



Sun StorageTek™ 5800 System Administration Guide

Sun Microsystems, Inc.
www.sun.com

Part No. 819-3810-10
May 2006, Revision A

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

Copyright 2006 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

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

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

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

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

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices.

SSH is a registered trademark of SSH Communications Security in the United States and in certain other jurisdictions.

Sun, Sun Microsystems, the Sun logo, Sun StorageTek, Java, AnswerBook2, JumpStart, docs.sun.com, et Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and in other countries.

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

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

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

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

Copyright 2006 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, Californie 95054, États-Unis. Tous droits réservés.

Sun Microsystems, Inc. possède les droits de propriété intellectuelle relatifs à la technologie décrite dans ce document. En particulier, et sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs des brevets américains listés sur le site <http://www.sun.com/patents>, un ou les plusieurs brevets supplémentaires ainsi que les demandes de brevet en attente aux États-Unis et dans d'autres pays.

Ce document et le produit auquel il se rapporte sont protégés par un copyright et distribués sous licences, celles-ci en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a.

Tout logiciel tiers, sa technologie relative aux polices de caractères, comprise, est protégé par un copyright et licencié par des fournisseurs de Sun.

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

SSH est une marque déposée registre de SSH Communications Security aux États-Unis et dans certaines autres juridictions.

Sun, Sun Microsystems, le logo Sun, Sun StorageTek, Java, AnswerBook2, JumpStart, docs.sun.com, et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux États-Unis et dans d'autres pays.

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

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

LA DOCUMENTATION EST FOURNIE "EN L'ÉTAT" ET TOUTES AUTRES CONDITIONS, DÉCLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES DANS LA LIMITE DE LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE À LA QUALITÉ MARCHANDE, À L'APTITUDE À UNE UTILISATION PARTICULIÈRE OU À L'ABSENCE DE CONTREFAÇON.



Contents

Preface xiii

1. Sun StorageTek 5800 System Overview 1

Product Overview 1

 Clustered Design 2

 Sun StorageTek 5800 System Cell 3

Hardware Overview 4

 Serviceability 5

Software Overview 6

 Metadata 7

 Data and Administrative System Access 8

 Data Access in Sun StorageTek 5800 System Storage 8

 APIs 8

 File System Views 9

 Data Reliability and Availability 9

 Placement Algorithms 11

 Stored Data Object Fragment Placement 11

 Node Failure and Recovery 13

2. Customizing the Configuration	15
Setting Network Configuration Parameters	15
Configuring an SMTP Server	16
Configuring a Gateway	16
Configuring External NTP Servers	16
Configuring an External Log Host	16
Configuring Authorized Clients	17
Configuring the Alert Subsystem	17
Setting External Virtual IP Addresses	17
3. Monitoring the System	19
Obtaining System Version and Status	19
Viewing the System Software Version	20
Obtaining FRU Listings	21
Getting Disk Status	22
Obtaining SMTP Email Alerts	23
4. Configuring Metadata and File System Views	25
Understanding Metadata and the System Schema	25
Relational Metadata	26
Metadata Structure and the Schema	28
Metadata Configuration and the XML File Structure	28
Fully-Qualified Names	30
DNS Namespaces	30
Reserved Domains	31
Schema Definition	31
Upgrading the Schema Configuration	32
Understanding File System Principles	33
Using WebDAV for File Browsing	34

Metadata Attributes and WebDAV Properties	35
Virtual View Definition	36

5. **Administering the System** 37

Managing Passwords or Public Keys	37
Changing the System Password	38
Configuring a Public Key	38
Executing a Script	39
Powering Up the System	40
Shutting Down the System	40
Rebooting the System	41
Recovering From a Power Failure	41
Upgrading the System	41
Wiping Data From the System	43

A. **CLI Command Reference** 45

alertcfg	46
Description	46
Syntax	46
Options	46
Examples	46
df	47
Description	47
Syntax	47
Options	47
Examples	47
help	48
Description	48
Syntax	48

Options	48
Examples	48
hwcfg	49
Description	49
Syntax	49
Options	49
Examples	49
hwstat	51
Description	51
Syntax	51
Options	51
Examples	52
ifconfig	54
Description	54
Syntax	54
Options	54
Examples	54
license	55
Description	55
Syntax	55
Options	55
Examples	55
logout	56
Description	56
Syntax	56
Options	56
Examples	56

mdconfig	57
Description	57
Syntax	57
Options	57
Example	58
netcfg	59
Description	59
Syntax	59
Options	59
Examples	60
passwd	61
Description	61
Syntax	61
Options	61
Examples	61
reboot	63
Description	63
Syntax	63
Options	63
Examples	63
shutdown	64
Description	64
Syntax	64
Options	64
Examples	64
sysstat	65
Description	65
Syntax	65

Options	65
Examples	65
upgrade	67
Description	67
Syntax	67
Options	67
Examples	67
version	68
Description	68
Syntax	68
Options	68
Example.	68
wipe	69
Description	69
Syntax	69
Options	69
Examples	69

Glossary 71

Index 75

Figures

FIGURE 1-1	Sun StorageTek 5800 System Clustered Architecture	2
FIGURE 1-2	Sun StorageTek 5800 System Half-Cell and Single Cell Silo	3
FIGURE 1-3	Sun StorageTek 5800 System Redundant Network	4
FIGURE 1-4	Sun StorageTek 5800 System View (Front)	5
FIGURE 1-5	Front Panel Storage Node Controls	6
FIGURE 1-6	File System View Example	9
FIGURE 1-7	5+2 Encoding Example With 5 Data Fragments and 2 Parity Fragments	10
FIGURE 1-8	Decoding Example With Missing Fragments	10
FIGURE 1-9	Data and Parity Fragment Storage Example	12
FIGURE 1-10	Node Failure and Fragment Relocation Example	13
FIGURE 4-1	The addMetadata Operation	27

Tables

TABLE 1-1	Sample Medical Record Schema	7
TABLE 4-1	Valid Metadata Types	27
TABLE 4-2	Sun StorageTek 5800 System Reserved Domains	31
TABLE 4-3	Sun StorageTek 5800 System Reserved Domains	31
TABLE 4-4	WebDAV Property Names and User Metdata Attributes	35

Preface

This document is written for system administrators of the Sun StorageTek™ 5800 System. The administrator should be familiar with network troubleshooting and UNIX system administrator activities.

How This Book Is Organized

[Chapter 1](#) provides an overview of the system hardware and software, as well as an introduction to the system components and software functionality.

[Chapter 2](#) details the functions of the `netcfg` and `alertcfg` commands and explains how they are used to configure system parameters and the alert subsystem.

[Chapter 3](#) describes the monitoring capabilities of the Sun StorageTek 5800 system.

[Chapter 4](#) provides more detail on metadata and the system schema, discusses upgrading the schema configuration, and then describes file system views and their definition.

[Chapter 5](#) explains how to perform routine maintenance on the Sun StorageTek 5800 system.

[Appendix A](#) provides instructions on how to access the CLI, as well as descriptions, syntax, and examples for each of the CLI commands.

The [Glossary](#) provides a list of pertinent words and phrases and their definitions for the Sun StorageTek 5800 system.

Using UNIX Commands

This document does not contain information on basic UNIX® commands and procedures. Refer to the following for this information:

- Software documentation that you received with your system
- Solaris™ Operating System documentation, which is at <http://docs.sun.com>

Shell Prompts

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

Typographic Conventions

Typeface*	Meaning	Examples
AaBbCc123	The names of commands, files, and directories; on-screen computer output	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. % You have mail.
AaBbCc123	What you type, when contrasted with on-screen computer output	% su password:
<i>AaBbCc123</i>	Book titles, new words or terms, words to be emphasized. Replace command-line variables with real names or values.	Read Chapter 6 in the <i>User's Guide</i> . These are called <i>class</i> options. You <i>must</i> be superuser to do this. To delete a file, type <code>rm filename</code> .

* The settings on your browser might differ from these settings.

Related Documentation

The following are available online at Sun's documentation web site:

www.sun.com/documentation

Application	Title	Part Number	Format
Initial instructions for the installation and configuration of the system	<i>Sun StorageTek 5800 System Getting Started Guide</i>	819-3808-10	PDF and HTML
Late-breaking material not included in the documentation set	<i>Sun StorageTek 5800 System Release Notes</i>	819-3813-10	PDF and HTML

Note – In addition, there are two programmer guides available for the Sun StorageTek 5800 System, including a Client API Reference (part no. 819-3811-10) and an SDK Developer's Guide (part no. 819-5501-10). For copies of these documents, contact your local sales or support representative.

Documentation, Support, and Training

Sun Function	URL
Documentation	http://www.sun.com/documentation/
Support	http://www.sun.com/support/
Training	http://www.sun.com/training/

Third-Party Web Sites

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

Sun Welcomes Your Comments

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

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

Please include the title and part number of your document with your feedback:

Sun StorageTek 5800 System Administration Guide, part number 819-3810-10

Sun StorageTek 5800 System Overview

This chapter introduces the hardware and software that make up the Sun StorageTek 5800 system. It provides an introduction to the system components and software functions and contains the following sections:

- [“Product Overview” on page 1](#)
- [“Hardware Overview” on page 4](#)
- [“Software Overview” on page 6](#)

Product Overview

The Sun StorageTek 5800 system is an online storage appliance featuring a fully integrated hardware and software architecture in which the disk-based storage nodes are arranged in a symmetric cluster. Data can be associated with metadata for easy reference as discussed in [“Metadata” on page 7](#).

Both data and metadata are distributed across disks and nodes. There is no dedicated metadata server or master node, and the system presents a simple, single image for client and administrator access. The clustered and redundant design provides high availability, good performance, and exceptional data integrity.

Clustered Design

The Sun StorageTek 5800 system uses a symmetric, clustered, architecture (FIGURE 1-1). All storage control, data, and metadata path operations are distributed across the cluster to provide both reliability and performance scaling. Each node is independent of all other nodes, and there is complete symmetry in both hardware and software on each node.

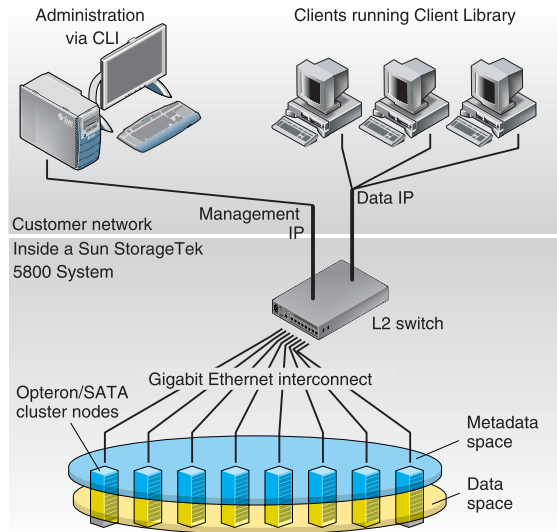


FIGURE 1-1 Sun StorageTek 5800 System Clustered Architecture

The Sun StorageTek 5800 system's stateless design principles ensure that there is no single point of failure or contention for resources. Its cost of ownership, maintainability, and reliability designs also help improve cluster scaling.

A Sun StorageTek 5800 system consists of a *cluster* of nodes in which each node contains a standard processor, memory, and storage disks. The cluster is accessed using an application programming interface (API) or a file system view. For more information, see [“Data Access in Sun StorageTek 5800 System Storage” on page 8](#).

Each node in the system is capable of handling data operations using this API. A load-spreading switch balances traffic from the clients to the nodes. Some housekeeping functions are delegated to an elected master node, although those services can fail over to any other node in the system without affecting reliability or availability.

Sun StorageTek 5800 System Cell

A Sun StorageTek 5800 system or *silo* consists of a half-cell of 8 nodes or a full cell of 16 nodes. (Multicell systems are not currently available.)

Half-Cell
Sun StorageTek 5800
System Silo



Full-Cell
Sun StorageTek 5800
System Silo

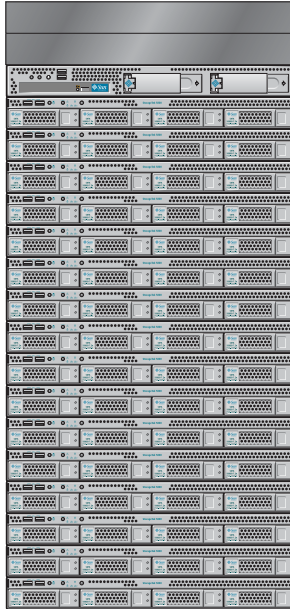


FIGURE 1-2 Sun StorageTek 5800 System Half-Cell and Single Cell Silo

The *cell* is the basic building block of the system and consists of 8 or 16 Sun StorageTek 5800 system storage nodes, 2 Load Balancers, and 1 Service Node. Each node of the Sun StorageTek 5800 system is connected to both Load Balancers for high availability, forming a dual-star topology. The remaining ports of the Load Balancer are used for front-end and uplink connections and heartbeat connections for switch failover ([FIGURE 1-3](#)).

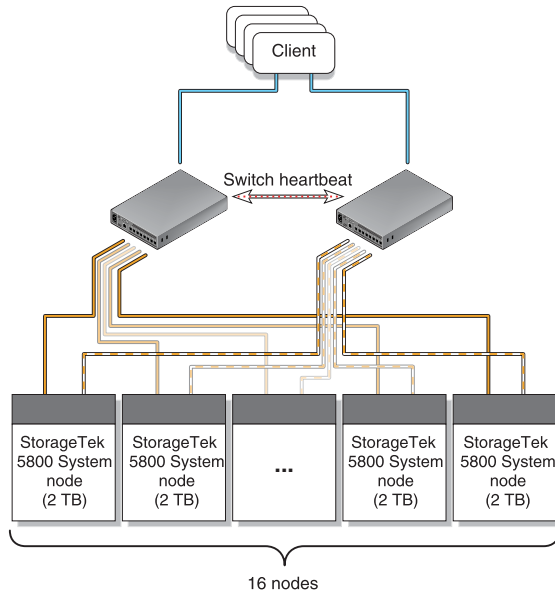


FIGURE 1-3 Sun StorageTek 5800 System Redundant Network

Hardware Overview

The Sun StorageTek 5800 system ([FIGURE 1-4](#)) is a rack-mounted system designed to have no single point of failure and to be serviceable without disruption. Each 1U Sun StorageTek 5800 system server node runs the Solaris OS and consists of a 1U socket 939 AMD Opteron™ processor, a server management board, and 4 Serial ATA (SATA), 3.5 inch (8.89 cm) drives.

A pair of Load Balancers provide for failover and load spreading across nodes. Because the system features a fail-in-place self-healing design, much of the urgency normally associated with switch, disk, Network Interface Card (NIC), Central Processing Unit (CPU), or other hardware failures is removed.

The minimum system configuration consists of a half-cell of eight storage nodes with four SATA 500-gigabyte (GB) drives per node, two Load Balancers, and one Sun Fire™ x2100 server acting as the service node. The maximum configuration consists of one cell of 16 storage nodes with four SATA 500-GB drives per node, two Load Balancers, and one Sun Fire x2100 server acting as the service node ([FIGURE 1-4](#)).

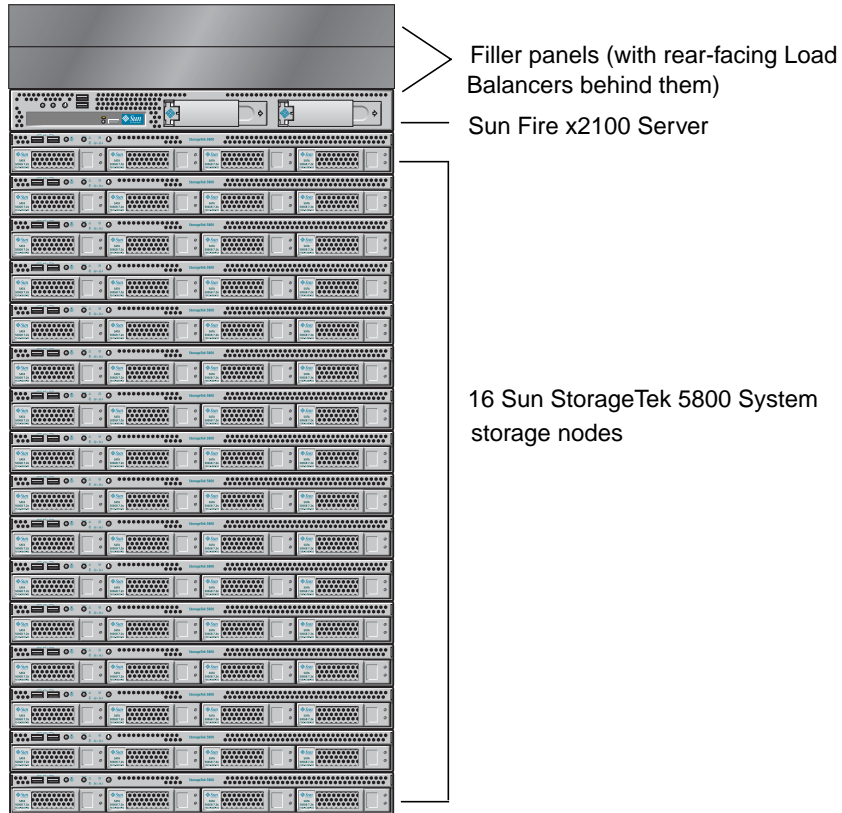


FIGURE 1-4 Sun StorageTek 5800 System View (Front)

Serviceability

The Sun StorageTek 5800 system features a series of field-replaceable units (FRUs). Components can fail in place without being immediately serviced. When a component fails, alerts are sent through email. If a disk fails, a drive fault LED on the front panel ([FIGURE 1-5](#)) lights up to indicate the fault.

In addition, command-line interface (CLI) commands such as `hwstat` provide status information on the failed components. The [“CLI Command Reference” on page 45](#) contains more information on this and all of the other CLI commands. For instructions on connecting to the system to execute the CLI commands, see the *Sun StorageTek 5800 System Getting Started Guide*.

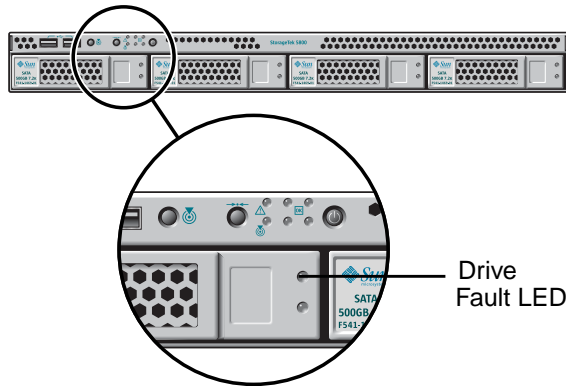


FIGURE 1-5 Front Panel Storage Node Controls

Software Overview

The Sun StorageTek 5800 system is a unique product that combines servers, storage, networking, and distributed-systems software in a single solution. Many of its features are implemented in software, thus permitting:

- Lower costs for the underlying hardware platform
- Easy field-upgrades for existing systems
- Future flexibility with the platform itself

The Sun StorageTek 5800 system software includes a variety of layers that all contribute to the overall product goal of reducing costs while improving data manageability. For example, the following components work seamlessly and operate with complete transparency to end users:

- Software system upgrade management
- Management, logging, and telemetry systems
- Data fragmentation, placement, and healing algorithms
- Cluster management
- Software development kit (SDK)
- Metadata and search systems

Metadata

Metadata is extra information about the data object. There are two main types of metadata in the Sun StorageTek 5800 system: system and user, or extended, metadata. The *system metadata* includes a unique identifier for each stored object, called the object ID or OID, as well as information on creation time (ctime), data length, and data hash.

The *OID* is returned by the API when an object is stored and used to retrieve the object. It is also returned when queries are made against *user metadata* that has been associated with the OID. For more information, see [“Understanding Metadata and the System Schema” on page 25](#).

The Sun StorageTek 5800 system’s user or extended metadata provides the ability to store application-level attributes associated with data objects. User metadata also allows you to arbitrarily define a *schema* (using the EXtensible Markup Language (XML)). Typically, user queries are executed against application-stored, user metadata. Optionally, they can be issued against system metadata as well.

You can define a set of metadata attributes associated with your stored objects. For example, in an application that stores medical records, the metadata attributes include things like patient name, doctor’s name, reason for visit, deductible, medical record number and insurance company. You can then run a query to retrieve the record using these fields, or combine the query to retrieve all records for a given doctor or insurance company on a particular date. Extensible metadata provides unlimited scope when designing the application.

The schema defines the way that the Sun StorageTek 5800 system metadata is structured. It consists of attributes, each of which has a defined type. For example, the previous medical record might contain the names of attributes shown in [TABLE 1-1](#).

TABLE 1-1 Sample Medical Record Schema

Attribute	Type
PatientName	String
DoctorName	String
MedicalRecordNumber	Long
InsuranceCompany	String
Deductible	Double
ReasonForVisit	String

The attributes defined in the schema can be assigned values when data is stored. There is a single schema for the system. However, you do not need to specify all attributes when storing an object.

Because metadata is stored on the Sun StorageTek 5800 system and not with the client application, it scales along with stored data, and gains the same advantages of integrity and availability as the data.

Data and Administrative System Access

The Sun StorageTek 5800 system exports two IP addresses: one for data access and one for administrative access. Data is accessed using a single IP address, called the data virtual IP (VIP) address. Your interaction with the Sun StorageTek 5800 system does not require a knowledge of the underlying hardware. Instead, you access it as a single very large storage system through the API.

You perform administrative tasks on the Sun StorageTek 5800 system using the CLI available through the administrative VIP. You access the CLI with the `ssh`® command, and if you wish, you can script CLI commands. Administrators can monitor individual components including disks and nodes through the CLI and can also enable and disable individual disks and nodes.

The CLI supports standard administrative tasks such as shutting down, powering down, rebooting, and obtaining CLI help. If you wish, you can also configure the Sun StorageTek 5800 system to send email alerts. For more information on each of the specific CLI commands and their functions, see the [“CLI Command Reference” on page 45](#).

Data Access in Sun StorageTek 5800 System Storage

You access data in Sun StorageTek 5800 system storage in one of two ways:

- Through an API (Java and C client libraries)
- Through a file system view with WebDAV

APIs

The Sun StorageTek 5800 system Java and C APIs enable you to store, retrieve, query, and delete data and metadata through Java™ and C client libraries. Sample applications and command-line routines are provided in the SDK to demonstrate the Sun StorageTek 5800 system’s capabilities as well as provide good programming examples. The SDK also provides an emulator that enables you to test client applications without having to set up a StorageTek 5800 system server. For more

information on the SDK, see the *Sun StorageTek 5800 System SDK Developer's Guide*. For more information on the Java and C client APIs, see the *Sun StorageTek 5800 System Client API Reference Guide*.

File System Views

The Sun StorageTek 5800 system contains no internal hierarchical path structure. Virtual views are queries against metadata that are expressed externally as file system paths and file names. A virtual file system view is defined using the metadata attributes defined in the schema.

For example, using the preceding medical record schema shown in [TABLE 1-1](#), you can define a view that is organized at the top level directory by doctor name, then by patient name at the second level, and so on. Opening the top-level view for a given doctor shows a list of patients for that doctor ([FIGURE 1-6](#)).

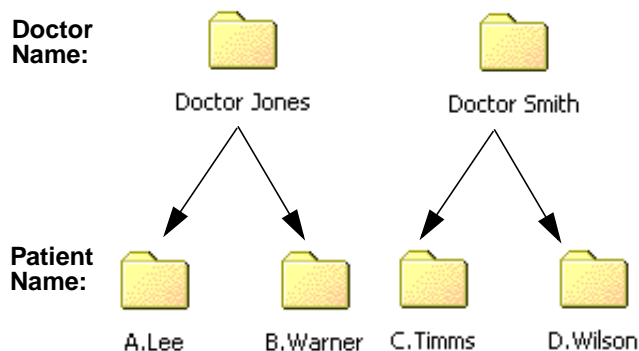


FIGURE 1-6 File System View Example

The system enables you to browse through virtual file system views using the Web-based Distributed Authoring and Versioning (WebDAV) protocol. WebDAV is a set of extensions to the HTTP protocol that provides a network protocol for creating interoperable, collaborative applications. For more information on WebDAV, see [“Using WebDAV for File Browsing” on page 34](#).

Data Reliability and Availability

To provide reliability, files are protected in the Sun StorageTek 5800 system using the *Reed Solomon (RS) encoding algorithm*. The RS algorithm is part of a code family that efficiently builds redundancy into a file to guarantee reliability in the face of multiple failures in the storage system.

The Sun StorageTek 5800 system stores *fragments* of files across multiple disks and nodes using 5+2 encoding. Thus, when an object of any type (for example, an MP3 binary or a text file) is stored in the Sun StorageTek 5800 system, it is divided into five data fragments and two corresponding parity fragments (FIGURE 1-7).

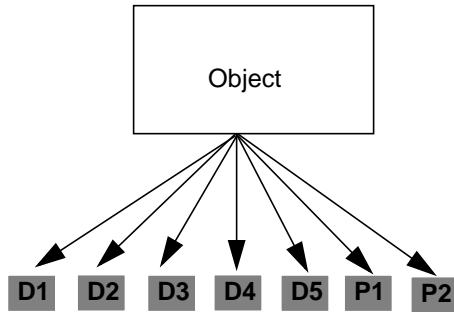


FIGURE 1-7 5+2 Encoding Example With 5 Data Fragments and 2 Parity Fragments

These fragments are stored on different disks in the system (FIGURE 1-9). The 5+2 encoding means that the Sun StorageTek 5800 system can tolerate up to two missing data or parity fragments. For example, if D2 and P1 are not accessible, the Sun StorageTek 5800 system can still reconstruct the object using the remaining fragments displayed in FIGURE 1-8.

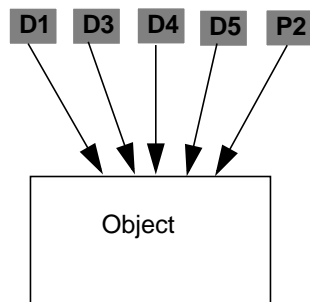


FIGURE 1-8 Decoding Example With Missing Fragments

Note – The placement of each fragment is designed with the physical hardware in mind and is done in such a way that it maximizes the separation of fragments from one another and minimizes the chance of a component failure leading to a loss of more than one fragment.

Placement Algorithms

The Sun StorageTek 5800 system uses internal placement algorithms to determine where data and parity fragments are placed in the cluster. The algorithms' goals are to:

- Avoid keeping a table that maps identifiers to physical locations.
A table requires constant updating as locations change in response to failures or changes in cluster membership.
- Minimize resource thrashing resulting from failure or changes in cluster membership.
- Enable cluster-wide parallel recovery in the event of drive or node failure.
- Maximize reliability and availability by ensuring that no two file fragments get stored on the same device and node.

Stored Data Object Fragment Placement

The fact that an object is divided into data and parity fragments is invisible to the user. The system gathers and decodes the appropriate fragments to reconstruct stored data objects as read requests are made. Any node can encode and store data, and any node can decode and return it.

[FIGURE 1-9](#) provides an example of how the fragments may be distributed in a Sun StorageTek 5800 system cluster.

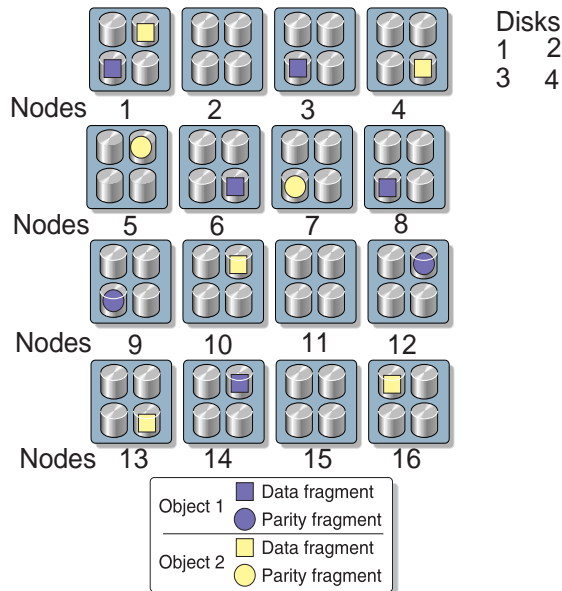


FIGURE 1-9 Data and Parity Fragment Storage Example

In [FIGURE 1-9](#), two different object files are shown as divided into five data fragments each that are then placed on different disks in different nodes. The two parity fragments that were generated for each of the two different objects are also placed on different disks on different nodes. Because the parity fragments are generated based on the contents of the data fragments, they can be decoded in combination with data fragments to recreate missing data fragments.

The data encoding and reconstruction process means that any two disks on different nodes, and more than two disks on the same node, can be lost without losing data. Sun StorageTek 5800 system nodes do not contain any more than one fragment of a file. Also, a node failure or even two node failures are unlikely to result in a loss of data because the fragments are distributed across nodes.

The objective here is to encode and distribute data and parity fragments in such a manner that the loss of a disk or even multiple disks does not result in a loss of data. Thus, if you lose a disk, the system regenerates the lost data elsewhere, and when you replace disks, they are automatically populated with data through background healing tasks. See the section that follows for more information.

Node Failure and Recovery

FIGURE 1-10 shows a graphical representation of a node failure. It first shows a failure in node 4 (X) and then displays the fragments on that failed node being relocated to different disks in different nodes. Specifically, the parity fragment of Object 1 is relocated to node 13, disk 1, while the data fragment of Object 2 is relocated to node 12, disk 3.

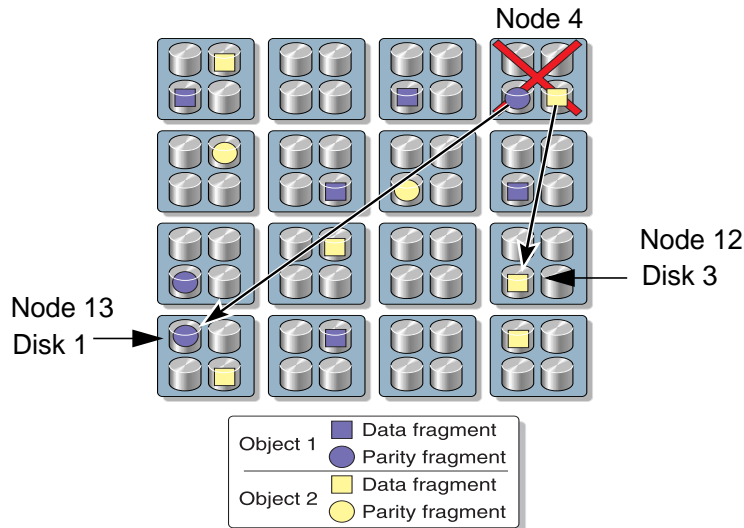


FIGURE 1-10 Node Failure and Fragment Relocation Example

As shown in **FIGURE 1-10**, the recovery process restores reliability by re-creating fragments on the failed node to different nodes in the system. Once the recovery process is complete, the system is again at full reliability. Thus, the urgency to replace the failed node is reduced, allowing for deferred maintenance.

A design that allows for deferred maintenance is a major advantage the Sun StorageTek 5800 system possesses over more traditional storage systems. When a Sun StorageTek 5800 system cell heals itself following the failure of a disk or server component, lost fragments are fully and reliably recovered without the use of dedicated, hot-spare components. As long as spare capacity remains in the system, recovery can occur over and over again without requiring the replacement of failed components. Indeed, for normal component failures, the Sun StorageTek 5800 system is intended to go four to six months without servicing.

Customizing the Configuration

This chapter describes the use of the `netcfg` command when configuring system parameters. It contains the following sections:

- [“Setting Network Configuration Parameters” on page 15](#)
- [“Configuring the Alert Subsystem” on page 17](#)
- [“Setting External Virtual IP Addresses” on page 17](#)

Setting Network Configuration Parameters

You use the `netcfg` command to set various network configuration parameters, including:

- An SMTP server
- A network gateway
- External NTP servers
- An external log host
- Authorized clients

The subsections that follow explain how to set these network configuration parameters.

Configuring an SMTP Server

- Type `netcfg smtp` followed by an IP address and port number to set the location of the SMTP server.

```
hc $ netcfg smtp 10.7.224.15 25
```

Note that the SMTP port number is not a required entry.

Configuring a Gateway

- Type `netcfg gw` followed by an IP address and subnet mask to set the default gateway (router) that the Sun StorageTek 5800 system uses.

```
hc $ netcfg gw 10.7.227.254 255.255.255.0
```

It is required that you specify a default gateway.

Configuring External NTP Servers

- Type `netcfg ntp` followed by the IP addresses to configure the external NTP servers that the Sun StorageTek 5800 system uses.

```
hc $ netcfg ntp 10.7.224.30,10.7.224.40
```

It is required that you specify external NTP servers. The Sun StorageTek 5800 system requires a time server be configured.

Configuring an External Log Host

- Type `netcfg loghost` followed by an IP address to set an external log host for the Sun StorageTek 5800 system.

```
hc $ netcfg loghost 10.7.224.15
```


Configuring Authorized Clients

1. Enter `netcfg clients` followed by the IP addresses to configure authorized clients.

```
hc $ netcfg clients 10.45.1.19,192.18.7.2,34.7.7.101
```

Note that specifying authorized clients is optional. If you do not set this property, data access to the system is granted to any client on the network.

2. Enter `netcfg clients all` to reset this property and allow all clients to enter the system.

```
hc $ netcfg clients all
```

Configuring the Alert Subsystem

You use the `alertcfg` command to configure the alert subsystem. You can set the external log and email alert hosts, as well as any other alert parameters, such as email addresses for email alerts. Should you enter `alertcfg` with no parameters, the current values will display.

- Type `alertcfg add to` or `alertcfg del cc` to add or delete an email address from the specified list type.

```
hc $ alertcfg add to alerts@samplecompany.com  
hc $ alertcfg del cc admin@samplecompany.com
```

Setting External Virtual IP Addresses

Use the `ifconfig` command to set the external virtual IP addresses. All network and node addresses that are internal to the cluster are private and not configurable. The cluster has two public host addresses, one for the data VIP and one for the administrative VIP.

1. **Assign a data VIP with `ifconfig` data.**

```
hc $ ifconfig data0 10.7.224.42
```

2. **Assign an administrative VIP with `ifconfig` admin.**

```
hc $ ifconfig admin0 10.7.224.41
```

Use data VIP for API access, and the administrative VIP to access the CLI.

Monitoring the System

This chapter describes the Sun StorageTek 5800 system monitoring capabilities. It contains the following sections:

- [“Obtaining System Version and Status” on page 19](#)
- [“Viewing the System Software Version” on page 20](#)
- [“Obtaining FRU Listings” on page 21](#)
- [“Getting Disk Status” on page 22](#)
- [“Obtaining SMTP Email Alerts” on page 23](#)

Obtaining System Version and Status

1. Obtain basic cluster state information with the `sysstat` command.

This command provides an estimate of free space in the cluster that is available for data storage. For a detailed breakdown of space usage per disk, refer to the `df` command described in [“Getting Disk Status” on page 22](#).

```
hc $ sysstat
Silo 1, Cell 1: Cluster Online. Estimated free space: 7T.
16 nodes online, 64 disks online, 0 disks unrecovered
Data VIP 10.7.225.222. Admin VIP 10.7.225.223.
Data services Online, Query Available
Failure Tolerance: Fault Tolerant
Last Recovery Cycle: Completed 5 hours ago.
Next Recovery Cycle: To complete in 9 hours.
```

The output that the `sysstat` command produces is as follows:

- Data services Online means that the system is available to read and write to, while Data services Offline means that the system is not available.
- Query Available means it is possible to query and add metadata, while Query Unavailable means it is not possible.
- Failure Tolerance- Fault Tolerant means the system is operational and able to withstand failures. Failure Tolerance- Degraded means there is one more failure before the system loses data, while Failure Tolerance- Not Tolerant indicates that there have been too many failures, or the cluster is not currently online.

2. Obtain extended cluster state information with the `-v` (verbose) option of the `sysstat` command.

Verbose output includes the online/offline status of each node and disk in the cluster. The online/offline status reported by the command refers to the logical system status. To see the state of hardware components, refer to the `hwstat` command described in [“Obtaining FRU Listings” on page 21](#).

```
hc $ sysstat -v
NODE-101      [ ONLINE ]
DISK-101:0    [ ONLINE ]
DISK-101:1    [ OFFLINE ]
DISK-101:2    [ ONLINE ]
DISK-101:3    [ ONLINE ]
NODE-102      [ ONLINE ]
DISK-102:0    [ ONLINE ]
DISK-102:1    [ ONLINE ]
DISK-102:2    [ ONLINE ]
DISK-102:3    [ ONLINE ]
NODE-103      [ ONLINE ]
DISK-103:0    [ ONLINE ]
.
.
.
```

Viewing the System Software Version

- Display the version of the system software using the `version` command.

```
hc $ version
Sun StorageTek 5800 System release [1.0 -8246]
```

Obtaining FRU Listings

1. Obtain a list of field-replaceable units (FRUs) with the `hwstat` command.

```
hc $ hwstat
NODE-101          [00E08159-7311-0000-CA80-11735981E000]
DISK-101:0        [HITACHI_HDS7250ASUN500G_0609K4MJHD=KRVN63ZAH4MJHD]
DISK-101:1        [HITACHI_HDS7250ASUN500G_0609K4NHRD=KRVN63ZAH4NHRD]
DISK-101:2        [HITACHI_HDS7250ASUN500G_0609K4USGD=KRVN63ZAH4USGD]
DISK-101:3        [HITACHI_HDS7250ASUN500G_0609K54SLD=KRVN63ZAH54SLD]
NODE-102          [00E08159-59FC-0000-7480-FC595981E000]
DISK-102:0        [HITACHI_HDS7250ASUN500G_0609K56DKD=KRVN63ZAH56DKD]
DISK-102:1        [HITACHI_HDS7250ASUN500G_0609K50VPD=KRVN63ZAH50VPD]
DISK-102:2        [HITACHI_HDS7250ASUN500G_0609K3P98D=KRVN63ZAH3P98D]
DISK-102:3        [HITACHI_HDS7250ASUN500G_0609K4UT8D=KRVN63ZAH4UT8D]
.
.
```

2. Obtain a list of all components (FRUs and others) and their statistics with the `-v` (verbose) option of the `hwstat` command.

```
hc $ hwstat -v
Component          Type      FRU ID                                     Status
-----
NODE-101           NODE      [00E08159-7311-0000-CA80-11735981E000]  [Online]
DISK-101:0         DISK      [HITACHI_HDS7250ASUN500G_0609K4MJHD=
[enabled]          KRVN63ZAH4MJHD]
DISK-101:1         DISK      [HITACHI_HDS7250ASUN500G_0609K4NHRD=
[enabled]          KRVN63ZAH4NHRD]
DISK-101:2         DISK      [HITACHI_HDS7250ASUN500G_0609K4USGD=
[enabled]          KRVN63ZAH4USGD]
DISK-101:3         DISK      [HITACHI_HDS7250ASUN500G_0609K54SLD=
[enabled]          KRVN63ZAH54SLD]
```

NODE-102	NODE	[00E08159-59FC-0000-7480-FC595981E00]	[Online]
DISK-102:0 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K56DKD=KRVN63ZAH56DKD]	
DISK-102:1 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K50VPD=KRVN63ZAH50VPD]	
DISK-102:2 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K3P98D=KRVN63ZAH3P98D]	
DISK-102:3 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K4UT8D=KRVN63ZAH4UT8D]	
.			
.			

Getting Disk Status

1. Obtain a summary of disk usage with the `df` command.

The displayed numbers refer to the logical space used by, or available for, the user's data storage. Reserved space is reserved by the system for data recovery operations, and is not available to the user. Total space is a sum of Available + Used + Reserved. The use percentage is calculated as Used / (Used + Available).

```
hc $ df
Logical space: Total: 18T; Available: 5T; Used: 9T; Reserved: 4T
```

2. Obtain information on the physical space available on the disk with the `-p` option.

```
hc $ df -p
DISK-101:0 Total: 449,128,448; Avail: 361,584,640; Used: 87,543,808; Use%: 19%
DISK-101:1 Total: 449,128,448; Avail: 388,771,840; Used: 60,356,608; Use%: 13%
DISK-101:2 Total: 449,128,448; Avail: 385,692,672; Used: 63,435,776; Use%: 14%
DISK-101:3 Total: 449,128,448; Avail: 388,822,720; Used: 63,305,728; Use%: 14%
```

Obtaining SMTP Email Alerts

If you set up SMTP email alerts, the Sun StorageTek 5800 system delivers email alerts for the following conditions:

- Enabling or disabling of a disk or server node through the CLI
- Cluster reboot
- Cluster shutdown
- Administrator password or public key change
- Server node joining or leaving the cluster
- Switch failover

To configure the Sun StorageTek 5800 system to send email notifications, you must specify an SMTP server and the email address of the recipient.

1. Specify an SMTP server that the Sun StorageTek 5800 system will use.

```
hc $ netcfg smtp 10.7.224.20
```

2. Specify an email recipient.

```
hc $ alertcfg add to alerts@example.com
```

Optionally, you may also specify a cc email address, for example:

```
hc $ alertcfg add cc admin@example.com
```


Configuring Metadata and File System Views

This chapter explains metadata and the system schema, discusses upgrading the schema configuration, and then describes file system views and their definition. It contains the following sections:

- [“Understanding Metadata and the System Schema” on page 25](#)
- [“Upgrading the Schema Configuration” on page 32](#)
- [“Understanding File System Principles” on page 33](#)

Understanding Metadata and the System Schema

Metadata is extra information about the data object. Each object stored in the Sun StorageTek 5800 system can have one or more descriptions attached to it. This description can contain any kind of information on the data.

Metadata can come from one of three sources:

- The data itself. For example, metadata might involve automatically extracting ID3 tags from an MP3 file, or extracting appropriate strings to later perform a text search.

Note – Automated metadata extraction is not supported in release v1.0 of the Sun StorageTek 5800 system.

- Information explicitly provided at the time an object is stored. For example, metadata might include the patient name, social security number, or body part that you enter when storing a medical image. (This is not necessarily derived from the data).
- Metadata created by the Sun StorageTek 5800 system storage system itself. Such attributes are not writable by client applications and are created by the Sun StorageTek 5800 system.

The Sun StorageTek 5800 system's metadata space is partitioned in namespaces (that is, a string followed by a period) for greater naming flexibility. Thus, the system reserves the system namespace for storing the metadata created by the Sun StorageTek 5800 system itself.

The system namespace includes a unique identifier for each stored object, called the object ID or OID. Namespaces are collections of names, identified by a uniform resource identifier (URI), that are used in XML documents as element types and attribute names. Using namespaces is completely optional for user metadata. However, the Sun StorageTek 5800 system reserves the `system` and `filesystem` namespaces.

Specifically, the `filesystem` namespace affects how the file system layer presents files. For example, it may include the user identifier (UID) and group identifier (GID). For more information on namespaces, see [“Metadata Configuration and the XML File Structure” on page 28](#), while [“Understanding File System Principles” on page 33](#) provides more detail on file systems.

Relational Metadata

The only supported metadata format in the Sun StorageTek 5800 system release 1.0 is a set of typed name-value pairs. Since the values are typed, the Sun StorageTek 5800 system needs to be pre-configured with names and their value type.

TABLE 4-1 lists the supported metadata types for version 1 of the Sun StorageTek 5800 system.

TABLE 4-1 Valid Metadata Types

Valid Types	Description
Long	64 bits Maximum Value: -9223372036854775808 Minimum Value: 9223372036854775807
Double	64 bits Maximum Value: 1.7976931348623157E308d Minimum Positive Value: 4.9E-324d
String	512 characters

Note that when performing a storage operation, you always store some metadata. If no name-value pairs are specified, the metadata only contains the fields populated by the Sun StorageTek 5800 system, such as creation time (ctime), data length, data hash etc. The returned OID provides a reference to both the data you just stored and its metadata.

You can then use an API function call (addMetadata) to attach an extra metadata description to an existing piece of data. As shown in FIGURE 4-1, this operation returns a new OID that designates the newly-stored metadata, as well as the data that was already in the system.

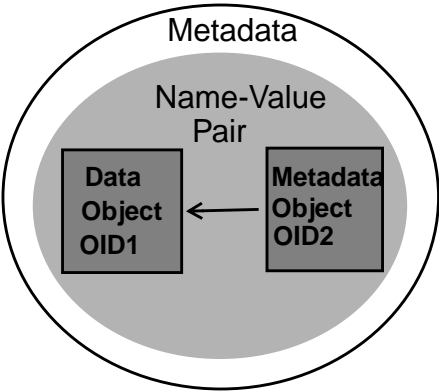


FIGURE 4-1 The addMetadata Operation

Metadata Structure and the Schema

The Sun StorageTek 5800 system uses an embedded database to perform its search functions.(This database is also referred to as the metadata index.) In order to make this index as efficient as possible, the metadata attributes and their types are predefined in a schema.

The schema specifies what the Sun StorageTek 5800 system metadata permits and how it is structured. It consists of the list of expected fields, their types, and a designation of whether or not they should be included in the index. A predefined schema is necessary to create internal indices of the actual metadata. In the Sun StorageTek 5800 system, the predefined schema contains the minimum set of attributes (that is, only the `system` and `filesystem` namespace).

Metadata Configuration and the XML File Structure

All metadata-specific configuration for the Sun StorageTek 5800 system is stored in an XML file. XML offers a widely adopted, standard way of representing text and data in a format that can be processed with relatively little human intervention and exchanged across diverse hardware, operating systems and applications.

XML is designed to describe information and its composition, while the HyperText Markup Language (HTML) is designed to display information. The tags you use to mark up HTML documents and the document's structure are predefined, so that you can only use tags that are defined in the HTML standard. By contrast, XML is extensible and allows you to define your own tags and your own document structure.

The metadata-specific configuration file for the Sun StorageTek 5800 system uses a standard XML file format and is structured as follows:

```
<?xml version="1.0" encoding="UTF-8"?>

<metadataConfig>

    <schema>
        Schema definition
    </schema>

    <fsViews>
        File system views specification
    </fsViews>

</metadataConfig>
```

Note – Only one <schema> and one <fsViews> subsection are permitted here.

The general schema subsection format is then organized as follows:

```
<schema>
  <namespace name= "namespace" [ writable="true" ] [ extensible="true" ] >
    <namespace name= "subnamespace">
      <field name= "fieldName" type= "fieldType" [ indexable = "true" ] />

    </namespace>
  <namespace>
    ...
</schema>
```

The namespace referenced here is a collection of names, identified by a URI, that are used in XML documents as element types and attribute names. Namespaces exist simply to keep names from separate sources from colliding unintentionally. Note that you can have as many namespaces as desired. There is also no limitation in the number of namespaces that can be encapsulated within a given namespace level.

When defining a namespace, you can define two optional properties:

- Writable

A writable namespace applies to the filesystem interface using WebDAV. Specifically, it means that you can specify any field in that domain at the time an object is stored. Note that once a domain is not writable, subdomains cannot overwrite the value and be writable. The default value is true.

- Extensible

You can extend an extensible domain in a subsequent configuration update, thus adding more attributes or more subdomains to an existing domain. Once a domain is not extensible, subdomains cannot overwrite the value and become extensible. The default value is true.

Fully-Qualified Names

When using an attribute name in an application, such as when storing metadata or querying, the form of the attribute name that is used is always the fully-qualified name. The fully-qualified name consists of all the enclosed namespace names from the broadest to the narrowest, separated by dots, followed by the attribute name itself. For example, the fully-qualified name of the attribute in the general schema subsection format shown previously is `namespace.subnamespace.fieldName`.

DNS Namespaces

Namespaces are arbitrary. However, as a convention it is recommended that you ensure that the namespaces relate directly to Domain Name Service (DNS) names, in a way similar to how Java classes are named.

DNS is a hierarchical naming system provided for computer networks. It is broken up hierarchically into domains and subdomains, which are logical organizations of computers that exist in a larger network. These domains exist at different levels and connect in a hierarchy that resembles the root structure of a tree.

Domain names are assigned by a well-known process through the Internet Assigned Numbers Authority (IANA), in conjunction with naming authorities within each commercial or government entity. As a result, DNS names from different organizations are already guaranteed not to conflict. This is the only quality of DNS names the Sun StorageTek 5800 system is using.

For example, it is recommended that an organization whose DNS name is `sample.company.com` should create a set of Sun StorageTek5800 system namespaces under the namespace heading `com.company.sample`. If you follow this convention, you can guarantee that storage items from separate sources that are combined into a single Sun StorageTek 5800 system cell will not conflict in their

metadata namespaces, even if the applications were not originally designed to work together. Note that this is the same convention by which Java class names are assigned.

Reserved Domains

[TABLE 4-2](#) lists the domains that Sun StorageTek 5800 system reserves.

TABLE 4-2 Sun StorageTek 5800 System Reserved Domains

Name	Writable	Extensible
system	false	false
filesystem	true	false
filesystem.ro	false	false

[TABLE 4-3](#) lists the content of the reserved system domain.

TABLE 4-3 Sun StorageTek 5800 System Reserved Domains

Attribute Name	Definition
system.object_id	The object identifier
system.object_ctime	Creation time
system.object_layoutMapId	Layout map used to store the object
system.object_size	Data size
system.object_hash	Hash value for the data
system.object_hash_alg	Algorithm used to compute the hash (for example, SHA1)

Schema Definition

The purpose of a Document Type Definition (DTD) is to define the legal building blocks of an XML document. The DTD defines the document structure with a list of legal elements, thus providing an application-independent way of sharing data.

For example, independent groups of people can agree to use a common DTD for interchanging data. The DTD allows you to check the logical structure of the data that you receive from the outside world and also verify your own data.

In the Sun StorageTek 5800 system, you define a schema using the schema DTD. The XML syntax you need to follow here is as follows:

```

<!DOCTYPE HCMETADATACONFIG [

<!ELEMENT metadataConfig (schema, fsViews)>

<!ELEMENT schema (namespace*, field*)>
<!ELEMENT namespace (namespace*, field*)>
<!ELEMENT field EMPTY>

<!ELEMENT fsViews (fsView*)>
<!ELEMENT fsView (attribute+)>

<!--
  namespace attributes
-->
<!-- namespace name -->
<!-- namespace writable -->
<!-- namespace indexable -->

<!--
  field attributes
-->
<!-- field name -->
<!-- field type -->
<!-- field indexable -->

<!--
  fsView attributes
-->
<!-- fsView name -->
<!-- fsView filename -->
<!-- fsView namespace -->

<!--
  attribute attributes
-->
<!-- attribute name -->
<!-- attribute unset -->

]>

```

Note that the only schema changes you can make in the Sun StorageTek 5800 system are:

- Adding new attributes in an existing namespace (if the namespace is extensible)
- Creating a new namespace (if the parent namespace is extensible)
- Creating new file system views

Upgrading the Schema Configuration

To upgrade the schema configuration:

1. **Create a schema overlay to extend an existing schema.**

A schema overlay is an XML file that follows the specification shown in [“Metadata Configuration and the XML File Structure” on page 28](#). It contains only the new namespaces and fields that you wish to add.

Note – For version 1 of the Sun StorageTek 5800 system, you must edit the XML file manually.

If you wish, you can use `mdconfig` followed by the `-t (--template)` option. This returns an XML template file that you can use as a starting point to create that overlay.

Once a version of the overlay is available, you can perform a validation through the CLI. The purpose of the validation is to ensure that the XML syntax is correct and also to provide an overview of the operation that will be performed if the overlay occurs.

2. Use the `mdconfig` command with no arguments to perform a validation.

For example, to validate the local `overlay.xml` file, type the following:

```
hc $ cat overlay.xml | ssh admin@<ADMIN IP> mdconfig
```

Once you are satisfied with the overlay, you must commit it so the Sun StorageTek 5800 system can go ahead and execute it.

3. Use `mdconfig` followed by the `-c (--commit)` option to commit the `overlay.xml` file.

For example, to continue the previous example, enter:

```
hc $ cat overlay.xml | ssh admin@<ADMIN IP> mdconfig -c
```

Note – The commit option first runs a validation before performing the commit operation. If the XML syntax is not correct, an error is produced.

Understanding File System Principles

The Sun StorageTek 5800 system’s file system is built on top of the metadata. Thus, a file system, or virtual view, specifies the metadata attributes you wish to use to browse the Sun StorageTek 5800 system.

Basically, a view transforms the simple collection of name-value pairs to a path (that is, a hierarchical directory structure). For the version 1.0 release, the file system view is accessible only through Web-based Distributed Authoring and Versioning (WebDAV).

Using WebDAV for File Browsing

WebDAV is a set of extensions to the HTTP protocol that enable you to read and delete files on remote web servers. Specifically, DAV is an extension of the HTTP/1.1 protocol that adds new HTTP methods and headers.

Each WebDAV request initiates a separate WebDAV operation. GUI and CLI clients then enable you to browse WebDAV collections as you would on any web site. Thus, you can use standard web browsers, like Internet Explorer, graphical browsers, such as Konqueror, Nautilus (UNIX), or Finder (OS X), or a CLI client, such as Cadaver (UNIX), to browse WebDAV collections. You go to the top-level *webdav* directory to see the *fsviews*.

For example, you can define the following sample schema for MP3s on a cluster:

```
<fsView name= "byArtist" namespace="mp3" filename="${title}.${type}">
<attribute name="artist"/>
<attribute name="album"/>
</fsView>
```

Note here that each:

- *fsView* entry specifies a new file system abstraction.
- Within the *fsView* attributes:
 - *name* is the top-level directory name of the abstraction.
 - *namespace* defines a schema namespace prefix for all of the names used (that is, *title* will be parsed as *mp3.title*).
 - *filename* defines the form of the files (leaves) that are exposed by the file system in that view.
- In addition to the attribute list, each *fsView* contains an ordered list of attributes that defines the view.

The following path:

```
/webdav/byArtist/Charlie Parker/Bird and Diz/Bloomdido.ogg
```

represents the query

```
mp3.artist='Charlie Parker' & mp3.album='Bird and Diz' & mp3.title=
'Bloomdido' & mp3.type='ogg'
```

In the browser’s address bar, you type the following to display a page of album names:

```
webdav://dev321-data:8080/webdav/byArtist/U2
```

You can then click a particular album name, and a web page with album tracks appears as with any ordinary web page.

Metadata Attributes and WebDAV Properties

Each file in the Sun StorageTek 5800 system archive appears as a file in the exported file system. The file attributes (stat data) are exported as WebDAV properties.

[TABLE 4-4](#) lists the WebDAV property names and user metadata attributes. Be aware that these attributes are regular metadata values accessible through API queries.

TABLE 4-4 WebDAV Property Names and User Metadata Attributes

	WebDAV Property	Metadata Attribute	Description
Pre-defined properties	DAV:getlastmodified	filesystem.mtime	Last modification time
	DAV:getcontentlength	system.object_size	Size of file
	DAV:creationdate	system.object_ctime	File creation time
	DAV:getcontenttype	filesystem.mimetype	MIME type
	DAV:displayname	<filename>	Name presented to user
New properties	HCFS:mode	filesystem.mode	File mode (permissions etc.)
	HCFS:uid	filesystem.uid	Owner ID
	HCFS:gid	filesystem.gid	Group ID

Note – The timestamps are all 64-bit signed offsets from the epoch -- 00:00:00 1/1/1970 Coordinated Universal Time (UTC), in milliseconds, while the range is 300 million years.)

The file size, uid and gid are numeric, unsigned 64-bit integers, while the creationdate property is returned as an ISO 8601 localized string. The format for getlastmodified is not defined by the protocol, so a value similar to that of date(1) is returned.

Virtual View Definition

A view presents stored data and metadata as a virtual, read-only file system. For example, for an MP3 file system view, you may present a view with a directory for the artist, a subdirectory for the album, and file names based on the title of the music files.

File system view declarations are useful for the file system protocol layers, such as WebDAV, to know what to export. To create a new virtual view, you add an entry in the `<fsviews>` section of the XML overlay file shown in [“Metadata Configuration and the XML File Structure” on page 28](#).

Administering the System

This chapter describes how to perform administrative tasks on your Sun StorageTek 5800 system. It contains the following sections:

- [“Managing Passwords or Public Keys” on page 37](#)
- [“Executing a Script” on page 39](#)
- [“Powering Up the System” on page 40](#)
- [“Shutting Down the System” on page 40](#)
- [“Rebooting the System” on page 41](#)
- [“Recovering From a Power Failure” on page 41](#)
- [“Upgrading the System” on page 41](#)
- [“Wiping Data From the System” on page 43](#)

Managing Passwords or Public Keys

You can use the `passwd` command interactively to set a new password, or use it non-interactively to supply a new public key.

Changing the System Password

- **Log in to the CLI and change the password interactively with the `passwd` command.**

```
hc $ passwd
passwd
password: password (again):
authentication tokens successfully updated
hc $
```

Note – Passwords must be at least six characters long.

Configuring a Public Key

A public key allows you to log in to the Sun StorageTek 5800 system from remote systems also carrying that key. Configuring a public key is primarily done to allow for scripting.

Create a public key without a pass phrase. The Sun StorageTek 5800 system comes configured with a password, so the first time you set the public key, you will be prompted to supply your password.

Once the key is set, you can log in without being prompted. If you send another key non-interactively, the new key replaces the old key. If you wish to return to interactive logins, remove your private key, or send a public key that you have created with a pass phrase.

1. **Configure `ssh` for password-free login by supplying a public key from a client system, for example:**

```
client_$ ssh admin@hc1-admin passwd --pubkey < file.pub
admin@hc1-admin's password:
authentication tokens successfully updated
```

where *file.pub* is the file containing the public key.

2. Verify password-free login.

```
client_$ ssh admin@hc1-admin
Sun StorageTek 5800 System Management Console
hc $
```

Executing a Script

The following simple example demonstrates how to use a CLI command within a client script.

1. Create a script file containing CLI commands.

You create scripts on ssh-capable client machines in any desired client-side shell, and then execute them across the network via `ssh`. In the examples that follow, note that the identity of the user on the client machine is the same as *admin* on the Sun StorageTek 5800 system. Otherwise, `ssh` will fail.

On the client-side, use any available editor to create the following script file.

```
#!/bin/sh
ssh admin@hc1-admin df
```

Substitute the name or VIP of the administrative interface of your cluster for `hc1-admin` in the example.

2. Save the file (for example, as `cli-script.sh`) and make it executable by typing:

```
client_$ chmod +x cli-script.sh
```

3. Run the script.

```
client_$ ./cli-script.sh
```

You should see the `df` output for the Sun StorageTek 5800 system cluster appear on your screen.

You can enter additional CLI commands with multiple `ssh` commands embedded in standard script logic to script CLI command activities.

Powering Up the System

1. Verify that the system is completely shut down by ensuring that the power switches on the front of the rack are set to the off or 0 position.
2. Switch both black power switches on the front of the rack to the on or 1 position.
3. Wait several minutes.
4. Log in to the CLI and verify that the Sun StorageTek 5800 system is operational using the `hwstat` command.

For more information, see [“hwstat” on page 51](#).

Shutting Down the System

1. Shut down (power off) the Sun StorageTek 5800 system with the `shutdown` command.

```
hc $ shutdown
shutdown? [y/N]: n
hc $ shutdown
shutdown? [y/N]: y
Connection to hcl-admin closed.
```

2. Switch both power switches on the front of the rack to the off or 0 position.

Rebooting the System

- **Reboot the Sun StorageTek 5800 system with the `reboot` command.**

```
hc $ reboot
Reboot? [y/N]: n
hc $ reboot
Reboot? [y/N]: y
Connection to hcl-admin closed.
hc $ reboot --force
Connection to hcl-admin closed.
```

Recovering From a Power Failure

When power is restored after a power failure, the Sun StorageTek 5800 system becomes operational automatically without administrator intervention. The system should not require any recovery time and there are no procedures for recovery after a power loss, other than restoring power.

While no special procedures are required here, you should still:

1. **Verify that all nodes and disks are online.**
2. **If some are not, check for potential system/hardware failures caused by the abrupt loss of power.**

No data loss should occur as a result of a power failure. Any stored data for which the client received an OID remains securely stored on the Sun StorageTek 5800 system. As you would expect, however, any client store operations that were in-progress at the time of power failure will fail.

Upgrading the System

You use the `upgrade` command to upgrade the Sun StorageTek 5800 system cluster to a newer version of the software. You perform an upgrade:

- From a DVD placed in the service node

- By downloading the image from an IP address and upgrading in one step
- By downloading the image from an IP address and upgrading at a later time
- **To perform an upgrade from a DVD, insert the DVD into the drive in the service node. Then, log into the CLI and type the following:**

```
hc $ upgrade dvd
```

- **To download the image from an IP address and upgrade in one step, type the following:**

```
hc $ upgrade http://123.45.67.89/honeycombdvd.iso
```

- **To download the image from an IP address and upgrade at a later time, type the following:**

```
hc $ upgrade download http://123.45.67.89/honeycombdvd.iso  
hc $ upgrade downloaded
```

Note here that:

- You cannot stop the upgrade with the intention of resuming it later.
- You also cannot remove or add a node during the upgrade.

Once the upgrade begins, it checks to see that all nodes and disks are available and then does not permit any changes. Be aware that the process is non-interactive. However, it does display a great deal of information about what is going on throughout. For instance, it unpacks the install image, upgrades the service node, the switches, and finally each node in the cluster.

If there is a problem with the images during an upgrade, the nodes will boot from the previous version. There is no procedure for manually rolling back to that previous version.

If there is a power outage during an upgrade, the system may or may not come back online. It depends upon where in the process, the upgrade was interrupted. Should the system not become operational, you should call your Sun Service representative.

A controlled reboot of the entire cell is the last phase of the upgrade process. First, the service node is rebooted. Next, each node except one is rebooted. Then, the switches are rebooted, and finally, the last node is rebooted. When the switch is rebooted, you are logged out of the CLI and are unable to connect again until the system has come back with the new image.

Wiping Data From the System

- Delete all data and metadata from the cluster with the `wipe` command.

```
hc $ wipe  
Destroy all data? [y/N]:
```

After you issue the `wipe` command, the schema is reset to the original factory settings, while other cluster configurations (such as network settings and passwords) are unaffected.

CLI Command Reference

The CLI allows you to see system status, configure system properties, and administer the hardware. It is also scriptable using the `ssh` keys as explained in [“Configuring a Public Key” on page 38](#).

The following pages provide a description, syntax, and examples for each CLI command. Text that you enter literally is shown in **bold**. Optional parameters are shown enclosed in [square brackets]. Variables for which you must substitute values are shown in *italics*. When you must select between multiple parameters, options are separated by a bar (|).

This appendix contains the following commands:

- [“alertcfg” on page 46](#)
- [“df” on page 47](#)
- [“help” on page 48](#)
- [“hwcfg” on page 49](#)
- [“hwstat” on page 51](#)
- [“ifconfig” on page 54](#)
- [“license” on page 55](#)
- [“logout” on page 56](#)
- [“mdconfig” on page 57](#)
- [“netcfg” on page 59](#)
- [“passwd” on page 61](#)
- [“reboot” on page 63](#)
- [“shutdown” on page 64](#)
- [“sysstat” on page 65](#)
- [“upgrade” on page 67](#)
- [“version” on page 68](#)
- [“wipe” on page 69](#)

alertcfg

Description

Adds or removes email addresses to which alerts will be sent.

Entering `alertcfg` with no parameters displays the current values of each.

Syntax

`alertcfg [add [to | cc] | del [to | cc]]`

Options

- `add to|cc`
Adds or carbon copies (cc) an email address to the specified list type.
- `del to|cc`
Removes an email address from the specified list type.

Examples

```
hc $ alertcfg add to fred@samplecompany.com
hc $ alertcfg del cc admin@samplecompany.com
```

df

Description

Reports estimated disk space usage on the system.

Syntax

`df [-h | --human-readable | -p | --physical]`

Options

- `-h, --human-readable`
Print the results in easily-readable summary format. Note that rounding of numbers means totals are approximate.
- `-p, --physical`
Prints the physical space available on the disk.

Examples

```
hc $ df
Total: 25,151,193.0881; Available: 24,895,438.848; Used: 255,754,240; Use: 1%
hc $ df -p
DISK-101:0 Total: 449,128,448; Avail: 361,584,640; Used: 87,543,808; Use%: 19%
DISK-101:1 Total: 449,128,448; Avail: 388,771,840; Used: 60,356,608; Use%: 13%
DISK-101:2 Total: 449,128,448; Avail: 385,692,672; Used: 63,435,776; Use%: 14%
DISK-101:3 Total: 449,128,448; Avail: 388,822,720; Used: 63,305,728; Use%: 14%
.
.
.
```

help

Description

Displays a list of available CLI commands.

Syntax

`help [command-name]`

Options

command-name

Specifies the command for which you wish to receive detailed help.

Examples

```
hc $ help
alertcfg  df          help          hwcfg      hwstat
ifconfig  license     logout      mdconfig   netcfg
passwd    reboot      shutdown   sysstat    upgrade
version   wipe
hc $
```

hwcfg

Description

Administers system hardware (nodes and disks).

Syntax

`hwcfg FRU ID [-E | --enable]`

`hwcfg FRU ID [-D | --disable]`

Options

- `-E` or `--enable`
Enables a disk identified by *FRU ID* (field-replaceable unit ID).
- `-D` or `--disable`
Disables a disk identified by *FRU ID*.

Examples

1. Disable a disk.

```
hc $ hwcfg DISK-101:1 --disable
Disabling [101:1]
Operation successfully completed
```

Disk status in `hwstat -v` output changes from `ONLINE` to `OFFLINE`.

2. Enable a disk.

```
hc $ hwcfg DISK-101:1 --enable
Enabling [101:1]
Operation successfully completed
```

Disk status in `hwstat -v` output changes from `OFFLINE` to `ONLINE`.

Note – If you reboot with the disk disabled, to enable the disk you must enable it and then reboot again.

hwstat

Description

Displays information about disk and server node field-replaceable units (FRUs) in the Sun StorageTek 5800 system or displays information about a particular FRU.

Syntax

```
hwstat [ -v | --verbose ]
```

Options

`-v, --verbose`

Shows a list of all components (FRUs and others) and their statistics.

Examples

```
hc $ hwstat
```

```

NODE-101      [00E08159-7311-0000-CA80-11735981E000]
DISK-101:0    [HITACHI_HDS7250ASUN500G_0609K4MJHD=KRVN63ZAH4MJHD]
DISK-101:1    [HITACHI_HDS7250ASUN500G_0609K4NHRD=KRVN63ZAH4NHRD]
DISK-101:2    [HITACHI_HDS7250ASUN500G_0609K4USGD=KRVN63ZAH4USGD]
DISK-101:3    [HITACHI_HDS7250ASUN500G_0609K54SLD=KRVN63ZAH54SLD]
NODE-102      [00E08159-59FC-0000-7480-FC595981E00]
DISK-102:0    [HITACHI_HDS7250ASUN500G_0609K56DKD=KRVN63ZAH56DKD]
DISK-102:1    [HITACHI_HDS7250ASUN500G_0609K50VPD=KRVN63ZAH50VPD]
DISK-102:2    [HITACHI_HDS7250ASUN500G_0609K3P98D=KRVN63ZAH3P98D]
DISK-102:3    [HITACHI_HDS7250ASUN500G_0609K4UT8D=KRVN63ZAH4UT8D]
.
.
```

```
hc $ hwstat -v
```

Component -----	Type ----	FRU ID -----	Status -----
NODE-101	NODE	[00E08159-7311-0000-CA80-11735981E000]	[Online]
DISK-101:0 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K4MJHD= KRVN63ZAH4MJHD]	
DISK-101:1 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K4NHRD= KRVN63ZAH4NHRD]	
DISK-101:2 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K4USGD= KRVN63ZAH4USGD]	
DISK-101:3 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K54SLD= KRVN63ZAH54SLD]	
NODE-102	NODE	[00E08159-59FC-0000-7480-FC595981E000]	[Online]
DISK-102:0 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K56DKD= KRVN63ZAH56DKD]	

DISK-102:1 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K50VPD= KRVN63ZAH50VPD]
DISK-102:2 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K3P98D= KRVN63ZAH3P98D]
DISK-102:3 [enabled]	DISK	[HITACHI_HDS7250ASUN500G_0609K4UT8D= KRVN63ZAH4UT8D]
.		
.		

ifconfig

Description

Configures or displays the admin and data virtual IP addresses.

Note – Release 1.0 supports changing only the IP address and netmask of the data and admin VIPs. A VIP is either data0 or admin0.

Syntax

ifconfig device netmask

Options

device

Specifies the device (data0 or admin0).

netmask

Specifies the IP address.

Examples

```
hc $ ifconfig data0 10.7.224.42
Please wait a couple of minutes while the configuration is
updated.
nopasswd@10.123.45.1's password:
You must reboot the cell with "reboot --all" for changes to take
effect.
hc $ ifconfig admin0 10.7.224.41
You must reboot the cell for changes to take effect.
```

license

Description

Displays or sets the current software license.

Entering `license` with no parameters displays the current license.

Syntax

`license` *[value]*

Options

value

Sets the current license to the specified value.

Examples

```
hc $ license
License Not Yet Entered
hc $ license STK5800-SEARCH-10001
hc $ license
STK5800-SEARCH-10001
hc $
```

logout

Description

Ends the administrative session.

Syntax

logout | exit

Options

None

Examples

```
client $ logout
hc $ logout Connection to hcl-admin closed.
client $
hc $ exit
Connection to hcl-admin closed.
client $
```

mdconfig

Description

Enables updates to the existing metadata schema.

Entering `mdconfig` with no parameters performs a validation on the current schema overlay file.

Syntax

```
mdconfig [-d | --dump | -t | --template | -c | --commit]
```

Options

- `-d` or `--dump`
Prints the current schema configuration as standard output.
- `-t` or `--template`
Returns an empty overlay file that you can use as a starting point to extend the schema configuration.
- `-c` or `--commit`
Commits the configuration change described in the overlay file to be read in as standard input.

Example

```
hc $ mdconfig -t
<?xml version="1.0" encoding="UTF-8"?>
<!--
    $Id: metadata_config_template.xml 7255 2006-03-01 00:1618Z wr152514

    Copyright 2005 Sun Microsystems, Inc. All rights reserved.
    Use is subject to license terms.
-->

<!--
    This template file provides a schema config overlay skeleton.
    Note that the fsview refers to the namespace of the schema.
-->

<metadataConfig>
  <schema>

    <namespace name="NAMESSPACE_NAME">

      <field name="FIELD1" type="string"/>

      <field name="FIELD2" type="string"/>

      <field name="FIELD3" type="string"/>

    </namespace>
  </schema>

  <fsViews>
    <fsView name="FSVIEW_NAME" filename="{FIELD3}.num" namespace=
      "NAMESSPACE_NAME">

      <attribute name="FIELD1"/>

      <attribute name="FIELD2"/>
    </fsView>

    ...
  </fsViews>

</metadataConfig>

hc $ cat overlay.xml | ssh admin@<ADMIN IP> mdconfig
hc $ cat overlay.xml | ssh admin@<ADMIN IP> mdconfig -c
```

netcfg

Description

Enables you to set and display various network configuration parameters, including:

- An SMTP server
- A network gateway
- External NTP servers
- An external log host
- Authorized clients

Syntax

```
netcfg [ smtp ] [ gw ] [ ntp ] [ loghost ] [ clients ]
```

Options

- `smtp`
Sets the location of the SMTP server that the Sun StorageTek 5800 system uses.
- `gw`
Sets the default gateway (router) that the Sun StorageTek 5800 system uses.
- `ntp`
Sets the external NTP servers that the Sun StorageTek 5800 system uses.
- `loghost`
Sets the external log host that the Sun StorageTek 5800 system uses.
- `clients`
Specifies the authorized clients for the Sun StorageTek 5800 system.

Examples

```
hc $ netcfg
Gateway: 10.7.227.254
Subnet: 255.255.252.0
Log Host: 10.123.45.1
NTP Server: 10.123.45.1
SMTP Server: 10.7.22.20
Authorized Clients: All

hc $ netcfg smtp 10.7.224.20 25
Please wait a couple of minutes while the configuration is updated.
nopasswd@10.123.45.1's password:
You must reboot the cell with "reboot --all" for changes to take
effect.
hc $ netcfg gw 10.7.226.254 255.255.255.0
Please wait a couple of minutes while the configuration is updated.
nopasswd@10.123.45.1's password:
You must reboot the cell with "reboot --all" for changes to take
effect.
hc $ netcfg ntp 10.7.224.30,10.7.224.40
Please wait a couple of minutes while the configuration is updated.
nopasswd@10.123.45.1's password:
You must reboot the cell with "reboot --all" for changes to take
effect.
hc $ netcfg loghost 10.7.224.15
Please wait a couple of minutes while the configuration is updated.
nopasswd@10.123.45.1's password:
You must reboot the cell with "reboot --all" for changes to take
effect.
hc $ netcfg clients 10.45.1.19,192.18.7.2,34.7.7.101
hc $ netcfg
Gateway: 10.7.226.254
Subnet: 255.255.255.0
Log Host: 10.7.224.15
NTP Server: 10.7.224.30,10.7.224.40
SMTP Server: 10.7.224.20:25
Authorized Clients: 10.45.1.19,192.18.7.2,34.7.7.101
hc $ netcfg clients all
hc $ netcfg
Gateway: 10.7.226.254
Subnet: 255.255.255.0
Log Host: 10.7.224.15
NTP Server: 10.7.224.30,10.7.224.40
SMTP Server: 10.7.224.20:25
Authorized Clients: all
```

passwd

Description

Allows the system administrator to change the admin password, as well as set and change an ssh pubkey for accessing the CLI.

Syntax

```
passwd [-P | --pubkey ]
```

Options

`-P` or `--pubkey`

Use this option if you are going to provide a public key to be used for login authorization.

Examples

```
hc $ passwd
password:
password (again):
authentication tokens successfully updated
hc $

client_$ ssh admin@cluster-admin passwd --pubkey < file.pub
authentication tokens successfully updated
```

where `file.pub` is the file containing the public key.

To return to interactive prompting, remove your private key or create a new public key with a pass phrase.

Note – Passwords must be at least six characters long.

Note – If you have already supplied a public key and then supply a new public key, the new key replaces the old key. Only one public key is allowed on the Sun StorageTek 5800 system.

reboot

Description

Reboots the cluster. Rebooting causes an alert to be sent.

Syntax

```
reboot [-F | --force ] [-A | --all ]
```

Options

-F or --force

Prompts for confirmation before forcing a reboot.

-A or --all

Reboots the switches along with the nodes.

Note – In non-interactive mode, reboot does not prompt for confirmation (no **-F** or **--force** is required).

Examples

```
hc $ reboot
Reboot? [y/N]: y
Connection to hcl-admin closed.
hc $ reboot --force
Connection to hcl-admin closed.
client_$ ssh admin@hcl-admin reboot
client_$
```

shutdown

Description

Shuts down (powers off) the cluster. System shutdown causes an alert to be sent.



Caution – Use the `shutdown` command only in consultation with Sun StorageTek 5800 system support personnel.

Syntax

`shutdown [-F | --force]`

Options

`-F` or `--force`

Forces a system shutdown without prompting for confirmation.

Note – In non-interactive mode, `shutdown` does not prompt for confirmation (no `-F` or `--force` is required).

Examples

```
hc $ shutdown
It is not safe to shutdown the system
Do you want to shutdown anyway? [y/N]: n
hc $ shutdown Shutdown? [y/N]: y
Connection to hcl-admin closed.
client_$ ssh admin@hcl-admin shutdown
Connection to hcl-admin closed.
```

sysstat

Description

Displays overall and performance statistics about the system.

Syntax

```
sysstat [ -i seconds | --interval=seconds] [ -v | --verbose]
```

Options

- **-i** or **--interval**
Specifies a number of seconds at which to repeat the statistics listing. (The default here is 5.)
- **-v** or **--verbose**
Displays voluminous information about the system.

Examples

```
hc $ sysstat  
Silo 1, Cell 1: Cluster Online. Estimated free space: 7T.  
16 nodes online, 64 disks online, 0 disks unrecovered  
Data VIP 10.7.225.222. Admin VIP 10.7.225.223.  
Data services Online, Query Available  
Failure Tolerance: Fault Tolerant  
Last Recovery Cycle: Completed 5 hours ago.  
Next Recovery Cycle: To complete in 9 hours.
```

The output that the `sysstat` command produces is as follows:

- **Data services Online** means that the system is available to read and write to, while **Data services Offline** means that the system is not available.

- Query Available means it is possible to query and add metadata, while Query Unavailable means it is not possible.
- Failure Tolerance- Fault Tolerant means the system is operational and able to withstand failures. Failure Tolerance- Degraded means there is one more failure before the system loses data, while Failure Tolerance- Not Tolerant indicates that there have been too many failures, or the cluster is not currently online.

```
hc $ sysstat -v
NODE-101      [ONLINE]
DISK-101:0    [ONLINE]
DISK-101:1    [OFFLINE]
DISK-101:2    [ONLINE]
DISK-101:3    [ONLINE]
NODE-102      [ONLINE]
DISK-102:0    [ONLINE]
DISK-102:1    [ONLINE]
DISK-102:2    [ONLINE]
DISK-102:3    [ONLINE]
NODE-103      [ONLINE]
DISK-103:0    [ONLINE]
.
.
.
```

Verbose output includes the online/offline status of each node and disk in the cluster. The online/offline status reported by the command refers to the logical system status. To see the state of hardware components, refer to [“hwstat” on page 51](#).

upgrade

Description

Upgrades the cluster to a newer version of the Sun StorageTek 5800 system software. You perform an upgrade from a DVD, or from an ISO image located on a web server. For a detailed description, see [“Upgrading the System” on page 41](#).

Syntax

`upgrade [dvd | <image-name> | downloaded]`

Options

- `dvd`
Performs an upgrade from a DVD.
- `<image-name>`
Downloads the image from an IP address and upgrades in one step.
- `downloaded`
Downloads the image from an IP address and upgrades at a later time.

Examples

```
hc $ upgrade dvd

hc $ upgrade http://123.45.67.89/honeycombdvd.iso

hc $ upgrade download http://123.45.67.89/honeycombdvd.iso
hc $ upgrade downloaded
```

version

Description

Displays the version of the system software.

Syntax

version

Options

None

Example

```
hc $ version  
Sun StorageTek 5800 System v1.0 build 1.0.7 built on Tues Apr 10  
18:05:32
```

wipe

Description

Deletes all data and metadata on the cell. It does not delete the system software, however. After issuing the `wipe` command, you use the `reboot` command to complete the process.



Caution – All data and metadata in the Sun StorageTek 5800 system is lost when you run the `wipe` command. Consult Sun StorageTek 5800 system technical support prior to using this command.

Syntax

`wipe [-F | --force]`

Options

`-F` or `--force`

Forces a system wipe without prompting for confirmation.

Examples

```
hc $ wipe
Destroy all data? [y/N]:
```


Glossary

API	Application Programming Interface. A set of routines, protocols, and tools used for building software applications.
attribute	An entry in the schema that associates a name with a type. For example, the name Doctor may be of type string. Metadata is stored by assigning a value of the appropriate type to an attribute name, and attributes may also be used to create views.
cell	The basic building block of the Sun StorageTek 5800 system. It consists of either 8 or 16 Sun StorageTek 5800 system nodes, 2 Load Balancers, and 1 Service Node.
client	An application that runs on a personal computer or workstation and relies on a server to perform some operations.
cluster	See the Sun StorageTek 5800 system cluster.
CPU	Central Processing Unit. The brains of the computer, it is sometimes referred to simply as the processor or central processor. It is where most calculations take place
ctime	creation time. The system metadata includes information on the creation time, data length, and data hash.
data object	A stored file associated with an object ID (OID).
data hash	Hashes are used for accessing data or for security. A hash, also called a message digest, is a number generated from a string of text. The hash is substantially smaller than the text itself, and is generated by a formula in such a way that it is extremely unlikely that some other text will produce the same hash value.
disk mask	A current record of disk availability across the cluster.
double	One of the defined attribute types.

DTD	Document Type Definition. It defines the legal building blocks of an XML document. The DTD defines the document structure with a list of legal elements, thus providing an application-independent way of sharing data.
dual star topology	A network topology in which two concentric rings connect each node on a network instead of one network ring that is used in a ring topology. Typically, the secondary ring in a dual-ring topology is redundant and is used as a backup in case the primary ring fails. In these configurations, data moves in opposite directions around the rings. Each ring is independent of the other until the primary ring fails and the two rings are connected to continue the flow of data traffic.
encoding algorithm	A coding formula or set of steps for solving a particular problem. To qualify as an algorithm, the set of rules must be unambiguous and have a clear stopping point.
fragment	A piece of a file. Files over a certain size are stored in several <i>chunks</i> or fragments rather than in a single contiguous sequence of bits in one place. The Sun StorageTek 5800 system stores fragments of files across multiple disks and nodes using 5+2 encoding. Thus, when an object of any type (for example, an MP3 binary or a text file) is stored in the Sun StorageTek 5800 system, it is divided into five data fragments and two corresponding parity fragments.
front end	The client portion of a client/server application that requests services across a network from a server, which is sometimes known as the back end.
FRU	Field-replaceable unit. Describes any hardware device, or more commonly a part or component of a device or system, that can easily be replaced by a skilled technician without having to send the entire device or system to be repaired. As the name implies, the unit can be replaced in the field (that is, at the user location).
GB	Gigabyte. Represents 2 to the 30th power (1,073,741,824) bytes. One gigabyte is equal to 1,024 megabytes.
HTML	HyperText Markup Language. Designed to display data and focus on how data looks. The tags you use to markup HTML documents and the document's structure are predefined, so that you can only use tags that are defined in the HTML standard.
HTTP	HyperText Transfer Protocol. Underlying protocol used by the World Wide Web. HTTP defines how messages are formatted and transmitted, and what actions Web servers and browsers should take in response to various commands.
metadata	Extra information about the data object. Describes how and when and by whom a particular set of data was collected, and how the data is formatted. There are two main types of metadata in the Sun StorageTek 5800 system: system and user metadata.

- MP3** Moving Pictures Experts Group (MPEG), audio layer 3 file. Layer 3 is one of three coding schemes (layer 1, layer 2 and layer 3) for the compression of audio signals.
- NIC** Network Interface Card. An expansion board inserted into a computer to allow the computer to be connected to a network. Most NICs are designed for a particular type of network, protocol, and media, although some can serve multiple networks.
- node** A processing location. A node can be a computer or some other device, such as a printer. Every node has a unique network address.
- NTP** Network Time Protocol. An Internet standard protocol (built on top of TCP/IP) that assures accurate synchronization to the millisecond of computer clock times in a network.
- object** Any item that can be individually selected and manipulated. For example, in object-oriented programming, an object is a self-contained entity that consists of both data and procedures to manipulate the data.
- OID** Object ID. A unique identifier for each stored object included in the system metadata.
- OS X** Version 10 of the Apple Macintosh operating system. It incorporates support for Unix-based applications as well as for those written just for the Macintosh.
- query** A request for information from a database.

**Reed-Solomon
Encoding Algorithm**

- Files are protected in the Sun StorageTek 5800 system using the Reed Solomon (RS) encoding algorithm. The RS algorithm is part of a code family that efficiently builds redundancy into a file to guarantee reliability in the face of multiple part failures in the storage system.
- SATA** Serial ATA. An evolution of the Parallel ATA physical storage interface. SATA is a serial link- a single cable with a minimum of four wires- that creates a point-to-point connection between devices. Transfer rates for SATA begin at 150MBps.
- schema** The schema defines the way that the Sun StorageTek 5800 system metadata is structured. It consists of attributes, each of which has a defined type.
- SMTP** Simple Mail Transfer Protocol. A protocol for sending e-mail messages between servers. Most e-mail systems that send mail over the Internet use SMTP to send messages from one server to another; the messages can then be retrieved with an e-mail client using either POP or IMAP.
- socket** A receptacle for a microprocessor or other hardware component.

stateless Having no information about what occurred previously. Most applications maintain state, which means that they remember what you were doing last time you ran the application, as well as remembering all your configuration settings.

The World Wide Web, on the other hand, is intrinsically stateless because each request for a new Web page is processed without any knowledge of previous pages requested. This is one of the chief drawbacks to the HTTP protocol. Because maintaining state is extremely useful, programmers have developed a number of techniques to add state to the World Wide Web, including server APIs and the use of cookies.

string A contiguous sequence of symbols or values, such as a character string (a sequence of characters) or a binary digit string (a sequence of binary values).

Sun StorageTek 5800 system cluster Refers to the 16 nodes of the Sun StorageTek 5800 system.

Sun StorageTek 5800 system or Silo Consists of a half-cell of 8 nodes or a full cell of 16 nodes.

system metadata Metadata that includes a unique identifier for each stored object, called the OID, as well as information on creation time (ctime), data length, and data hash. It is automatically maintained by the system.

user metadata Metadata that is added by the user of the Sun StorageTek 5800 system. User metadata consists of *name=value* pairs. The name is defined in the system schema as of a certain type (for example, a string), and the value is associated with the name at the time data is stored.

UTC Coordinated Universal Time. A time scale that couples Greenwich Mean Time, which is based solely on the Earth's inconsistent rotation rate, with highly accurate atomic time. When atomic time and Earth time approach a one second difference, a leap second is calculated into UTC.

VIP Virtual IP address. The Sun StorageTek 5800 system cluster has two public host addresses, one for the data VIP and one for the administrative VIP.

virtual views Queries against metadata that are expressed externally as file system paths and file names. A virtual file system view is defined using the metadata attributes defined in the schema.

WebDAV Web-based Distributed Authoring and Versioning. A set of extensions to the HTTP protocol that provides a network protocol for creating interoperable, collaborative applications.

XML XML offers a widely adopted standard way of representing text and data in a format that can be processed with relatively little human intervention and exchanged across diverse hardware, operating systems and applications.

Index

Numerics

5+2 encoding scheme, 10

A

accessing Sun StorageTek 5800 system storage
 through a file system view, 9
 through an API, 8

administrative interface, 8

administrative VIP, 8

architecture. *See* clustered design

B

basic system building blocks, 3

C

cells, 3

CLI

 accessing through the administrative VIP, 8

 command reference, 45 to 69

 functions, 8

client scripts, 39

clustered design, 2

commands

 alertcfg

 configuring an alert email address, 17

 configuring the alert subsystem, 17

 detailed description and syntax, 46

 obtaining SMTP email alerts, 23

 df

 detailed description and syntax, 47

 getting disk status, 22

 help

 detailed description and syntax, 48

 hwcfg

 detailed description and syntax, 49

 hwstat

 detailed description and syntax, 51

 obtaining FRU listings, 21

 ifconfig

 detailed description and syntax, 54

 setting external virtual IP addresses, 17

 license

 detailed description and syntax, 55

 logout

 detailed description and syntax, 56

 mdconfig

 detailed description and syntax, 57

 upgrading the schema configuration, 32

 netcfg

 configuring a gateway, 16

 configuring an external log host, 16

 configuring an external NTP server, 16

 configuring an external SMTP server, 16

 configuring authorized clients, 17

 detailed description and syntax, 59

 setting network configuration parameters, 15

 viewing system parameters, 59

 passwd

 changing the system password, 38

 detailed description and syntax, 61

 reboot

 detailed description and syntax, 63

 rebooting the system, 41

 shutdown

 detailed description and syntax, 64

 shutting down the system, 40

- ssh
 - configuring a public key, 38
- sysstat
 - detailed description and syntax, 65
 - getting system version and status, 19
- upgrade
 - detailed description and syntax, 67
 - upgrading the system, 41
- version
 - detailed description and syntax, 68
 - viewing the system software version, 20
- wipe
 - detailed description and syntax, 69
 - wiping data from the system, 43
- CLI reference, 45 to 69
- components
 - email alerts for failures, 5
 - replacement of individual units, 5
- composition of the Sun StorageTek 5800 system, 2
- creating new virtual views, 36
- customizing the configuration
 - configuring a gateway, 16
 - configuring an alert email address, 17
 - configuring an external log host, 16
 - configuring an external NTP server, 16
 - configuring an external SMTP server, 16
 - configuring authorized clients, 17
 - configuring the alert subsystem, 17
 - setting external virtual IP addresses, 17, 18
 - setting network configuration parameters, 15
 - setting the data and admin VIP addresses, 18
 - viewing system parameter settings, 59

D

- data and parity fragment storage example, 11
- data reliability and availability, 9
- data VIP, 8

E

- email alerts for component failures, 5

F

- file fragmentation, 11

H

- hardware overview, 4 to 5

M

- metadata
 - defining, 7
 - schema, 7
 - system and user (extended), 7
- minimum and maximum Sun StorageTek 5800
 - system configuration, 4
- monitoring the system
 - getting disk status, 22
 - getting system version and status, 19
 - obtaining FRU listings, 21
 - obtaining SMTP email alerts, 23
 - viewing system software version, 20

N

- node
 - failure and fragment relocation example, 13
 - failure and recovery, 13

O

- OA software layer, 2
- overview of the product, 1 to 3

P

- placement algorithm, 11
- product overview, 1 to 3
 - 5+2 encoding scheme, 10
 - basic system building blocks, 3
 - cells, 3
 - clustered design, 2
 - composition of the StorageTek 5800 system, 2
 - data and parity fragment storage example, 11
 - data reliability and availability, 9
 - file fragmentation, 11
- node
 - failure and fragment relocation example, 13
 - failure and recovery, 13
- nodes, 3, 27
- OA software layer, 2
- placement algorithm, 11

Q

- query, 7, 20, 34

R

- Reed Solomon algorithm, 9
- replacement of individual components, 5

S

- schema, 7
- serviceability and FRUs, 5
- software overview, 6 to 13
- storage access
 - admin VIP, 8
 - data VIP, 8
- system administration, 37 to ??
 - changing the system password, 38
 - configuring a public key, 38
 - powering up the system, 40
 - rebooting the system, 41
 - recovering from a power failure, 41
 - shutting down the system, 40
 - upgrading the system, 41
 - wiping data from the system, 43
- system nodes, 3, 27

T

- two ways to access Sun StorageTek 5800 system
 - storage
 - through API access, 8
 - through filesystem views, 8

U

- using CLI commands in a client script, 39

W

- WebDav, 9
 - browsing files with, 34
 - definition, 34
 - features, 34

