first activate an inactive policy. This action will automatically make the formerly active policy inactive, allowing you to delete it at that point.
Activate	Activates the selected, inactive policy while deactivating the currently active one.

Fields:

You can select a policy for which to view additional details or perform actions. The policy table lists the name and description of the policy and also provides the following:

State	The state of the policy can be active or inactive . Note: Only one policy can be active at any given time.
Last Active	Date and time that the policy was deactivated. Notes: <ol style="list-style-type: none">1. All date and time attributes are based in coordinated universal time (UTC).2. The “Last Active” state will be shown as “-” if the associated policy was never active or is currently active.
Modified	Date and time that the policy was either last modified or initially created. Note: All date and time attributes are based in coordinated universal time (UTC).

Related topics:

- “Activating a policy” on page 118
- “File placement” on page 43
- “Policies and rules” on page 44
- “Using the SAN File System console” on page 333

Policy Properties - Details

Use this panel to view the name and description of the policy, as well as the following policy properties:

Attributes:

State	Can be active or inactive . Note: Only one policy can be active at any given time.
Last Active	Date and time that the policy was deactivated. Notes: <ol style="list-style-type: none">1. All date and time attributes are based in coordinated universal time (UTC).2. The “Last Active” state will be shown as “-” if the associated policy was never active or is currently active.
Modified	Date and time that the policy was either last modified or initially created. Note: All date and time attributes are based in coordinated universal time (UTC).

Related topics:

- “File placement” on page 43
- “Policies and rules” on page 44

- “Using the SAN File System console” on page 333

Policy Properties - Rules

Use this panel to view policy rules and edit rules for inactive policies.

Notes:

1. Rules cannot be changed while a policy is in the active state.
2. All date and time attributes in these rules are based in coordinated universal time (UTC). Therefore, any time and date entry is assumed to be in universal time. Also, functions (for example, “dayofweek”) are based on UTC.

Attributes:

Rules

You can only edit rules for inactive policies.

Note: Although the rules for the active policy will be read-only, you can copy those rules to the clipboard to paste into another policy.

Related topics:

- “File placement” on page 43
- “Policies and rules” on page 44
- “Using the SAN File System console” on page 333

Processes

Use this panel to monitor SAN File System processes, also known as “threads.”

Actions:

General Properties

Displays information related to all the process limits at both cluster and Metadata server levels.

Fields:

The table provides the following information about each process:

ID	System-assigned identification number for the associated process.
Operation	Name of the process.
Started	The time the process command was started.
Running Time	The total elapsed time that the process has been running.

Related topics:

- “Cluster” on page 16
- “Metadata server” on page 33
- Chapter 3, “Monitoring”, on page 57
- “Using the SAN File System console” on page 333

Processes Properties - Cluster-Level Details

Use this panel to view process limits for the entire cluster.

Attributes:

The **Cluster-Level Process Limits** table displays the following information:

Server Workload	A per-Metadataserver value that applies only to workload operations.
Admin	A per-Metadataserver value that applies only to how many admin commands can be running simultaneously from both the SAN File System console and the command-line interface. The Admin Process Limit is primarily relevant to just the master Metadataserver, and is totally independent of the Server Workload Process Limit. Note: Very few commands, such as ones relating to changing the state of the cluster, can run on a temporary process even if the limit is currently maxed out.

Related topics:

- “Cluster” on page 16
- “Metadataserver” on page 33
- Chapter 3, “Monitoring”, on page 57
- “Using the SAN File System console” on page 333

Processes Properties - Server-Level Details

Use this panel to view process limits for each Metadataserver.

Attributes:

The **Server-Level Process Limits** table displays the following information:

Server Value	The name of the associated Metadataserver. (Range is 1 to 4.) The maximum number of processes (or “threads”) to be used for “garbage collecting” of deleted files. Note: This attribute cannot be reset after the initial installation and is not configurable from either the SAN File System console or the command-line interface.
--------------	---

Related topics:

- “Cluster” on page 16
- “Metadataserver” on page 33
- Chapter 3, “Monitoring”, on page 57
- “Using the SAN File System console” on page 333

Remove Volumes

Use this panel to unassign volumes from a particular storage pool. Unassigning the volumes redefines them as available LUNs. You can also select a method of removal (regular or forced).

Fields:

The **Volumes Selected for Removal** table lists the name and description for each volume, as well as the following information:

State	One of three volume states are possible: Active Only active volumes can have data saved to them. Suspended Volumes cannot have data stored on them when in the suspended state. Being Removed Volumes cannot have data stored on them when they are "Being Removed."
Storage Pool	Name of the storage pool to which the volume belongs. Note: If you entered this panel from the Storage Pool Properties – Current Volumes panel, the Storage Pool field does not display, but is listed in the panel header instead.
Size (MB)	Total space available in the associated volume, in megabytes.
Used (MB)	Amount of space currently used by the volume, in megabytes.
Used (%)	The percentage of the total volume size that is currently being used.

Related topics:

- "Managing storage pools" on page 122
- "Storage pools" on page 37
- "Volumes" on page 49
- "Using the SAN File System console" on page 333

Restart Service Statistics- Probe Overview

Use this panel to view information about Metadata server restart service probes. These probes monitor Metadata servers and automatically restart them as needed.

Attributes:

Probe State	The probed Metadata server status. The possible values are: “Not Probed” The probe has not started because the Metadata server restart service is either disabled, in standby or in an aborted state. “Probing” The Metadata server restart service has started the probe. “Live Server” The Metadata server restart service found the Metadata server and it is live. There is no need for the Metadata server restart service to restart the Metadata server. “Absent Server” The Metadata server restart service positively detected that the Metadata server is absent and will attempt to restart the Metadata server. “Unknown” The tests to determine if the Metadata server is live or absent have failed. The Metadata server restart service will not attempt to start the Metadata server.
Server Restart Service Started	Date and time that the Metadata server restart service was enabled for the associated Metadata server.
Last Probe	Data and time of the last probe.
Current Probe Retries	The number of retries in succession that are currently being tried. Note: This value will normally be 0.
Probes	Total number of probes performed so far.
Probe Retries	Total number of probe retries.
Highest Retries	The highest number of probe retries so far.
Lowest Retries	The lowest number of probe retries so far.

Related topics:

- “Metadata server” on page 33
- “Managing Metadata servers” on page 109
- “Engines” on page 23
- “Using the SAN File System console” on page 333

Restart Service Statistics- Test Details

Use this panel to view information about Metadata server restart service “liveness” and “absence” tests.

Attributes:

The **Liveness Test** table lists the following information:

Liveness Test Timeouts	The total number of times the liveness test has taken longer than the test timeout interval.
Last Liveness Test (msec)	The time taken by the last liveness test.
Highest Liveness Test (msec)	The highest value for the time taken by the liveness test.
Lowest Liveness Test (msec)	The lowest value for the time taken by the liveness test.

Attributes:

The **Absence Test** table lists the following information:

Absence Tests	Counter of the number of times the absence test was started.
Last Absence Test (msec)	Time taken by the last absence test.
Highest Absence Test (msec)	Highest value for the time taken by the last absence test.
Lowest Absence Test (msec)	Lowest value for the time taken by the last absence test.

Related topics:

- “Metadata server” on page 33
- “Managing Metadata servers” on page 109
- “Engines” on page 23
- “Using the SAN File System console” on page 333

Restart Service Statistics- Tuning Details

Use this panel to view information about Metadata server restart service probe retry limits, probe intervals, and liveness test timeouts. These service probes monitor Metadata servers and automatically restart them as needed.

Attributes:

Retry Limit	Maximum number of times to try to detect the liveness of a Metadata server if the Metadata server is not declared “dead.” When this limit is reached, the Metadata server restart service will be turned off.
Probe Interval (sec)	The Metadata server restart service will start a probe at this interval.
Liveness Test Timeouts (sec)	Maximum time to wait for a Metadata server to respond before declaring it “dead.”

Related topics:

- “Metadata server” on page 33
- “Managing Metadata servers” on page 109
- “Engines” on page 23
- “Using the SAN File System console” on page 333

Revert Fileset to Image

Use this panel to review and verify the image for the fileset (container) to be reverted to, as well as select a reversion method (regular or forced).

Note: You cannot revert a fileset to an image if the fileset has child filesets (that is, subfilesets). You must detach child filesets first.

Attributes:

The **Image and Fileset (Container) to be Reverted** table displays the name and description of the selected FlashCopy image so that you can verify your choice before reverting the associated fileset.

Attention: More recent images will be deleted after reverting to an older one.

The table also lists the following detail for the selected image:

Fileset (Container)	Fileset that will be reverted to the associated image.
Fileset (Container) State	Two fileset states are possible: <ul style="list-style-type: none">• Attached• Detached
Directory Name	Directory name where the image is located. Note: The FlashCopy image directory is the one below the “.flashcopy” directory of the fileset.
Directory Path	Path where the FlashCopy image directory is located, using the following naming convention: <ul style="list-style-type: none">• the fileset’s attach point• “/.flashcopy”• /Directory name
Date and Time	Indicates when the FlashCopy image was taken.

Related topics:

- “Attaching a fileset” on page 103
- “Filesets” on page 24
- “Creating a FlashCopy image” on page 107
- “Detaching a fileset” on page 105
- “FlashCopy images” on page 26
- “FlashCopy Images” on page 395
- “Reverting to a previous FlashCopy image” on page 108
- “Using the SAN File System console” on page 333

Roles

Use this panel to view the types and corresponding number of roles configured for your SAN File System. Roles refer to the level of access that a user has to SAN File System functionality.

Fields:

Name	The title of the role within the SAN File System. The role name possibilities are: <ul style="list-style-type: none">• Administrator• Operator• Monitor• Backup <p>Note: User roles are shown in ascending order of access and control.</p>
Defined Users	The number of users who have been granted that role, and its associated access level, within SAN File System. Users are defined by their highest assigned level of access to the system.

Related topics:

- “Administrative security” on page 39
- “Managing users” on page 126
- “User roles” on page 46
- “Using the SAN File System console” on page 333

Sign on

Use this panel to log into the SAN File System console.

Actions:

The rules associated with signing onto the SAN File System console are determined by the Lightweight Directory Access Protocol (LDAP) server in your environment, which manages SAN File System authentication. Please see your LDAP administrator for details regarding your SAN File System user name and password.

In addition to LDAP standards, the following login restrictions apply:

User name	Must be alpha-numeric and no more than 256 characters.
Password	Any valid ASCII is accepted up to 256 characters.

Related topics:

- “Using the Help Assistant” on page 433
- “Using the SAN File System console” on page 333

Security Log

This panel displays the SAN File System admin authentication log, which shows the admin user login activity.

Actions:

You have the option to set certain parameters regarding which messages the security log retrieves.

Messages to Retrieve

You have three choices regarding which type of messages to retrieve:

Most Recent (default)

Always gets the most recent rows in the log.

Note: This choice is not relative to the current log view, unlike the other two choices.

Previous Messages

Selects rows down the log relative to its current view.

Next Messages

Selects rows up the log relative to its current view.

Number of Messages

Determines how many messages display at one time. You can choose from 10, 25, 50, or 100.

Refresh

Reloads the log based on the adjacent log-retrieval settings.

Fields:

The **Security Log** table lists the following details:

Message ID

Follows this convention:

“XXX” Component

“YY” Subcomponent

“nnnn”

Message code

“Z” Severity (Severe, Error, Warning, or Information level)

Level

Based on the severity level of the Message ID. Four levels are possible:

•  Severe

•  Error

•  Warning

•  Information

Server

The Metadata server name that is the source of the message.

Date and Time	System (not local) time that the message was issued.
Message	Text of the message.

Related topics:

- “Common Information Model” on page 10
- “Metadata server” on page 33
- Chapter 3, “Monitoring”, on page 57
- “Using the SAN File System console” on page 333

Server Properties - Details

Use this panel to view additional attributes for the selected Metadata server.

Attributes:

You can view the Metadata server name, current time, as well as the following Metadata server details:

State	The Metadata server can be in one of the following states: <ul style="list-style-type: none"> • not running • failed initialization • initializing • not added • joining • offline • fully quiescent • partly quiescent • online • unknown
Last State Change	Time when the state last changed.
Target State	State the Metadata server is heading toward.
Last Target State Change	Time when the target state last changed.
Server Role	Gives the status of the master-subordinate relationship among the servers. Note: Only one Metadata server can be the master. The master role is specified during the initial installation.
Filesets (Containers)	The number of filesets being served by the associated Metadata server.
Last Boot	Date and time of the last boot for the associated Metadata server.
Current Time	Current system (not local) time.
Software Version	Current software version on the cluster. It may or may not have yet been committed.

Related topics:

- “Cluster” on page 356
- “Filesets” on page 24
- “Managing Metadata servers” on page 109

- “Metadata server” on page 33
- “Metadata server states” on page 35
- “Using the SAN File System console” on page 333

Server Properties - Networking Details

Use this panel to view networking properties for the selected Metadata server.

Attributes:

You can see the server-to-server and client-server transport protocols, as well as the IP address. You can also view the following assigned ports:

Client-Server Port	This port is used for communication between a Metadata server and a SAN File System client using the SAN File System protocol. Note: The Client-Server port needs to be the same for the client and the server.
Cluster Management Port	This port is used for internal communication among the Metadata server within a cluster.
Heartbeat Port	This port is used for internal health monitoring of Metadata server within a cluster.
Administration Port	This port is used for receiving administrative requests from an administrative client (applies to both the SAN File System console and the command-line interface).

Related topics:

- “Managing Metadata servers” on page 109
- “Metadata server” on page 33
- “Metadata server states” on page 35
- “Using the SAN File System console” on page 333

Server Restart Service

Use this panel to start or stop the Metadata server restart service, or to view information about individual servers. This service enables probes to periodically check the Metadata server condition. They will automatically restart the Metadata servers as appropriate, as they continually monitor Metadata server state.

Note: If a Metadata server has the restart service enabled, and the Metadata server is later manually shut down, the restart service is disabled and you will need to start it again.

Actions:

Start Service	Enables the Metadata server restart service.
Stop Service	Disables the Metadata server restart service.
Statistics	Displays additional properties for the selected Metadata server, including tuning details, test details and a probe overview.

Fields:

Server

Lists all of the Metadata servers in the SAN File System cluster. Each Metadata server name links to statistical details for that particular Metadata server in the **Probe Overview** panel.

Service State

State of the automatic Metadata server restart service. The possible values are:

“Off” The restart service is off.

“On”

“Running”

The restart service is up and running.

“Standby”

The Metadata server has been manually shutdown and the restart service will automatically turn itself on when the Metadata server is restarted.

Note: If a Metadata server has the restart service enabled, and the Metadata server is later manually shut down, the restart service is disabled and you will need to start it again.

“Failed”

The restart service has detected a non-live Metadata server too many times in succession.

“Unknown”

The state of the restart service is indeterminate because the Metadata server could not be reached.

Probe State	<p>The probed Metadata server status. The possible values are:</p> <p>“Not Probed” The probe has not started because the Metadata server restart service is either disabled, in standby or in an aborted state.</p> <p>“Probing” The Metadata server restart service has started the probe.</p> <p>“Live Server” The Metadata server restart service found the Metadata server and it is live. There is no need for the Metadata server restart service to restart the Metadata server.</p> <p>“Absent Server” The Metadata server restart service positively detected that the Metadata server is absent and will attempt to restart the Metadata server.</p> <p>“Unknown” The liveness and absence tests failed. The Metadata server restart service will not attempt to start the Metadata server.</p>
Last Probe	Date and time of the last probe.
Probes	Total number of probes performed so far.
Highest Retries	The highest number of probe retries so far.

Related topics:

- “Engines” on page 23
- “Metadata server” on page 33
- “Managing Metadata servers” on page 109
- “Starting the Metadata server restart service” on page 113
- “Stopping the Metadata server restart service” on page 115
- “Using the SAN File System console” on page 333

Servers

Use this panel to perform an action on SAN File System servers, or to view Metadata server properties.

Actions:

Properties	Displays additional detailed properties of the selected Metadata server, including networking properties.
------------	---

Start

Will attempt to put the Metadata server or servers selected in an available clustered state, if they are not already in such a state.

Attention: You should not use the start action to recycle a Metadata server. The selected servers should only be started if they are in one of the following states:

- not running
- unknown

Notes:

1. If the master Metadata server is either “not running” or “unknown,” then starting subordinate servers, without also starting the master, is not allowed.
2. The system will try to match the servers to the cluster’s current target state (online, offline, fully or partly quiescent).

Stop

Allows you to gracefully stop the selected Metadata server or servers. The servers and their associated states, roles, and filesets are displayed for review and confirmation before the stop is executed.

Note: Only active servers can be stopped. Active servers are those in one of the following states:

- offline
- fully quiescent
- partly quiescent
- online

Fields:

You can select a Metadata server on which to view additional details or perform actions. The **Servers** table lists the Metadata server’s state and the date and time it was last booted. The table also provides the following:

Name	Links to additional details about the selected Metadata server.
Server Role	Gives the status of the master-subordinate relationship among the servers. Note: Only one Metadata server can be the master. The master role is specified during the initial installation.
Filesets (Containers)	Number of filesets being served by the associated Metadata server.

Related topics:

- “Changing the master Metadata server” on page 110
- “Cluster” on page 16
- “Filesets” on page 24
- “Managing Metadata servers” on page 109
- “Metadata server” on page 33

- “Metadata server states” on page 35
- “Using the SAN File System console” on page 333

SNMP Properties - SNMP Events

Use this panel to specify the severity levels for which you want SNMP traps to be sent to the appropriate SNMP manager.

Attributes:

Severe events	Select this checkbox if you want SNMP traps sent for severe events.
Error events	Select this checkbox if you want SNMP traps sent for error events.
Warning events	Select this checkbox if you want SNMP traps sent for warning events.
Information events	Select this checkbox if you want SNMP traps sent for information events.

Related topics:

- “Alerts” on page 4
- “Cluster” on page 16
- “Filesets” on page 24
- “Logs” on page 30
- “Storage Pools” on page 423
- “Metadata server” on page 33
- “Using the SAN File System console” on page 333

SNMP Properties - SNMP Managers

Use this panel to specify up to two destination SNMP managers. SNMP trap messages are sent to SNMP managers.

Attributes:

Complete the fields to specify up to two destination SNMP managers. To enable sending SNMP trap messages to an SNMP manager select the associated checkbox and enter its Destination IP and Destination Port. To disable sending SNMP trap messages, clear the associated checkbox.

Note: One of the SNMP managers should be configured to be the Master console to use the Service Alert or remote access features.

SNMP Manager check boxes	To enable sending SNMP trap messages to an SNMP manager, select a checkbox and complete the corresponding details. Note: When this checkbox is selected, the Destination IP and Port become required.
Destination IP	(Required if the corresponding SNMP Manager check box is selected.) Enter an IP address to enable SNMP trap messages to be sent to the associated SNMP manager. You must enter the address in dotted decimal notation.

Destination Port	(Required if the corresponding SNMP Manager check box is selected.) Enter an associated port for the intended SNMP manager. Note: The destination port can range between 1 and 65 535.
SNMP Version	Select one of the following options: <ul style="list-style-type: none"> • V1 (default) • V2C Note: The “C” stands for “Community.”
SNMP Community	(Optional) Used for authenticating V1 and V2C. (default: public)

Related topics:

- “Alerts” on page 4
- “Service Alert” on page 9
- “Master console” on page 32
- “Remote access” on page 39
- “Using the SAN File System console” on page 333

Statistics - Client Sessions

Use this panel to view lock and lease information about SAN File System client interactions.

Fields:

Attribute	Details information for expired and current clients and total client sessions, as well as session, data and byte-range locks.
Total	The total for all client sessions.
per Client Session	The average per total client sessions. Displays “-” if not applicable.

Attributes:

Expired Clients	The clients that have expired client-server leases. Attention: An expired lease might indicate the need for action on the client side by the application administrator. When client leases expire, they could lose access to a set of data if another client creates a lock to it.
Current Clients	The clients that have valid client-server leases.
Total Client Sessions	Total sessions (current or expired) for all clients.
Session Locks	The current number of session locks for all clients. Clients must acquire session locks to do any operation with a filename.

Data Locks	The current number of data locks held for all clients. Clients must hold data locks in order to cache data pages, attributes of files, and to cache read-only attributes and contents of directories and links.
Byte Range Locks	The current number of byte range locks held for all clients. These locks are used to implement lock system calls and have no direct effect on the contents or attributes of SAN File System objects or other file system opportunities.

Related topics:

- “Clients” on page 11
- “Locks and leases” on page 30
- “Metadata server” on page 33
- “Using the SAN File System console” on page 333

Statistics - Cluster

Use this panel to view high-level information about the workload on the SAN File System cluster master Metadata server.

Attributes:

The **Cluster Master Workload on Master Server** table lists the following information:

Update Transactions	The total number of metadata transactions relating to metadata updates for system objects.
Total Transactions	The total number of metadata transactions relating to metadata activity.
Dirty Buffers	The current number of dirty buffers for system metadata activity. Dirty buffers contain data awaiting input/output (I/O) to disk.
Clean Buffers	The current number of clean buffers for system metadata activity. Clean buffers contain data, but area available for reuse.
Free Buffers	The current number of free buffers for system metadata activity. Free buffers are those that are currently not in use.
Total Buffers	The current number of total buffers for system metadata activity. Total buffers are composed of clean buffers, dirty buffers, and free buffers.

Related topics:

- “Cluster” on page 16
- “Metadata server” on page 33
- “Using the SAN File System console” on page 333

Statistics - Create Report

Use this panel to extract high-level overview information about SAN File System components, including statistics on the cluster, servers, Metadata server transactions, buffers, and locks, client sessions, filesets (containers), storage pools, volumes, LUNs, and engines.

Fields:

Cluster - Master Workload	Returns high-level information about the workload on the SAN File System cluster master Metadata server.
Servers	Returns high-level information about the active or inactive status of servers that are part of the SAN File System cluster.
Server Transactions - Client Workload	Returns high-level information about client-level workload, read, and read/write transactions being performed on all the servers, including the master Metadata server.
Server Buffers - Client Workload	Returns high-level information about client-level work regarding dirty, clean, and free buffers being done on all the servers, including the master Metadata server.
Server Locks - Client Workload	Returns high-level information about client-level work regarding session, data and byte-range locks being done on all the servers, including the master Metadata server.
Client Sessions	Returns high-level information about lock and lease interactions between SAN File System and its clients.
Filesets (Containers)	Returns high-level information about SAN File System filesets, including attachment status, soft quota and usage thresholds, FlashCopy image details, and metadata transactions.
Storage Pools	Returns high-level information about SAN File System storage pools, including size and percentage used.
Volumes	Returns high-level information about SAN File System volumes, including activity status and percentage used.
LUNs	Returns high-level information about SAN File System LUNs, including availability and size.
Engines	Returns high-level information about SAN File System engines, including warnings, number of engines, as well as memory and CPU usage.

Fields:

Pushbuttons.

Create Report	Produces a statistics report based on your SAN File System component selections for this panel.
----------------------	---

Related topics:

- “Cluster” on page 16
- “Filesets” on page 24
- “Locks and leases” on page 30
- “Engines” on page 23
- “Storage pools” on page 37
- “Metadata server” on page 33
- “Clients” on page 11
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Statistics - Engines

Use this panel to view high-level information about SAN File System engines, including warnings, number of engines, as well as memory and CPU usage.

Fields:

Attribute	Detailed information for engines including those with warnings or errors; normal engines; total engines; the memory used and memory available; and the highest CPU used.
Total	The total for all engines.
per Engine	The average per active engine. Displays “-” if not applicable.

Attributes:

Engines with Warnings or Errors An engine is considered to have a warning or an error if it has one or more warning or fan states from the following components:

Temperature

One or more “Warning” or higher-level thresholds are surpassed.

Voltage

One or more “Warning” or higher-level thresholds are surpassed.

Fans

One or more fans is below the threshold (15%) fan speed.

Normal Engines Engines that are functioning in a “normal capacity,” meaning those engines for which none of the thresholds for temperatures, voltages, or fans are surpassed.

Total Engines

Attention: The normal setting will be the default if no temperature or voltage thresholds have been set.

All engines, whether they have warnings or errors, or are functioning normally.

Related topics:

- “Engines” on page 23

- “Using the SAN File System console” on page 333

Statistics - Filesets

Use this panel to view high-level information about filesets (containers), including attachment status, soft quota thresholds, and FlashCopy image details.

Fields:

Attribute	Detailed information for detached, attached and total filesets; soft quota thresholds; and FlashCopy images.
Total	The total for all filesets.
per Fileset (Container)	The average per fileset. Displays “-” if not applicable.

Attributes:

Detached Filesets (Containers)	The number of filesets that are detached from the global namespace.
Attached Filesets (Containers)	The number of filesets that are attached to the global namespace.
Total Filesets (Containers)	Total for all filesets.
Filesets (Containers) above Soft Quota Threshold	For soft quotas that are enabled, a quota is over threshold when usage is greater than or equal to the quota setting.
FlashCopy Images	Total number of images of all filesets.
Last FlashCopy Image	Date and time that the last image was taken.

Related topics:

- “Filesets” on page 24
- “FlashCopy images” on page 26
- “Storage management” on page 43
- “Using the SAN File System console” on page 333

Statistics - LUNs

Use this panel to view high-level information about SAN File System LUNs, including availability and size.

Fields:

Attribute	Detailed information for LUNs including those that are unavailable, available or assigned; total number of LUNs; and available LUN size.
Total	The total for all LUNs.
per LUN	The average per LUN. Displays “-” if not applicable.

Attributes:

Unavailable LUNs	LUNs that are unassigned to a volume, and are “read only.” Note: Unavailable and Available LUNs are collectively known as Unassigned LUNs.
Available LUNs	LUNs that are not currently assigned to a storage pool, but are writable (write or read/write). Note: Unavailable and Available LUNs are collectively known as Unassigned LUNs.
Assigned LUNs	LUNs that are currently assigned to a storage pool. Note: An assigned LUN is also known as a volume.
Total LUNs	Total available, unavailable, and assigned LUNs.
Available-LUN size (MB)	Total size of available LUNs only.

Related topics:

- “Storage pools” on page 37
- “Storage management” on page 43
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Statistics - Report

Use this panel to view online, or print high-level overview information about SAN File System components, including statistics on the cluster, servers, server transactions, buffers, and locks, client sessions, filesets (containers), storage pools, volumes, LUNs, and engines.

For an explanation of the statistics that you selected for the report, refer to the corresponding statistics panel related to the items you selected.

Fields:

Pushbuttons.

Close

Closes the report and returns to the **Create Report** panel.

Related topics:

- “Cluster” on page 16
- “Filesets” on page 24
- “Locks and leases” on page 30
- “Engines” on page 23
- “Storage pools” on page 37
- “Metadata server” on page 33
- “Statistics - Client Sessions” on page 413
- “Statistics - Cluster” on page 414
- “Statistics - Filesets” on page 417
- “Statistics - Engines” on page 416

- “Statistics - LUNs” on page 417
- “Statistics - Storage Pools”
- “Statistics - Servers”
- “Clients” on page 11
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Statistics - Storage Pools

Use this panel to view high-level information about SAN File System storage pools, including size and percentage used.

Fields:

Attribute	Detailed information for storage pools including those above and below usage threshold; total storage pools; total volumes on the storage pools; size of the storage pools in megabytes; the percentage used; and the highest percentage used.
Total	The total for all storage pools.
per Storage Pool	The average per storage pool. Displays “-” if not applicable.

Attributes:

Storage Pools above Usage Threshold	The number of storage pools that have reached their percentage-used alert limit.
Storage Pools below Usage Threshold	The number of storage pools that have not reached their percentage-used alert limit.
Total Storage Pools	Total values for all storage pools.
Volumes	The number of volumes on all the storage pools.
Size (MB)	The total amount of space (used and unused) in the storage pools.
Used (%)	The percentage of storage that is used in the storage pools.
Highest “Used (%)”	The highest value of percentage used of all the storage pools (that is, the percentage used for the storage pool that is closest to reaching capacity).

Related topics:

- “Storage pools” on page 37
- “Storage management” on page 43
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Statistics - Servers

Use this panel to view high-level information about SAN File System Metadata servers, including client workload details.

Fields:

The **Servers** table lists the following information:

Attribute	Details information for inactive, active and total Metadata servers.
Total	The total for all Metadata servers.
per Server	The average per total Metadata servers. Displays “-” if not applicable.

Attributes:

Inactive Servers	Metadata servers in one of the following states: <ul style="list-style-type: none">• Not running• Failed initialization• Initializing• Not added• Joining
Active Servers	Metadata servers in one of the following states: <ul style="list-style-type: none">• Offline• Partly quiescent• Fully quiescent• Online
Total Servers	All the Metadata servers that have been added to the cluster, including those that are in an unclustered meta-state.

Fields:

The **Server Transactions — Client Workload** table pertains to client-level transaction work being done on all the Metadata servers, including the master Metadata server. The table lists each Metadata server’s name and a total for all Metadata servers regarding the following details:

Update	Update (“write”) transactions.
Total	All Metadata server transactions (“read” and “write”).

Fields:

The **Server Buffers — Client Workload** table pertains to client-level work, regarding buffers, being done on all the Metadata servers, including the master Metadata server. The table lists each Metadata server’s name and a total for all Metadata servers regarding the following details:

Dirty	The current number of dirty buffers for client/user metadata activity. Dirty buffers contain data awaiting input/output (I/O) to disk.
Clean	The current number of clean buffers for client/user metadata activity. Clean buffers contain data, but are available for reuse.

Free	The current number of free buffers for client/user metadata activity. Free buffers are those that are currently not in use.
Total	All buffer types per Metadata server.

Fields:

The **Server Locks — Client Workload** table pertains to client-level work regarding locks, being done on all the Metadata servers, including the master Metadata server. The table lists each Metadata server’s name and a total workload for all Metadata servers regarding the following details:

Session	The current number of session locks for each Metadata server. A session lock holds a reference on an engine. A client must acquire a session lock to do any operation with a filename.
Data	The current number of data locks held for each Metadata server. Clients must hold data locks in order to cache data pages, attributes of files, and to cache read-only attributes and contents of directories and links.
Byte Range	The current number of byte range locks held for each Metadata server. These locks are used to implement lock system calls and have no direct effect on the contents or attributes of SAN File System objects or other file system operations.
Total	All lock types per Metadata server.

Related topics:

- “Clients” on page 11
- “Cluster” on page 16
- “Locks and leases” on page 30
- “Metadata server” on page 33
- “Using the SAN File System console” on page 333

Statistics - Volumes

Use this panel to view high-level information about SAN File System volumes, including activity status and percentage used.

Fields:

Attribute	Details information for volumes including inactive, active, and total volumes; size of the volumes in megabytes; the percentage used; and the highest percentage used.
Total	The total value for all volumes.
per Volume	The average value per volume. Displays “-” if not applicable.

Attributes:

Inactive Volumes	Includes volumes that are: <ul style="list-style-type: none">• Suspended• Being removed
Active Volumes	Those volumes that are being used (that is, not “Suspended” or “Being removed”).
Total Volumes	Total value for all volumes.
Size (MB)	The total amount of volume space.
Used (%)	The percentage of storage that is used in the volumes.
Highest “Used (%)”	The highest value of percentage used of all the volumes (that is, the percentage used for the volume that is closest to reaching capacity).

Related topics:

- “Storage pools” on page 37
- “Storage management” on page 43
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Stop Cluster

Use this panel to view and stop all servers in the SAN File System cluster.

Attributes:

The name is given for each Metadata server to be stopped, as well as the following information:

State	State that the associated Metadata server is in, prior to the stop being executed.
Server Role	Gives the status of the master-subordinate relationship among the servers. Note: Only one Metadata server can be the master. The master role is specified during the initial installation.
Filesets (Containers)	Number of filesets being served by the associated Metadata server.

Related topics:

- “Cluster” on page 16
- “Cluster states” on page 19
- “Filesets” on page 24
- “Managing the cluster” on page 82
- “Metadata server” on page 33
- “Metadata server states” on page 35
- “Using the SAN File System console” on page 333

Stop Servers

Use this panel to view and stop the selected servers in the SAN File System cluster.

Attention: If you select to stop the master Metadata server this way, it will disruptively stop all cluster activity and all client sessions.

Attributes:

The name is given for each Metadata server to be stopped, as well as the following information:

State	State that the associated Metadata server is in, prior to the stop being executed. Note: Only active servers can be stopped. Active servers are those in one of the following states: <ul style="list-style-type: none">• offline• fully quiescent• partly quiescent• online
Server Role	Gives the status of the master-subordinate relationship among the servers. Note: Only one Metadata server can be the master. The master role is specified during the initial installation.
Filesets (Containers)	Number of filesets being served by the associated Metadata server.

Related topics:

- “Cluster” on page 16
- “Cluster states” on page 19
- “Filesets” on page 24
- “Managing Metadata servers” on page 109
- “Metadata server” on page 33
- “Metadata server states” on page 35
- “Using the SAN File System console” on page 333

Storage Pools

Use this panel to create and delete storage pools; as well as select a storage pool to serve as the default. You can also view storage pool details.

Actions:

Create	Takes you to a wizard for creating a new storage pool.
Properties	Displays, and allows you to set, certain properties for the selected storage pool or storage pools, such as current volumes and available LUNs.

Delete	Deletes the selected storage pool from SAN File System. Note: You can only delete user storage pools. If you attempt to delete the user default or system storage pool, you will get an error message before the final delete action.
Set as Default	Sets the selected storage pool as the default storage pool. Note: You cannot make the system storage pool the default storage pool.

Fields:

You can select a storage pool on which to view additional details or perform actions. In addition to its description, the table provides the following information about each storage pool:

Name	Links to additional details about the associated storage pool.
Type	One of three storage pool types are possible: User storage pools Data storage pools in which user data gets stored in blocks. User Default storage pool The user storage pool that data gets assigned to, if not picked up by a policy rule. System storage pool The storage pool that stores the metadata (regarding filesets, policies, and so forth). Note: It is important to ensure that the system storage pool never runs out of space. You can monitor the system storage pool usage by setting and enabling a usage threshold alert against it on the Storage Pool Properties – General Settings panel.
Size (MB)	Total space available in the associated storage pool.
Used (MB)	Amount of space currently used by the storage pool.
Used (%)	Percentage of the total storage pool size that is currently being used.
Threshold (%)	Storage pool usage percentage threshold. When the Used (%) reaches this threshold, SAN File System will send an alert.
Volumes	Number of volumes in the associated storage pool.

Related topics:

- “Managing storage pools” on page 122
- “Storage pools” on page 37
- “Viewing storage pool details” on page 125
- “Viewing storage pool statistics” on page 126

- “Managing volumes and LUNs” on page 128
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Storage Pool Properties - Available LUNs

Use this panel to select one or more available LUNs to be added as volumes to the associated storage pool. Available LUNs are both unassigned and writable.

Actions:

Add as Volumes

Initiates the **Add Volumes wizard** displaying the Available LUNs that you select from this panel. You can then verify and modify additional settings in the wizard.

Fields:

The Available LUNs table allows you to select one or more LUNs to add as volumes to the associated SAN File System storage pool. In addition, the table displays the following information:

LUN ID	Links to additional details about the associated LUN.
Size (MB)	Total size of the LUN in megabytes.
Vendor	The vendor of the storage device associated with the LUN. Note: The system displays “-” if the vendor cannot be determined.
Product	The product name of the storage device associated with the LUN. Note: The system displays “-” if the product cannot be determined.
Version	The version of the storage device, for example: “4.1.” Note: The system displays “-” if the version cannot be determined.
Engine WWN	Uniquely identifies the storage device by its World Wide Name (WWN). This information is helpful if you want to ensure that all LUNs in a storage pool are from the same storage device.

Related topics:

- “Managing storage pools” on page 122
- “Managing volumes and LUNs” on page 128
- “Storage devices” on page 46
- “Storage pools” on page 37
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Storage Pool Properties - Current Volumes

Use this panel to view and remove volumes associated with the selected storage pool.

Actions:

Remove Volumes

Will remove from the storage pool any volumes that you select in the table.

Fields:

The **Remove Volumes** table allows you to select one or more volumes to remove from the associated storage pool. The table displays a description for each volume in addition to the following information:

Name	Links to additional details about the associated volume.
State	Three volume states are possible: Active Only active volumes can have data saved to them. Suspended Volumes cannot have data stored on them when in the suspended state. Being Removed Volumes cannot have data stored on them when they are "Being Removed."
Size (MB)	Total volume size in megabytes.
Used (MB)	Number of megabytes currently used by the volume.
Used (%)	Percentage of total volume size currently used by the volume.

Related topics:

- "Managing volumes and LUNs" on page 128
- "Managing storage pools" on page 122
- "Storage pools" on page 37
- "Volumes" on page 49
- "Using the SAN File System console" on page 333

Storage Pool Properties - Details

Use this panel to view additional attributes for a selected storage pool.

Attributes:

You can view the name, partition size, and description for the storage pool. In addition, the **Details** panel provides the following information:

Type	<p>Three storage pool types are possible:</p> <p>User storage pools Data storage pools in which user data gets stored in blocks.</p> <p>User Default storage pool The user storage pool that data gets assigned to, if not picked up by a policy rule.</p> <p>System storage pool The storage pool that stores the metadata (regarding filesets, policies, and so forth). Note: It is important to ensure that the system storage pool never runs out of space.</p>
Size (MB)	The total space available in the associated storage pool.
Used (MB)	The amount of space being used in the storage pool.
Used (%)	The percentage of the total storage pool size that is currently being used.
Threshold (%)	The storage pool usage percentage threshold. When the Used (%) reaches this threshold, SAN File System will send an alert.
Allocation Size (KB)	<p>The size allocated for the storage pool may be one of three possible (kilobyte) values:</p> <ul style="list-style-type: none"> • System Automated • 4 - Fixed • 128 - Fixed
Volumes	Number of volumes in the associated storage pool.

Related topics:

- “Managing storage pools” on page 122
- “Storage pools” on page 37
- “Volumes” on page 49
- “Using the SAN File System console” on page 333

Storage Pool Properties - General Settings

Use this panel to change the selected storage pool’s name, description, and percentage used threshold alert.

Attributes:

Name	<p>Enter a name for the storage pool (required). Note: This field is read-only for the system storage pool, which will always be named “System” storage pool.</p>
Description	Enter a description for the storage pool (optional).
Enable usage alerts	You can optionally select to enable an alert when the storage pool reaches a particular percentage of system use.

Used Threshold

Specify a percentage of system use that, when reached, will cause SAN File System to issue an alert (if the **Enable usage alerts** checkbox is also selected). The range is 1 to 100%.

Related topics:

- “Managing alerts and logs” on page 67
- “Managing storage pools” on page 122
- “Storage pools” on page 37
- “Using the SAN File System console” on page 333

System Overview




The **System Overview** panel is the highest-level monitoring panel in SAN File System. From this panel you can monitor major system components, including cluster state, clients, filesets, servers, hardware, and recent events.

Attributes:

The **Cluster** section of the System overview provides the following details:






Cluster state

States are associated with various icons as follows:

-  Cluster state is either Not running, Offline, Fully quiescent, Partly quiescent, or Online.
-  Cluster state is Forming.
-  Cluster state is Unknown.

Master Workload Transactions

Current actual per-minute number. Following are the icon indicators, which are trend arrows that will indicate graphically the transaction rate changes since the previous update:

-  Major Increase
-  Minor Increase
-  Stable
-  Minor Decrease
-  Major Decrease











Actions:

The **Panel control** section of the System overview allows you to control the panel refresh interval.

Refresh Interval	You can disable refreshing or choose an interval of one, two, five or fifteen minutes. The refresh will occur only if the items monitored by the System Overview panel are active.
Submit	Submits your choice for refresh interval. Note: You will see this selection reflected upon the next panel update, not necessarily when you click the Submit button, unless the panel update interval exceeds the chosen refresh interval.

Fields:

The **Activity and Availability per Server** section of the System Overview lists the following details:

Aggregate status icon	<p>The icon displayed above the table represents the highest-level status icon contained within the table. The possibilities are:</p> <ul style="list-style-type: none">  Severe  Error  Warning  Unknown  Normal
Server	<p>The list of all servers in the cluster. The name links to additional Metadata server details. The master Metadata server will be indicated by the  icon.</p>
Clients	<p>Total number of clients that are currently in session with the associated Metadata server.</p> <p>Note: If five or more clients are in session,  is displayed.</p>
Fileset (Container) Alerts	<p>Alert level for the filesets on the associated Metadata server. Three alert levels are possible:</p> <ul style="list-style-type: none">  Error — Indicates that there are one or more hard quota violations.  Warning — Indicates that there are not any hard quota violations, but one or more usage (Used %), or soft quota violations.  Normal — Indicates that there are no fileset alerts. All filesets are below quotas and Used (%) alerts. <p>Also listed is the total number of filesets that surpass one of the quota-related thresholds (percentage used, soft quota, or hard quota), as well as the highest alert severity level of those that are not “Normal” (that is, Error or Warning).</p>

Load Trend (%)

The relative load change for the associated Metadata server since the previous refresh. The load trend is calculated using the following formula:

$$\text{running average} = (c - n)/t$$

c= total transactions completed up until the current refresh

n= total transactions that were completed before the page was created

t= time elapsed since the page was created

The current average is similar:

$$\text{current average} = (c - r)/d$$

c= total transactions completed up until the current refresh






r= total transactions that were completed up until the last refresh (the one before this one)

d= difference between current time and the time of last refresh.

The % change is calculated by using
(current average - running average) /
running average

This % change determines which arrow displays.

Following are the icon indicators, which are trend arrows that will indicate graphically the load trend changes since the previous update:

-  Major Increase
-  Minor Increase
-  Stable
-  Minor Decrease
-  Major Decrease











The actual percentage value of change is also listed next to the associated trend arrow.

Load Share (%)

The associated Metadata server's percentage share of the total load on all the Metadata servers.

Server State

The state of the associated Metadata server will be one of the following:






-  Not running
-  Failed initialization
-  Initializing
-  Not added
-  Joining
-  Offline
-  Fully quiescent
-  Partly quiescent
-  Online
-  Unknown

Fields:

The **Recent Messages** section of the System Overview displays the ten most recent messages that are in the cluster log, and that match the Message Types setting for this section.

Aggregate message icon

The icon displayed above the table represents the highest-level message icon contained within the table. Five icons are possible:

-  Severe
-  Error
-  Warning
-  Unknown
-  Normal

Actions:

Message Types

You can choose to filter from All Messages or Events Only.





Note: An event is a significant or major message that will often, but not necessarily, trigger an alert.

Submit
Submits your choice for message type.
Note: This selection will be reflected upon the next panel update, not necessarily when you click the **Submit** button, unless the automatic refresh interval is now less than the elapsed time since the last update.

Fields:

Message ID Follows this convention:
 "XXX" Component
 "YY" Subcomponent
 "nnnn" Message code
 "Z" Severity (Severe, Error, Warning, or Information level)

Level Based on the severity level of the Message ID. Four levels are possible:

-  Severe
-  Error
-  Warning
-  Information

Type Messages types can be three categories:

- Normal — Messages that are not events.
- Event — A significant or major message that will often, but not necessarily, trigger an alert.
- Audit
- Trace

Server The Metadata server name that is the source of the message.

Date and Time System (not local) time that the message was issued.

Message Text of the message.

Related topics:

- "Alerts" on page 4
- "Cluster" on page 16
- "Filesets" on page 24
- "Cluster states" on page 19
- "Metadata server" on page 33
- "Metadata server states" on page 35
- Chapter 3, "Monitoring", on page 57
- "Using the SAN File System console" on page 333

Users

Use this panel to view all SAN File System users who are defined and configured in the system by your LDAP server. SAN File System allows various levels of access depending on the tasks each user needs to perform.

Actions:

Timeout All Authorizations Will require LDAP server reauthentication for all users.

Fields:

Name	Lists all users defined on the LDAP server as authorized to perform some level of administrative actions on SAN File System.
Role	The highest level of access that the associated user has within the system.
Authorization	Status of authorization for user's current session. Authorization can be one of two states: Current Always corresponds with an Authorization Timeout of greater than "0 sec." Not Current Always corresponds with an Authorization Timeout of "0 sec."
Authorization Timeout	Short interval of time (measured in seconds) in which users have permission to make requests. During this time the LDAP server will not be contacted for authentication. If authorization is "Not Current" the value will be "0 sec."

Related topics:

- "Administrative security" on page 39
- "Managing users" on page 126
- "User roles" on page 46
- "Using the SAN File System console" on page 333

Using the Help Assistant

The Help Assistant provides field-level descriptions for the actions that you can perform, and the information that you can find, on each panel of the SAN File System console. Because the panel help is embedded into the overall console, you can obtain context-sensitive information without interrupting your current activity. Each time you select a different panel, the information in the Help Assistant changes accordingly.

Each Help Assistant frame contains three possible sections, depending on the type of panel elements being described. The three sections are:


Actions

Provides details about the tasks available from panel drop-down menus.


Fields Provides details about column headings or other field descriptions for which you can make selections or enter configurable information.

Attributes

Provides details about read-only panel attributes, such as those shown on a tabular “notebook” panel displaying property groupings for a particular SAN File System object.

Within each section, the name of the action, field, or attribute appears with a triangular “twistie” icon () next to it . Select the twistie to expand or collapse the item’s description. The item names are initially collapsed to facilitate easy scanning of the available terms and to reduce scrolling. You can choose to select only those items for which you would like additional information and prevent having the Help Assistant frame cluttered with details you do not need.





At the bottom of each Help Assistant frame are links to additional SAN File System topics. These links provide information on related concepts or tasks. When you choose a “Related Topics” link, the information for that topic appears within the Help Assistant frame.



You can launch the Help Assistant by selecting the **Toggle Help Assistant** icon () from the right side of the task bar. Because this button performs in a toggle

Note: You can also close the Help Assistant by selecting the close button for the frame in which it displays.

There are six tool bar controls associated with the Help Assistant:

Table 17. Help Assistant controls

Control	Action
 Back icon	Displays the last topic you viewed in the Help Assistant.
 Forward icon	Displays the topic in the Help Assistant that you viewed before clicking the Back icon.
Search field and Go button	Perform an information search and bring up the search returns in the Information Center.
 Return	Displays panel help in the Help Assistant for the current console panel.
 Print icon	Prints the displayed Help Assistant topic.

You can also view all SAN File System product and help documentation by selecting the question mark icon () located above the work content area, next to the Help Assistant icon (). This action will launch the SAN File System Information Center in a separate window.

Related topics:

- “Using the SAN File System console” on page 333

Volume Properties - Details

Use this panel to view additional attributes for a selected volume.

Attributes:

In addition to its name, ID, and description, the table provides the following information about each volume:

State	Three volume states are possible: Active Only active volumes can have data saved to them. Suspended Volumes cannot have data stored on them when in the suspended state. Being Removed Volumes cannot have data stored on them when they are "Being Removed."
Storage Pool	Name of the storage pool to which the volume belongs.
Size (MB)	Total space available in the associated volume.
Used (MB)	Amount of space currently used by the volume.
Used (%)	The percentage of the total volume size that is currently being used.
OS Device Path	The path that points to the SAN Adapter driver. Note: This driver is local to the SAN File System Metadata server cluster and is not on the LUN storage device.

Related topics:

- "Managing volumes and LUNs" on page 128
- "Storage pools" on page 37
- "Volumes" on page 49
- "Using the SAN File System console" on page 333

Volume Properties - General Settings

Use this panel to change the selected volume's name or description.

Related topics:

- "Managing volumes and LUNs" on page 128
- "Storage pools" on page 37
- "Volumes" on page 49
- "Using the SAN File System console" on page 333

Volumes

Use this panel to perform actions on a SAN File System volume, or view volume properties.

Actions:

Add	Takes you to a four-step wizard to add a volume to a SAN File System storage pool.
Properties	Displays additional detailed properties of the selected volume. Also allows you to change general volume settings.
Remove	Removes and unassigns volumes from a storage pool. Note: After volumes are unassigned, they become available LUNs and can be reassigned whenever needed.
Activate	Activates the selected volume, meaning that it can have data saved to it.
Suspend	Suspends the volume, meaning that it cannot have data saved to it while in this state.

Fields:

You can select a volume on which to view additional details or perform actions. In addition to its description, the table provides the following information about each volume:

Name	Links to additional details about the associated volume.
State	Three volume states are possible: Active Only active volumes can have data saved to them. Suspended Volumes cannot have data stored on them when in the suspended state. Being Removed Volumes cannot have data stored on them when they are "Being Removed."
Storage Pool	Name of the storage pool to which the volume belongs.
Size (MB)	Total space available in the associated volume.
Used (MB)	Amount of space currently used by the volume.
Used (%)	The percentage of the total volume size that is currently being used.

Related topics:

- "Managing volumes and LUNs" on page 128
- "Storage pools" on page 37
- "Volumes" on page 49
- "Using the SAN File System console" on page 333

Appendix H. Prerequisites

This topic introduces the prerequisites for certain tasks.

Prerequisites:

Before performing certain tasks in SAN File System, there are prerequisites that must be confirmed.

Related topics:

- “Data-migration prerequisites”

Supported browsers

Web browser support:

Most SAN File System functions and online documentation are accessed through standard Web browsers. SAN File System supports the following Web browsers (others may work, but have not been tested):

- Internet Explorer 6.0 and above

Note: For Internet Explorer 6.0, Service Pack 1 is also needed.

- Netscape 6.2 and above

Note: While Netscape 6.2 is supported, Netscape 7.0 and above are preferred.

Limitations:

The **Back**, **Forward**, **Refresh** or **Reload** functions of either browser are not supported and may cause unexpected rendering problems. Additionally, opening a hyperlink into a separate browser window is not supported.

Related topics:

- “Using the SAN File System console” on page 333

Data-migration prerequisites

Verify the following conditions before starting the data-migration utility:

- SAN File System and clients must be installed and properly configured.
- SAN File System must be set up with the appropriate containers, pools, policies, and security.
- The clients must be able to access the source file systems (for example, directly-attached, network-attached storage (NAS), or storage area network (SAN) disks) and SAN File System during the data-migration process.
- When migrating data from a Windows client, you must create the destination directory in the SAN File System file system and verify that the security attributes of the destination directory match that of the source directory. Otherwise, the verification phase of data migration will fail and the migrated data will have incorrect permissions.

- All applications that modify the data being migrated (including database and application servers) must be stopped until the migration completes to guarantee data integrity.
- Twice the space of the data is available for migration. Note that the data-migration utility does not verify whether there is enough space in the system pools where data is being migrated.
- Compressed files are expanded during data migration. Sufficient space must be available in the SAN File System to store the expanded files. Refer to the documentation for your operating system to determine the compression ratio and estimate the amount of space required.
- Sparse files become dense, or full, files during data migration. Sufficient space must be available in the SAN File System to store the dense files.
- To invoke the **migratedata** command, you must either supply the full path or update the PATH environment variable to include the migration directory.
For clients based on UNIX[®], this is done in your shell profile (for example, export PATH=\$PATH:/usr/tank/migration/bin).
For clients for Windows, edit the PATH environment variable (for example, from the Control Panel, double-click **System**, and then click the **Advanced** tab and **Environment Variables**. In the **User variables** group, click the PATH variable and then click **Edit**. At the end of the text in the **Variable Value** field, type: **c:\Program Files\IBM\Storage Tank\Migration**).
- You must have superuser privileges (for UNIX-based clients) or administrator privileges (for Windows clients) to migrate data.

Related topics:

- Chapter 4, “Migrating data”, on page 63
- “Data migration” on page 22

Appendix I. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
MW9A/050
5600 Cottle Road
San Jose, CA 95193
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice and represent goals and objectives only.

Trademarks

The following terms are trademarks of International Business Machines Corporation or Tivoli Systems Inc. in the United States or other countries or both:

AIX	IBM	Tivoli
DB2	IBM logo	TotalStorage
Enterprise Storage Server	SecureWay	WebSphere
FlashCopy	StorageTank	xSeries

Java and all Java-based trademarks are registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks or registered trademarks of Microsoft Corporation.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.



Glossary

This glossary includes terms and definitions from:

- *The American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies can be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018. Definitions are identified by the symbol (A) after the definition.
- *The ANSI/EIA Standard - 440A: Fiber Optic Terminology*, copyright 1989 by the Electronics Industries Association (EIA). Copies can be purchased from the Electronics Industries Association, 2001 Pennsylvania Avenue N.W., Washington, D.C. 20006. Definitions are identified by the symbol (E) after the definition.
- *The Information Technology Vocabulary*, developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1). Definitions of published parts of this vocabulary are identified by the symbol (I) after the definition; definitions taken from draft international standards, committee drafts, and working papers being developed by ISO/IEC JTC1/SC1 are identified by the symbol (T) after the definition, indicating that final agreement has not yet been reached among the participating National Bodies of SC1.
- *The Storage Networking Dictionary*, available online at the Storage Networking Industry Association (SNIA) Web site:
www.snia.org/education/dictionary/
- The Distributed Management Task Force (www.dmtf.org), copyright 2003 by the Distributed Management Task Force, Inc., 225 SE Main Street Portland, OR 97214. Definitions derived from this book have the symbol (D) after the definition.

This glossary uses the following cross-reference forms:

- See** This refers the reader to one of two kinds of related information:
- A term that is the expanded form of an abbreviation or acronym. This expanded form of the term contains the full definition.

- A synonym or more preferred term

See also

This refers the reader to one or more related terms.

ACLI. See *Administrative command-line interface (ACLI)*.

Administrative command-line interface (ACLI). A command-line interface used to administer all aspects of the SAN File System. The ACLI runs on all engines that host Metadata servers and the Administrative server.

administrative log. A log that maintains a history of messages created by the Administrative server.

Administrative server. For SAN File System, a set of servlets running within a customized instance of WebSphere Application Server that handles all SAN File System administrative requests from the SAN File System console. See also *SAN File System console*.

alert. A message or other indication that identifies a problem or an impending problem.

audit log. A log that maintains the history of all commands issued by any administrator for all Metadata servers in the cluster.

CIM. See *Common Information Model*.

CIM client application. A storage management program that initiates CIM requests to the Administrative agent for the device.

CIM namespace. The scope within which a CIM schema applies.

CIM object manager (CIMOM). The common conceptual framework for data management that receives, validates, and authenticates the CIM requests from the client application and then directs the requests to the appropriate component or device provider.

CIMOM. See *CIM object manager*.

client. For SAN File System, a client is a system that can access the SAN File System. These clients act as servers to a broader clientele, providing Network File System or Common Internet File System access to the global namespace or hosting applications (such as database servers or Web-hosting services that use multiple servers).

class. The definition of an object within a specific hierarchy. An object class can have properties and methods and serve as the target of an association.

CLI. See *Administrative command-line interface*.

client state manager (CSM). A component of the client kernel that provides protocol support for the client.

cluster. A group of engines that is managed as a set and presents a single point of control for configuration and service activity.

cluster log. A log that maintains a history of messages created by all Metadata servers in the cluster.

cluster state. A status condition of the cluster. Cluster states can be inactive (Not running or Forming), active (Online, Offline, Partly quiescent, or Fully quiescent) or unknown. See also *Forming*, *Fully quiescent*, *Not running*, *Offline*, *Online*, and *Partly quiescent*.

Common Information Model (CIM). A set of standards from the Distributed Management Task Force Inc. (DMTF). CIM provides a conceptual framework for storage management and an open approach to the design and implementation of storage systems, applications, databases, networks, and devices.

coordinated universal time (UTC). The time scale, based on the System International (SI) second, as defined and recommended by the Comitb Consultatif International de la Radio (CCIR) and maintained (using an atomic clock) by the Bureau International des Poids et Mesures (BIPM).

CSM. See *client state manager*.

default user storage pool. A storage pool that stores file data that SAN File System has not assigned (using the active policy) to a user storage pool, as well as file data that is assigned directly to this storage pool. There is only one default user storage pool; however, you can assign any user storage pool as the default storage pool. See also *user storage pool*.

engine. The hardware unit that hosts the software for the Metadata server.

event log. (1) A log that maintains a history of event messages issued by all Metadata servers in the cluster. (2) IBM Term: A log that contains information about events for a particular system or group, for a particular metric, or for all the events that are associated with a specific monitor.

file-placement rule. A rule that controls in what pool SAN File System places files in the global namespace. See also *rule* and *global namespace*.

fileset. A hierarchical grouping of files managed as a unit for balancing workload across a cluster.

FlashCopy image. A space-efficient image of the contents of part of the SAN File System at a particular moment.

Forming. A status condition where the cluster has a master and is in the process of forming. This state is always the initial one whenever a cluster is newly formed.

Fully quiescent. A status condition that cuts off all client communication with the cluster.

global namespace. A single file system that provides complete, shared access to both Windows and UNIX clients in the same environment.

ID. See *identifier*.

identifier (ID). A sequence of bits or characters that identifies a user, program, device, or system to another user, program, device, or system.

Initializing. A status condition during which a Metadata server or the entire cluster is set up for the first time.

key. A property that is used to provide a unique identifier for an instance of a class. Key properties are marked with the Key qualifier. (D)

lease. The amount of time that a client can hold a lock.

lock. A restriction that allows clients to have exclusive access to files. Types of locks include *data locks*, *session locks*, and *range locks*.

logical unit (LU). In open systems, a logical disk drive.

logical unit number (LUN). In the small computer system interface (SCSI) protocol, a unique number used on a SCSI bus to enable it to differentiate between up to sixteen separate devices per SCSI ID address, each of which is a logical unit.

LU. See *logical unit*.

LUN. See *logical unit number*.

managed object format (MOF). A compiled language for defining classes and instances. A MOF compiler offers a textual means of adding data to the CIM Object Manager repository. MOF eliminates the need to write code, thus providing a simple and fast technique for modifying the CIM Object Manager repository. (D)

master Metadata server. In SAN File System, the Metadata server in a cluster that is responsible for physical-space allocation.

metadata. Data that describes the characteristics of stored data; descriptive data.

Metadata server. In SAN File System, a server that offloads the metadata processing from the data-storage environment to improve SAN performance. An instance

of the SAN File System runs on each engine, and together the Metadata servers form a cluster. See also *cluster*.

method. A way to implement a function on a class.

MOF. See *managed object format*.

Not running. A status condition where one or more servers in the cluster are not added and therefore the cluster cannot perform any functions.

object name. An object that consists of a CIM namespace path and a model path. The namespace path provides access to the CIM implementation managed by the CIM agent, and the model path provides navigation within the implementation.

Offline. A status condition during which clients are not being serviced and the cluster is responding only to administrative requests.

Online. A status condition that indicates the normal operational state for the cluster.

Partly Quiesced. A state in which the cluster or server is in a “quiet” client communications mode to allow other operations to occur.

Partly quiescent. A status condition that allows existing metadata activity and client communication to continue on the cluster, but prohibits new communication.

policy. A list of file-placement rules that define characteristics and placement of files. Several policies can be defined within the configuration, but only one policy is active at one time. See also *file-placement rule* and *service-class rule*.

pool. See *storage pool*.

property. An attribute that is used to characterize instances of a class.

qualifier. A value that provides additional information about a class, association, indication, method, method parameter, instance, property, or reference.

quota. A limit on the amount of disk space a user can use.

rule. The lines within a policy that specify which actions will occur when certain conditions are met. Conditions include attributes about an object (file name, type or extension, dates, owner, and groups) and the fileset name associated with the object.

SAN File System console. A Web user interface used to monitor and control the SAN File System remotely by using any standard Web browser.

schema. A group of object classes defined for and applicable to a single namespace. Within the CIM

agent, the supported schemas are loaded through the managed object format (MOF) compiler.

security log. A log that maintains a history of Administrative server login activity.

service location protocol. A directory service that the CIM client application calls to locate the CIMOM.

Shutdown. A status condition that describes when the cluster is shut down as intended.

SLP. See *service location protocol*.

Starting. A status condition when a Metadata server is starting as designed but is not ready to accept connections from clients.

storage pool. A named set of storage volumes that is the destination for storing client data.

symbolic link. A type of file that contains the path name of and acts as a pointer to another file or directory.

system storage pool. A storage pool that contains the system metadata (system and file attributes, configuration information, and Metadata server state) that is accessible to all Metadata servers in the cluster. There is only one system storage pool. See also *Metadata server*.

user storage pool. An optional storage pool that contains blocks of data that compose the files that are created by SAN File System clients. See also *storage pool* and *default user storage pool*.

volume. A labeled logical unit, which can be a physical device or a logical device. For SAN File System, there is a one to one relationship between volumes and LUNs. See also *logical unit number*.

UTC. See *coordinated universal time*

Index

A

About the Administrator's Guide and Reference vii
access levels 39
accessibility
 disability 137
 keyboard 137
 shortcut keys 137
activatevol 148
activating
 volume 148
activating a policy 118, 287
activating a volume 129
adding
 LUNs to SAN File System 129
 SNMP manager 149
 volume to a storage pool 227
additional services, purchasing 325
addsnmpmgr 149
administrative
 commands 142
 log 327
 security 39
 server 3
Administrative CLI
 logging in 141
Administrative CLI (ACLI)
 accessing 139
administrative log
 clearing 68
 viewing 71
administrative user
 creating 234
administrative users
 listing 172, 213
AIX client
 commands 288
 starting 78
AIX client,
 stopping 80
AIX, client for
 controlling disk access 297
 creating 294
 destroying 294
 displaying status for 304
 loading the file-system driver 300
 mounting the global namespace 302
 scanning SAN File System for new
 and removed volumes 297
alerts 4
API method
 backing up 73
assigning
 filesets to a new Metadata server 249
attachcontainer 151
attaching a fileset 151
audit log
 clearing 68, 165
 viewing 71
Audit log 328
authorizations, timing out 128

available LUN 350

B

backing up
 API method 73
 LUN method 72
backup
 file data 5
 metadata 8
backup and recovery (BAR) 5
BAR (backup and recovery) 5
browsers, supported 437
buildscript 153

C

catlog 154
catpolicy 156
CD, publications viii
changing
 fileset settings 103
 policy name 120
changing policy rules 118
changing storage pool settings 123
changing, volume settings 130
chclusterconfig 157
chcontainer 159
check metadata
 panel 352
check metadata progress
 panel 353
checking metadata 112
chpool 161
chvol 163
CIM (Common Information Model) 10
clearing
 audit log 165
 cluster log 165
 logs 68
clearlog 165
client
 commands 288
 creating 293, 294
 destroying 294
 displaying status for 304
 listing active 179
 listing sessions 76
 listing those with root privileges 77
 loading the file-system driver 293,
 300
 mounting the global namespace 293,
 302
 rediscovering volumes accessible
 to 77
 removing 292
 revoking root privileges 77
 setting up 293
 starting 78
 starting AIX 78
client (*continued*)
 starting Windows 79
 stopping 80
 stopping AIX 80
 stopping Windows 81
 unmounting the global
 namespace 305
Client CLI
 accessing 139
client privileges
 panel 354
client sessions
 panel 355
client, granting root privileges 76
client, viewing driver version 75
client, viewing session details 81
client, viewing session statistics 82
client, viewing statistics 61
clients
 security 41
 supported 11
 UNIX-based 13
 Windows-based 14
cluster
 displaying status 267
 log 330
 modifying settings 157
 panel 356
 properties
 details panel 360
 tuning details panel 361
 tuning panel 352, 361
 starting 261
 stop panel 422
 stopping 280
 upgrading 286
cluster configuration, listing 84
cluster details, viewing 85
cluster log
 clearing 68, 165
 viewing 71
cluster states 82
cluster tuning 83
cluster, starting 84
cluster, stopping 84
cluster, viewing statistics 61, 85
collect diagnostic data
 panel 363
collect diagnostic data progress
 panel 364
collectdiag 166
collecting diagnostic data 166
command
 activatevol 148
 addsnmpmgr 149
 attachcontainer 151
 buildscript 153
 catlog 154
 catpolicy 156
 chclusterconfig 157
 chcontainer 159

- command (*continued*)
 - chpool 161
 - chvol 163
 - clearlog 165
 - collectdiag 166
 - detachcontainer 168
 - exit 170
 - help 171
 - lsadmuser 172
 - lsautorestart 175
 - lsclient 179
 - lscontainer 183
 - lsdrfile 187
 - lsengine 189
 - lsimage 192
 - lslun 195
 - lspolicy 199
 - lspool 202
 - lsproc 205
 - lsserver 207
 - lssnmpmgr 211
 - lstrapsetting 212
 - lsvol 213
 - migratedata 289, 308
 - mkcontainer 217
 - mkdrfile 220
 - mkimage 221
 - mkpolicy 223
 - mkpool 225
 - mkvol 227
 - modes 311
 - parameters, standard 312
 - quiescecluster 229
 - quit 231
 - reportvolfiles 232
 - resetadmuser 234
 - restartengine 235
 - resumecluster 236
 - reverttoimage 237
 - rmcontainer 239
 - rmdrfile 241
 - rmimage 242
 - rmpolicy 244
 - rmpool 245
 - rmsnmpmgr 246
 - rmstclient 292
 - rmvol 247
 - setcontainerserver 249
 - setdefaultpool 251
 - setmaster 252
 - setoutput 255
 - settrap 257
 - setupstclient 293
 - startautorestart 259
 - startcluster 261
 - startengine 262
 - startmetadacheck 263
 - startserver 265
 - statcluster 267
 - statcontainer 271
 - statengine 272
 - statsserver 276
 - stfsclient 294
 - stfsdisk 297
 - stfsdriver 300
 - stfsmount 302
 - stfsstatus 304

- command (*continued*)
 - stfsumount 305
 - stopautorestart 279
 - stopcluster 280
 - stopengine 281
 - stopmetadacheck 282
 - stopsver 283
 - suspendvol 285
 - tankpasswd 141
 - usepolicy 287
- commands
 - administrative 142
 - AIX client 288
 - client 288
 - user assistance 142
 - Windows client 307
- Common Information Model (CIM) 10
- components
 - SAN File System 21
- Configuration/Setup Utility 98
- configuring cluster tuning 83
- container (see fileset) 22
- controlling disk access for a client for AIX 297
- conventions, syntax diagram 315
- copying a policy 119
- create recovery file
 - panel 372
- create storage pool
 - add volume 373
 - introduction 374
 - set properties 374
 - verify settings 374
- creating
 - client 293
 - client for AIX 294
 - disaster-recovery dump files 220
 - disaster-recovery script files 153
 - filesets 217
 - FlashCopy image 221
 - FlashCopy image details 107
 - policies 223
 - storage pools 225
- creating a policy 119
- creating a report of statistics 57

D

- data migrating, estimating time for 63
- data migration 63
 - definition 22
 - prerequisites 437
 - stopping 65
- data migration, resuming 65
- data, collecting diagnostic 166
- default storage pool, setting 125
- deleting
 - disaster-recovery dump files 241
 - FlashCopy image 108, 242
 - storage pool 245
- deleting a fileset 105
- Deleting a FlashCopy image 108
- deleting a policy 120
- deleting a recovery file 87
- deleting a storage pool 124
- deleting SNMP managers 69

- destroying
 - client for AIX 294
- detachcontainer 168
- detaching a fileset 168
 - fileset, detaching 105
- details
 - active and inactive policies 199
 - administrative users 172
 - available LUNs 195
 - client 179
 - cluster 267
 - disaster-recovery files 187
 - engine 272
 - engines 189
 - fileset 183
 - FlashCopy image 192
 - Metadata server 276
- details of client session
 - panel 377
- details panel
 - cluster properties 360
 - policy properties 398
- diagnostic data collection
 - engines 94
- disabling
 - automatic restart of the Metadata server 279
- disaster recovery
 - panel 381
- disaster-recovery dump files
 - creating 220
 - deleting 241
- disaster-recovery files
 - listing 187
- disaster-recovery script files
 - creating 153
- discovering LUNs 129
- displaying
 - automatic restart of the Metadata server 175
 - cluster status 267
 - engine status 272
 - log-file content 154
 - long-running processes 205
 - Metadata server status 276
 - policy rules 156
 - SNMP managers 211
 - status for a client for AIX 304
- displaying version of client driver 75

E

- enabling
 - automatic restart of the Metadata server 259
- ending a tanktool session 170, 231
- engine
 - definition 23
 - displaying status 272
 - listing 189
 - powering off 281
 - restarting 97, 235
 - starting 262
- engine fan status 98
- engine power status 99
- engine temperatures 100
- engine, powering off 95

- engine, powering on 96
- engine, viewing statistics 61, 99
- engines
 - collecting diagnostic data on 94
 - listing 95
 - properties
 - fans panel 384
 - power panel 385
 - summary panel 385
 - temperatures panel 387
 - timeouts panel 389
 - vital engine data panel 390
 - voltages panel 390
- engines status summary 100
- engines timeouts 101
- engines voltage status 101
- engines, vital data 102
- estimating time to migrate 63
- event
 - log 329
- event log
 - clearing 68
 - viewing 71
- exit 170

F

- fan status, engine 98
- fan, replacing 51
- file placement, policy-based 43
- file system, global 29
- file-system drive
 - version of 304
- files
 - listing, in a volume 232
- fileset
 - attaching 151
 - changing settings 103
 - creating 217
 - description of 24
 - detaching 168
 - listing 106, 183
 - managing storage space in 132
 - modifying settings 159
 - reassigning to a Metadata server 249
 - removing 239
- fileset (container)
 - panel, attach 348
 - properties
 - Details panel 391
 - General Settings panel 392
- fileset details, viewing 106
- fileset settings, viewing 106
- fileset, attaching 103
- fileset, creating 104
- fileset, deleting 105
- fileset, viewing statistics 61, 106
- filesets
 - panel 394
- first failure data capture 271
- FlashCopy image
 - creating 107, 221
 - deleting 108, 242
 - listing 108, 192
 - reverting to 237
- FlashCopy image, deleting 108

- FlashCopy image, reverting to
 - previous 108
- FlashCopy image, viewing details 109
- FlashCopy images
 - description of 26
- FlashCopy Images of Filesets (Containers)
 - panel 366
- FlashCopy Images of Filesets (Containers)
 - panel 366

G

- getting help 171
- global file system 29
- global namespace
 - attaching a fileset to 151
- granting root privileges to a client 76
- granting user access 126
- guidelines, naming 311

H

- hardware vital product data 58
- help 171
 - general 321
 - online 323
 - telephone 323
- Help Assistant
 - using 433
- hot-swap power supply, installing 52

I

- importing data into the SAN File System 64
- installing hot-swap fan 51
- installing hot-swap power supply 52

L

- label, volume 49
- lease 30
- limited warranty viii
- listing
 - active and inactive policies 199
 - active clients 179
 - administrative users 172, 213
 - automatic restart of the Metadata server 175
 - available LUNs 131, 195
 - client sessions 76
 - client with root privileges 77
 - disaster-recovery files 187
 - engines 95, 189
 - files in a volume 232
 - files on a volume 131
 - fileset 106, 183
 - FlashCopy image 108, 192
 - long-running processes 121, 205
 - LUNs 131
 - Metadata server 112, 207
 - policies 120
 - recovery files 88
 - SNMP managers 211
 - storage pool 124, 202

- listing (*continued*)
 - trap settings 212
 - volumes 131
 - volumes in a storage pool 125
- listing parameters, standard 314
- listing user roles 127
- listing users 127
- loading the client file-system driver 293, 300
- lock 30
- log
 - administrative 31, 327
 - audit 31
 - cluster 31, 330
 - event 31, 329
 - security 31, 330
- log file
 - displaying the content of 154
- logging in to the Administrative CLI 141
- logging into the SAN File System console 405
- logical unit (LUN) 49
- logs 325
 - clearing 68, 165
 - viewing 71
- logs, clearing 68
- long-running processes
 - listing 205
- lsadmuser 172
- lsautorestart 175
- lsclient 179
- lscontainer 183
- lsdrfile 187
- lsengine 189
- lsimage 192
- lslun 195
- lspolicy 199
- lspool 202
- lsproc 205
- lsserver 207
- lssnmpmgr 211
- lstrapsetting 212
- lsvol 213
- LUN details, viewing 134
- LUN method
 - backing up 72
- LUNs
 - adding 129
 - listing 131
 - listing available 131, 195
- LUNs, viewing available 134
- LUNs, viewing statistics 61, 134

M

- managing allocated storage space 132
- managing free storage space 132
- managing storage space 132
- Master console 32
- master console, replacing
 - components 51
- master server, changing 110
- messages, viewing most recent 62
- metadata
 - checking 112
 - server 33

- metadata (*continued*)
 - starting the checker 263
 - stopping the checker 282
- Metadata server
 - disabling automatic restart 279
 - displaying status 276
 - enabling automatic restart 175, 259
 - listing 112, 207
 - properties
 - networking details panel 408
 - setting the master 252
 - software version 407
 - starting 265
 - starting all in the cluster 261
 - stopping 283
 - stopping all in the cluster 280
- Metadata server details, viewing 115
- Metadata server restart service 113
- Metadata server restart service, stopping 115
- Metadata server, determining availability 62
- Metadata server, starting 112
- Metadata server, stopping 114
- Metadata server, viewing activity 62
- Metadata server, viewing statistics 61, 116
- metadata, checking 112
- migrated data, backing out 66
- migratedata 289, 308
- migrating data 63
- migrating data to SAN File System 289, 308
- migration, data 22, 437
- mkcontainer 217
- mkdrfile 220
- mkimage 221
- mkpolicy 223
- mkpool 225
- mkvol 227
- modifying
 - cluster settings 157
 - fileset settings 159
 - pool settings 161
 - volume settings 163
- monitoring 57
- mounting the global namespace on a client for AIX 302
- mounting the global namespace on the client 293

N

- name
 - changing for a policy 120
- naming guidelines 311
- navigating by keyboard 137
- notices 439
- notices used in this guide viii

O

- one-button data collection utility 57
- output format
 - setting 255
- Overview 1

P

- panel
 - available LUNs 350
 - check metadata 352
 - check metadata progress 353
 - cluster 356
 - collect diagnostic data 363
 - collect diagnostic data progress 364
 - create recovery file 372
 - disaster recovery 381
 - policies 397
 - Signon 405
 - system overview 428
- policies
 - listing 120
 - listing active and inactive 199
 - removing 244
- Policies panel 397
- policy
 - activating 287
 - changing the name of 120
 - copying 119
 - creating 223
 - properties
 - details panel 398
 - rules panel 399
- policy details, viewing 121
- policy rules
 - displaying 156
- Policy wizard
 - Add Rules 369
 - Edit Rules 371
 - High-Level Settings 371
 - Introduction 372
- policy, activating 118
- policy, changing rules 118
- policy, creating 119
- policy, deleting 120
- pool
 - creating 225
 - default 38
 - modifying settings 161
 - overview 37
 - setting the default 251
 - system 37
 - user 37
- power status, engine 99
- power supply, installing 52
- powering off an engine 281
- powering off the engine 95
- powering on the engine 96
- prerequisites 437
 - data migration 437
- privileges
 - listing clients with root 77
 - revoking from a client 77
- privileges, granting to a client 76
- process limits 122
- processes
 - listing 121
 - listing long-running 205
- publications viii
- publications CD viii

Q

- quiescecluster 229
 - quiescing 229
 - resuming 236
- quiescing the cluster 229
- quit 231

R

- reassigning
 - filesets to a new Metadata server 249
- rebooting engine 235
- rebooting the engine 97
- recovery 5
- recovery file, creating 87
- recovery file, deleting 87
- recovery files
 - listing 88
- rediscovering
 - volumes accessible to a client 77
- release notes viii
- remote access 39
- removing
 - client 292
 - fileset 239
 - policies 244
 - SNMP manager 246
 - volume 247
- removing down-level Metadata server software 117
- removing volumes from a storage pool 133
- replacing
 - master console components 51
 - storage engine components 51
- replacing hot-swap fan 51
- report, creating 57
- reportvolfiles 232
- resetadmuser 234
- resetting
 - administrative user 234
- restart service 113
- restart service, stopping 115
- restartengine 235
- restarting
 - Metadata server automatically 259, 279
- restarting engine 235
- restarting the engine 97
- resumecluster 236
- resuming a data migration 65
- resuming the cluster 236
- reverting to a FlashCopy image 108, 237
- reverttoimage 237
- revoking root privileges from a client 77
- rmcontainer 239
- rmdrfile 241
- rmimage 242
- rmpolicy 244
- rmpool 245
- rmsnmpmgr 246
- rmstclient 292
- rmvol 247
- roles 126
- root privileges
 - listing clients with 77

- root privileges (*continued*)
 - revoking from a client 77
- root privileges, granting to a client 76
- rules panel, policy properties 399
- rules, changing for a policy 118
- running Administrative commands 140

S

- safety information viii
- safety notices, translated viii
- SAN File System accessibility
 - features 137
- SAN File System console
 - banner 334
 - closing 334
 - entry fields 343
 - Help Assistant 340
 - Information Center 340
 - messages 342
 - My Work frame 337
 - overview 333
 - property notebook 343
 - status and progress indicators 340
 - system overview 343
 - tables
 - sorting and filtering 340
 - task bar 334
 - title bar 334
 - User assistance 340
 - using 333
 - overview 333
 - status, help, and tables 340
 - tasks and panels 336
 - work area 338
- SAN File System, importing data to 64
- security
 - administrative 39
 - clients 41
 - File System Access Security
 - feature 39
 - log 330
- security log
 - clearing 68
 - viewing 71
- server
 - administrative 3
 - metadata 33
 - properties
 - details panel 407
- servers
 - panel 410
 - stop panel 423
- service
 - phone 323
 - support 322
- Service Alert 9
- setcontainerserver 249
- setdefaultpool 251
- setmaster 252
- setoutput 255
- setting
 - master Metadata server 252
 - SNMP traps 257
- setting output format 255
- setting the default storage pool 251
- setting up the clients 293, 304

- settings
 - listing, trap 212
- settrap 257
- setupstclient 293
- shutting down the engine 95
- signing onto the SAN File System
 - console 405
- Simple Network Management Protocol (SNMP)
 - components 42
 - traps 4
- skills needed to administer this
 - product vii
- SNMP
 - adding a manager 149
 - listing managers 211
 - removing a manager 246
 - setting a trap 257
- SNMP (Simple Network Management Protocol)
 - components 42
 - traps 4
- SNMP manager, adding 67
- SNMP managers, deleting 69
- SNMP managers, modifying 69
- SNMP managers, viewing 71
- SNMP traps, modifying 70
- SNMP traps, setting up 70
- software vital product data 59
- standard
 - command parameters 312
 - listing parameters 314
- Standby mode 95
- startautorestart 259
- startcluster 261
- startengine 262
- starting
 - AIX client 78
 - all Metadata servers in the
 - cluster 261
 - client 78
 - metadata checker 263
 - tanktool session 140
 - Windows client 79
- starting an engine 262
- starting an Metadata server 265
- starting the engine 96
- startmetadatacheck 263
- startserver 265
- statcluster 267
- statcontainer 271
- statengine 272
- statistics
 - client sessions panel 413
 - cluster panel 414
 - create report panel 415
 - engines panel 416
 - filesets (containers) panel 417
 - LUNs panel 417
 - report panel 418
 - servers panel 419
 - storage pools panel 419
 - volumes panel 421
- statistics, creating a report of 57
- statistics, viewing cluster 85
- statistics, viewing fileset 106
- statistics, viewing for 135

- statistics, viewing for client sessions 82
- statistics, viewing for engine 99
- statistics, viewing for LUNs 134
- statistics, viewing for major
 - components 61
- statistics, viewing for Metadata
 - server 116
- statistics, viewing for storage pools 126
- statserver 276
- status
 - cluster 267
 - engine 272
 - Metadata server 276
- status, engine 100
- stfsclient 294
- stfsdisk 297
- stfsdriver 300
- stfsmount 302
- stfsstatus 304
- stfsumount 305
- stop panel, cluster 422
- stop panel, servers 423
- stopautorestart 279
- stopcluster 280
- stopengine 281
- stopmetadatacheck 282
- stopping
 - AIX client 80
 - all Metadata servers in the
 - cluster 280
 - client 80
 - metadata checker 282
 - stopping a data migration 65
 - stopping a Metadata server 114
 - stopping an Metadata server 283
 - stopping client for Windows
 - Windows client 81
 - stopping Metadata server restart
 - service 115
 - stopping the cluster 84
 - stopping the engine 95
 - stopserver 283
 - storage device, supported 46
 - storage engine, replacing components 51
 - storage management 43
 - storage pool
 - adding volumes to 227
 - creating 225
 - deleting 245
 - listing 124, 202
 - listing volume in 125
 - managing storage space in 132
 - setting the default 251
 - storage pool details, viewing 125
 - storage pool, changing settings 123
 - storage pool, creating 123
 - storage pool, default 125
 - storage pool, deleting 124
 - storage pool, viewing statistics 61, 126
 - storage space
 - managing 132
- support
 - general 321
 - online 323
 - service 322
 - telephone 323

- supported browsers
 - limitations 437
- suspending
 - volume 133
- suspending a volume 285
- suspendvol 285
- syntax diagram conventions 315
- system overview
 - panel 428
- system status, viewing 62

T

- tankpasswd 141
- tanktool
 - ending the session 170, 231
 - getting help 171
 - starting a session 140
 - utility 140
- task steps for filtering tables 341
- task steps for sorting tables 341
- temperature engine 100
- timeouts, engine 101
- trademarks 440
- traps
 - listing settings 212
- traps, Simple Network Management Protocol (SNMP) 4
- tuning details panel, cluster 361
- tuning panel, cluster 352, 361
- tuning, cluster, details 85

U

- UNIX-based clients 13
- unmounting the global namespace on a client for AIX 305
- upgrading cluster 286
- usepolicy 287
- user access, granting 126
- user authorizations, timing out 128
- user interface
 - Web-based 47
- user role
 - Administrator 47
 - backup 47
 - monitor 46
 - operator 47
- user roles, listing 127
- users, listing 127
- Using the SAN File System console 333
 - overview 333
 - status, help, and tables 340
 - tasks and panels 336
- utilities
 - tanktool 140

V

- version, cluster software 86
- viewing
 - logs 71
- viewing available LUN details 134
- viewing client-session statistics 82
- viewing client—session details 81
- viewing cluster details 85

- viewing cluster statistics 85
- viewing cluster tuning details 85
- viewing engine statistics 99
- viewing fileset details 106
- viewing fileset settings 106
- viewing fileset statistics 106
- viewing FlashCopy image details 109
- viewing LUN details 134
- viewing LUN statistics 134
- viewing Metadata server details 115
- viewing Metadata server statistics 116
- viewing policy details 121
- viewing statistics for major components 61
- viewing storage pool statistics 126
- viewing system status 62
- viewing the cluster software version 86
- viewing volume statistics 135
- vital engine data 102
- vital product data (VPD)
 - hardware 58
 - software 59
- voltage status, engine 101
- volume
 - activating 148
 - adding to a storage pool 227
 - definition 49
 - listing files in 232
 - listing files on 131
 - listing in a storage pool 125
 - modifying settings 163
 - rediscovering those accessible to a client 77
 - removing 247
 - scanning SAN File System for 297
 - suspending 133, 285
- volume details 135
- volume label 49
- volume settings 135
- volume, activating 129
- volume, changing settings 130
- volume, viewing statistics 61, 135

volumes

- listing 131

volumes, adding to storage pool 130

VPD (vital product data)

- hardware 58
- software 59

W

- Web browsers supported 437
- Web sites ix, 321
- Who should use this guide vii
- Windows client
 - command 307
 - starting 79
 - stopping 81
- Windows-based client
 - administrative privileges 15
 - case sensitivity 14
 - command line interface (CLI) 15
 - file management 15
 - NTFS features 15
- Windows-based clients 14

Readers' Comments — We'd Like to Hear from You

IBM TotalStorage™ SAN File System
(based on IBM Storage Tank™ technology)
Administrator's Guide and Reference
Version 1 Release 1

Publication No. GA27-4317-00

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Complete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to find	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Easy to understand	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Well organized	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Applicable to your tasks	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? Yes No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.



Fold and Tape

Please do not staple

Fold and Tape



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corp
Dept. CGFA
PO Box 12195
Research Triangle Park, NC 27709-9990



Fold and Tape

Please do not staple

Fold and Tape



Printed in U.S.A.

GA27-4317-00

