



Sun StorageTek™ 5800 System Administration Guide

Version 1.1

Sun Microsystems, Inc.
www.sun.com

Part No. 819-7555-10
November 2007, Revision A

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

Copyright © 2007 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun, Sun Microsystems, the Sun logo, Java, Solaris, Sun StorageTek 5800 System, Java Web Start, JDBC, SPARC, Sun Fire, and Sun StorageTek are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

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

UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Products covered by and information contained in this service manual are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

Use of any spare or replacement CPUs is limited to repair or one-for-one replacement of CPUs in products exported in compliance with U.S. export laws. Use of CPUs as product upgrades unless authorized by the U.S. Government is strictly prohibited.

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 © 2007 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, Etats-Unis. Tous droits réservés.

Sun, Sun Microsystems, le logo Sun, Java, Solaris, Sun StorageTek 5800 System, Java Web Start, JDBC, SPARC, Sun Fire, et Sun StorageTek sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

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

UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Ce produit est soumis à la législation américaine en matière de contrôle des exportations et peut être soumis à la réglementation en vigueur dans d'autres pays dans le domaine des exportations et importations. Les utilisations, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes biologiques et chimiques ou du nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers les pays sous embargo américain, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exhaustive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites. L'utilisation de pièces détachées ou d'unités centrales de remplacement est limitée aux réparations ou à l'échange standard d'unités centrales pour les produits exportés, conformément à la législation américaine en matière d'exportation. Sauf autorisation par les autorités des Etats-Unis, l'utilisation d'unités centrales pour procéder à des mises à jour de produits est rigoureusement interdite.

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



Adobe PostScript

Contents

Preface xix

1. Product Overview 1

About the 5800 System 1

5800 System Hardware 3

 Full-Cell and Half-Cell Configurations 3

 Storage Nodes 9

 Service Node 12

 Gigabit Ethernet Switches 14

 Network Patch Panel 15

5800 System Software 16

 Bundled Software 16

 Metadata 17

 System Metadata 17

 Extended Metadata 17

 The Placement Algorithm 17

 User Interfaces 18

 Application Program Interface (API) 18

 WebDAV 19

 CLI and GUI 19

2. Using the Administrative Interfaces	21
About Administrative Interfaces	21
Hive-Level and Cell-Level Functions	22
Using the CLI	22
Logging In To the CLI	23
Logging Out of the CLI	23
▼ To Create and Execute a Script for CLI Commands	24
Getting Help on CLI Commands	25
Using the GUI	26
▼ To Launch the GUI from a Browser	26
▼ To Set MIME Associations for the Browser	27
▼ To Launch the GUI from the Command Line	28
Getting Help on the GUI	29
Logging Out of the GUI	33
Multiple Users Accessing the CLI or GUI	33
Task Overview	34
3. Configuring System Access Parameters	37
About System Access	37
Administrative IP Address	38
▼ To Configure the Administrative IP Address Using the CLI	38
▼ To Configure the Administrative IP Address Using the GUI	38
Data IP Address	39
▼ To Configure the Data IP Address Using the CLI	39
▼ To Configure the Data IP Address Using the GUI	40
Service Node IP Address	40
▼ To Configure the Service Node IP Address Using the CLI	40
▼ To Configure the Service Node IP Address Using the GUI	41
Administrative Password	41

- ▼ To Configure the Administrative Password Using the CLI 42
- ▼ To Configure the Administrative Password Using the GUI 42
- The Public Key 42
 - ▼ To Configure a Public Key Using the CLI 43
 - Configuring a Public Key Using the GUI 44
- Authorized Subnetworks 44
 - ▼ To Configure Authorized Subnetworks Using the CLI 45
 - ▼ To Configure Authorized Subnetworks Using the GUI 45
- 4. Configuring Network Parameters 47**
 - About Network Parameters 47
 - The Gateway 48
 - ▼ To Configure a Gateway Using the CLI 48
 - ▼ To Configure a Gateway Using the GUI 48
 - NTP Servers 49
 - ▼ To Configure an NTP Server Using the CLI 49
 - ▼ To Configure an NTP Server Using the GUI 49
 - ▼ To Check the System Time Using the CLI 50
 - ▼ To Check the System Time Using the GUI 50
 - DNS 50
 - ▼ To Configure DNS Using the CLI 50
 - ▼ To Configure DNS Using the GUI 51
- 5. Managing System Hardware and Data 53**
 - Starting and Stopping System Components 53
 - ▼ To Shut Down a Cell Using the CLI 54
 - ▼ To Shut Down a Cell Using the GUI 54
 - ▼ To Reboot a Cell Using the CLI 55
 - ▼ To Reboot a Cell Using the GUI 55

▼ To Power Up a Cell	55
System Performance and Capacity Impact after Disks or Nodes are Offline	56
Recovering From a Power Failure	57
Data Availability After Power Loss	57
Deleting All Data From the System	58
▼ To Delete All Data Using the CLI	59
▼ To Delete All Data Using the GUI	59
6. Configuring System Notifications and Online Registration	61
Configuring System Notifications	61
▼ To Configure Email Notifications Using the CLI	62
▼ To Configure Email Notifications Using the GUI	63
▼ To Configure an External Log Host Using the CLI	63
▼ To Configure an External Log Host Using the GUI	64
Registering Online	64
Registering Using the CLI	64
▼ To Register Using the GUI	64
7. Configuring Metadata and Virtual File System Views	67
The Metadata Schema	67
Metadata Schema File	68
Schema File Structure	68
Schema File DTD	69
Example Schema File	71
Metadata	73
System Metadata	73
Extended Metadata	73
Metadata Types	74
Namespaces	74

Writable and Extensible Namespaces	75
Reserved Namespaces	75
The <code>system</code> Namespace	76
The <code>filesystem</code> Namespace	76
Fully Qualified Names	76
Planning Namespaces	77
Tables and Columns	77
Table Example	78
The <code>length</code> Attribute for Fields	78
Planning Tables	79
Planning Table Rows	79
Planning Table Rows Example	80
Planning Tables Checklist	81
Indexes	82
Planning Indexes	82
Planning Indexes Examples	83
Excluding Attributes from Indexes and Queries	87
Planning Tables and Indexes Checklist	87
Virtual File System Views	87
WebDAV	88
Using WebDAV to Browse Virtual File System Views	88
WebDAV Example	88
Metadata Attributes and WebDAV Properties	90
Including Additional File Attributes in Virtual Views	90
Directory Structure in a Virtual File System View	91
Virtual File System Views in the Schema File	93
Summary of Metadata Schema Elements	94
Configuring the Metadata Schema Using the CLI	95

- ▼ To Modify the Schema File Using the CLI 95
- Configuring the Metadata Schema Using the GUI 96
 - ▼ To Display the Current Metadata Schema 97
 - ▼ To Display the Fields in a Namespace 97
 - ▼ To Display the Fields in a Table 97
 - ▼ To Change the Metadata Schema 97
- Creating Namespaces 98
 - ▼ To Create Namespaces 98
- Creating Tables 99
 - ▼ To Create Tables 99
 - ▼ To Add Fields to an Existing Namespace 100
- Configuring Virtual File System Views Using the GUI 101
 - ▼ To Display the Current Virtual File System Views 101
 - ▼ To Display the Fields in a View 101
 - ▼ To Create New Virtual File System Views 101
 - ▼ To Preview Virtual File System Views 103
- 8. Monitoring the System Using the CLI 105**
 - Obtaining System Status 105
 - Displaying Performance Statistics 108
 - Viewing the System Software Version 109
 - Obtaining FRU Listings 110
 - Obtaining Disk Status 111
 - Obtaining Voltage, Temperature, and Fan Speed Data 113
- 9. Monitoring the 5800 System Using the GUI 115**
 - Monitoring the System 115
 - ▼ To Display Failed Components 116
 - ▼ To Display the System Space Usage 116

▼	To Display System Performance Statistics	116
▼	To Display the Environmental Status	116
	Monitoring Cells	117
▼	To Display the System Software Version on a Cell	117
▼	To Display the Nodes in a Cell	117
▼	To Display the Disks in a Cell	117
▼	To Display the Cell IP Addresses	118
	Monitoring Nodes and Disks	118
▼	To Display the FRU ID of a Node	118
▼	To Display Node Space Usage	118
▼	To Display the Node Status	119
▼	To Display the Disks in a Node	119
▼	To Display Disk Statistics	119
10.	Implementing Disaster Recovery	121
	About the 5800 System Implementation of NDMP	121
	Using NetVault to Implement Disaster Recovery	122
	Checking NDMP Status	123
	General Guidelines for Backing Up Data	124
	General Guidelines for Restoring Data	124
A.	Reference: CLI Commands	127
	alertcfg	129
	Description	129
	Syntax	129
	Options	129
	Examples	129
	cellcfg	130
	Description	130

- Syntax 130
- Options 130
- Examples 131
- copyright 132
 - Description 132
 - Syntax 132
 - Options 132
 - Examples 132
- date 133
 - Description 133
 - Syntax 133
 - Option 133
 - Example 133
- df 134
 - Description 134
 - Syntax 134
 - Options 134
 - Examples 135
- help 136
 - Description 136
 - Syntax 136
 - Option 136
 - Examples 137
- hiveadm -s|--status 138
 - Description 138
 - Syntax 138
 - Options 138
 - Example 138

hivecfg	139
Description	139
Syntax	139
Options	139
Examples	141
hwstat	142
Description	142
Syntax	142
Options	142
Examples	143
logout	144
Description	144
Syntax	144
Options	144
Example	144
mdconfig	145
Description	145
Syntax	145
Options	145
Example	146
passwd	148
Description	148
Syntax	148
Option	148
Examples	149
perfstats	150
Description	150
Syntax	150

Options	150
Examples	151
	152
reboot	153
Description	153
Syntax	153
Options	153
Examples	153
sensors	154
Description	154
Syntax	154
Option	154
Example	155
shutdown	156
Description	156
Syntax	156
Options	156
Examples	156
sysstat	157
Description	157
Syntax	157
Options	157
Examples	158
version	160
Description	160
Syntax	160
Options	160
Example	161

wipe	162
Description	162
Syntax	162
Options	162
Example	162

Glossary	163
-----------------	------------

Index	169
--------------	------------

Figures

FIGURE 1-1	5800 System Full-Cell Front View	4
FIGURE 1-2	Rear View of a Two-Cell System, Showing Network Patch Panel Connections	6
FIGURE 1-3	5800 System Half-Cell Front View	8
FIGURE 1-4	Storage Node Front Panel Components	10
FIGURE 1-5	Storage Node Back Panel Components	12
FIGURE 1-6	Service Node Front Panel Components	13
FIGURE 1-7	Service Node Back Panel Components	14
FIGURE 1-8	Gigabit Ethernet Switch	15
FIGURE 1-9	Network Patch Panel	15
FIGURE 2-1	Main GUI Screen	29
FIGURE 2-2	Help Button in Toolbar	29
FIGURE 2-3	GUI Help Pane	31
FIGURE 2-4	Log Out Button in Toolbar	33
FIGURE 7-1	Virtual File System View Configuration Example	103
FIGURE 9-1	Online Node	119
FIGURE 9-2	Offline Node	119

Tables

TABLE 1-1	Storage Node Features	9
TABLE 1-2	Storage Node Front Panel LED and Switch Descriptions	10
TABLE 1-3	Service Node Features	12
TABLE 1-4	Gigabit Ethernet Switch Connections to the Network Patch Panel	15
TABLE 1-5	5800 Storage System User Interface Functional Capabilities	18
TABLE 2-1	Help Tabs	32
TABLE 2-2	Help Icons	32
TABLE 2-3	5800 System Administrative Tasks	34
TABLE 7-1	Supported Metadata Types	74
TABLE 7-2	Reserved Domains	75
TABLE 7-3	<code>system</code> Namespace Contents	76
TABLE 7-4	<code>filesystem</code> Namespace Contents	76
TABLE 7-5	Example Table (<i>reference</i> Table)	78
TABLE 7-6	Number of Bytes Used By Each Column in a Table Row Definition	80
TABLE 7-7	Examples of Fields to Be Added to a Table	80
TABLE 7-8	Number of Bytes Used By Each Element in Index	83
TABLE 7-9	Example of Fields to Be Added to a Table	83
TABLE 7-10	WebDAV Property Names and System Metadata Attributes	90
TABLE 7-11	Metadata Schema Fields	94
TABLE 9-1	Mouse-Over Values on the Status at a Glance Panel	120

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](#) describes the interfaces you can use to perform administrative tasks and lists which tasks are available through the command-line interface (CLI) and graphical user interface (GUI).

[Chapter 3](#) describes the system access parameters and how to configure them.

[Chapter 4](#) describes the network access parameters and how to configure them.

[Chapter 5](#) describes how to start, stop, and reboot system components.

[Chapter 6](#) describes how to set up email notifications and the system log file.

[Chapter 7](#) describes metadata and virtual file system views and how to configure them.

[Chapter 8](#) describes how to get information about the system hardware and software using the CLI.

[Chapter 9](#) describes how to get information about the system hardware and software using the GUI.

[Chapter 10](#) describes how to protect your 5800 system from catastrophic loss.

[Appendix A](#) provides complete syntax for all CLI commands.

The [Glossary](#) provides a list of pertinent words and phrases and their definitions for the 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 table lists the documents for this product. The online documentation is available at:

<http://docs.sun.com/app/docs/prod/stortek.5800#hic>

Title	Part Number	Format	Location
<i>Sun StorageTek 5800 System Regulatory and Safety Compliance Manual</i>	819-3809-xx	HTML PDF	Online
<i>Sun StorageTek 5800 System Site Preparation Guide</i>	820-1635-xx	HTML PDF	Online
<i>Sun StorageTek 5800 System Release Notes</i>	819-7559-xx	HTML PDF	Online
<i>Sun StorageTek 5800 System Client API Reference Guide</i>	819-7557-xx	PDF	Online
<i>Sun StorageTek 5800 System SDK Developer's Guide</i>	819-7558-xx	PDF	Online

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 819-7555-10

Product Overview

This chapter provides an overview of the Sun StorageTek™ 5800 System. It includes the following sections:

- [“About the 5800 System” on page 1](#)
- [“5800 System Hardware” on page 3](#)
- [“5800 System Software” on page 16](#)

About the 5800 System

The 5800 system is a self-contained storage unit that attaches directly to a network. It does not use typical data and file structures: there are no pools, volumes, logical unit numbers (LUNs), or redundant arrays of independent disks (RAID) to set up or manage.

Instead, the 5800 system uses an object-oriented methodology to store fixed-content data files (files that will never be modified) as if they were individual objects. The storage system assigns each of these *data objects* a unique identifier called an Object ID (OID) based on the attributes of the data object. Applications use the OIDs to query and retrieve the data objects.

The 5800 system employs a cluster of *storage nodes*. Each storage node is an individual server with CPU processing power, RAM, and four Serial Advanced Technology Attachment (Serial ATA) disk drives for storage. Each storage node has hardware and software that is identical to that of the other storage nodes, but each storage node operates independently. This configuration allows for all storage processing and data path operations to be distributed across the available processing power of the system and enhances both system reliability and performance.

Each 5800 system includes a single *service node* with preconfigured software and firmware. The system uses the service node for initial configuration, for troubleshooting, and to upgrade the system software.

The basic 5800 system is a *full-cell configuration* that includes 16 storage nodes, 1 service node, 2 gigabit Ethernet switches, a network patch panel, and preinstalled operating system and software. A *half-cell configuration*, which includes only 8 storage nodes, is also allowed. You can expand (scale) a half-cell configuration to a full-cell configuration. You can also expand a full-cell configuration to create *multicell configurations*, also referred to as *hives*. Only full-cells are allowed in multicell configurations.

Features of the 5800 system include:

- Both a command-line interface (CLI) and a graphical user interface (GUI) to monitor system performance and status and perform administration tasks.
- Metadata associated with each data object that you can customize to:
 - Emulate a database query. An application programming interface (API) allows developers to write applications for storing, retrieving, querying, and deleting data.
 - Present a simulated hierarchical directory structure with the Web-based Distributed Authoring and Versioning (WebDAV) tool. (WebDAV is not supported for multi-cell configurations.)
- Two virtual IP address per half- or full-cell: one for data processing and one for management operations. An integrated Domain Name Service (DNS) establishes the host name.
- Self-healing operations that increase system reliability. When a disk drive fails, the system rebuilds the data on other disks in the system.
- Hot-pluggable disks that can be removed and replaced easily after issuing a command to take the disk offline.
- A distributed data storage model that utilizes the Reed Solomon algorithm commonly used in RAID systems to increase system reliability.
- An integrated Sun Solaris 10 Operating System (Solaris OS).

5800 System Hardware

The 5800 system ships from the factory fully installed in a rack and with the software loaded in the default configuration.

This section includes the following topics:

- [“Full-Cell and Half-Cell Configurations” on page 3](#)
- [“Storage Nodes” on page 9](#)
- [“Service Node” on page 12](#)
- [“Gigabit Ethernet Switches” on page 14](#)
- [“Network Patch Panel” on page 15](#)

Full-Cell and Half-Cell Configurations

The full-cell is the basic building block of the 5800 system. A full-cell includes a service node, 16 storage nodes, 2 gigabit Ethernet switches, and a network patch panel. The front view of a single full-cell is shown in [FIGURE 1-1](#). Additional full-cells in a multicell system (referred to as a *hive*) are identical.

FIGURE 1-1 5800 System Full-Cell Front View

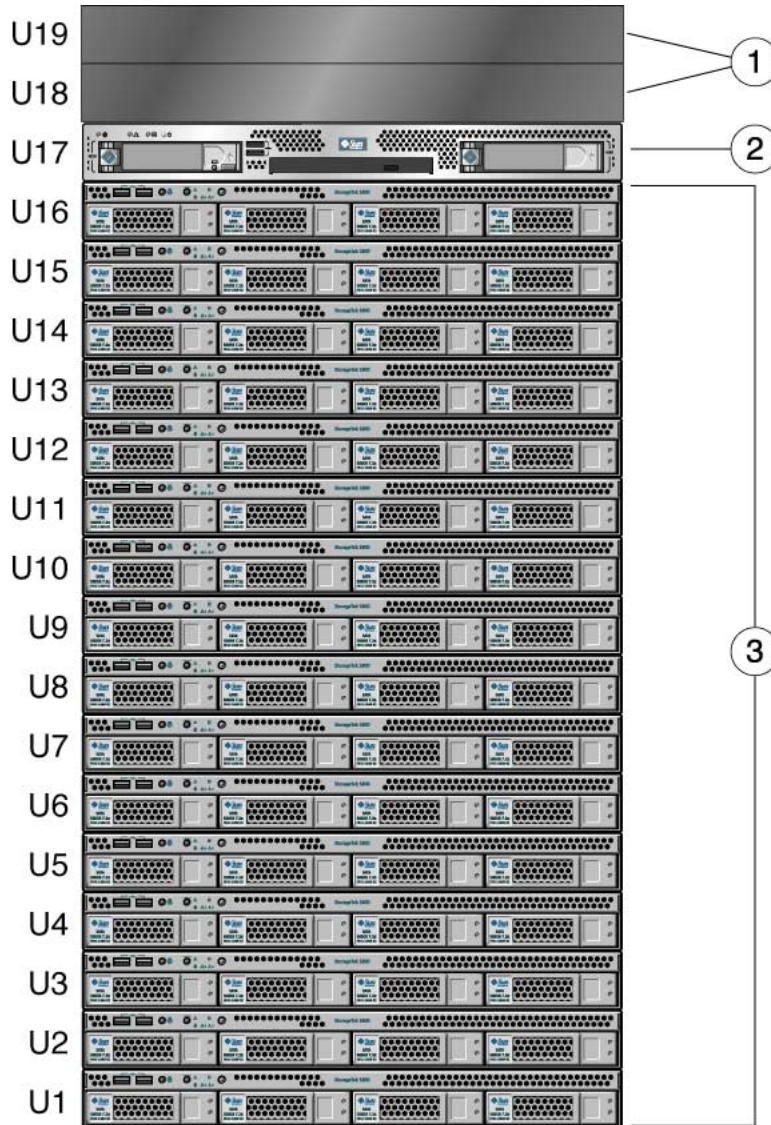


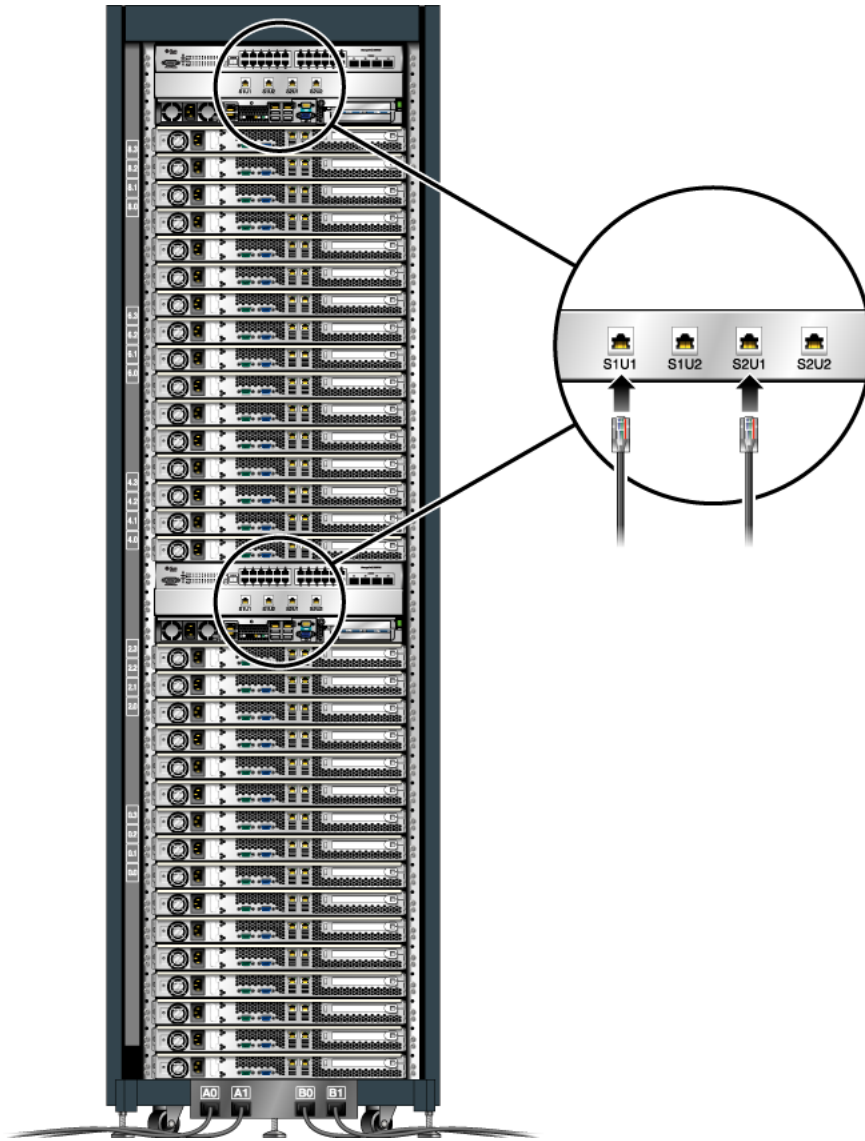
Figure Legend

- 1 Filler panels hiding two rear-facing gigabit Ethernet switches and a rear-facing network patch panel located behind the lower switch
- 2 Service node
- 3 16 storage nodes

The gigabit Ethernet switches are installed with their ports facing towards the back of the cabinet. Filler panels are installed in the front of the cabinet at the same elevation as the two switches. The network patch panel is mounted behind the lower switch on the back of the cabinet (see [FIGURE 1-2](#)). Both of the gigabit Ethernet switches are connected to the service node, to all the storage nodes, and to the network patch panel.

Storage node number 101 is on the bottom of the cell; the numbering of the storage nodes gets higher as you go up the cell, with node 102 above node 101, followed by node 103 above that and so on.

FIGURE 1-2 Rear View of a Two-Cell System, Showing Network Patch Panel Connections



If you chose to populate the cabinet with only a half-cell (eight storage nodes), slots U9 through U16 have filler panels on the front, as shown in [FIGURE 1-3](#). A half-cell configuration is stand-alone and cannot be combined with another full-cell. To add another full-cell to the hive, you must first add additional storage nodes to expand the half-cell to a full-cell configuration. Because a half-cell has a reduced number of storage nodes, it does not have the same inherent reliability as that of a full-cell with 16 storage nodes.

FIGURE 1-3 5800 System Half-Cell Front View

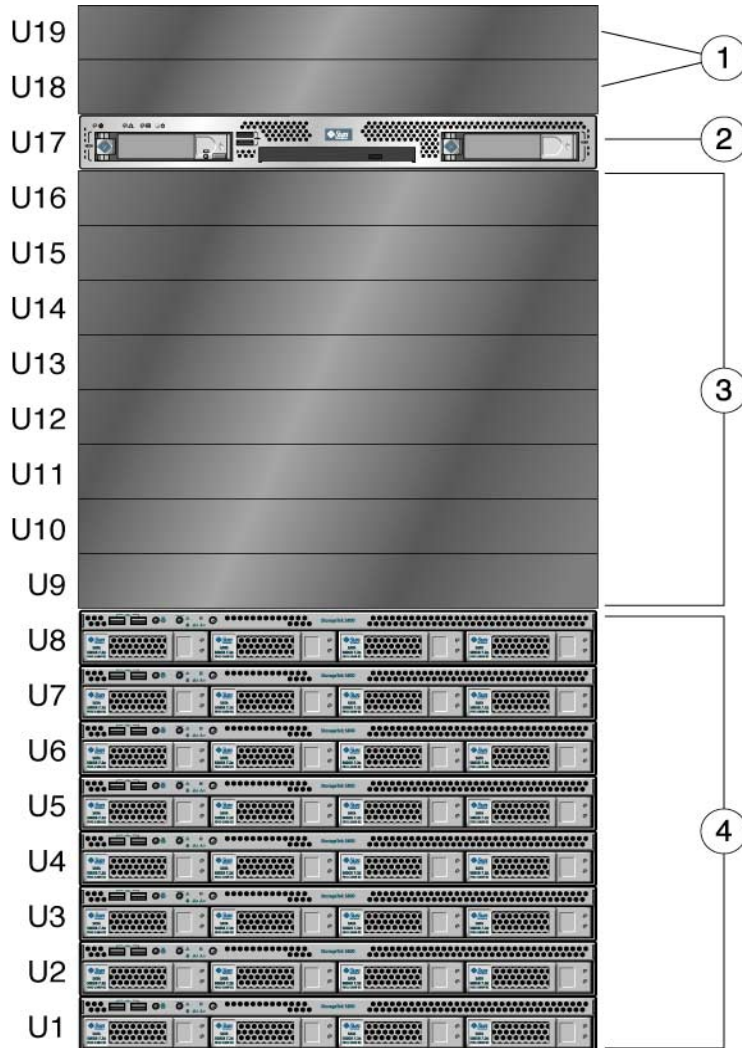


Figure Legend

-
- 1 Filler panels hiding two rear-facing gigabit Ethernet switches and a rear-facing network patch panel located behind the lower switch
 - 2 Service node
 - 3 8 filler panels
 - 4 8 storage nodes
-

Three configurations of the 5800 system can be housed in a single cabinet:

- A full-cell with 16 storage nodes
- A half-cell with eight storage nodes
- Two full-cells for a total of 32 storage nodes

Systems having more than two full-cells must be accommodated with additional cabinets.

Storage Nodes

The key features of the 5800 system storage node are listed in [TABLE 1-1](#).

TABLE 1-1 Storage Node Features

Component	Description
CPU	<ul style="list-style-type: none">• One single-core AMD Opteron processor• Processor frequency: 2.2 GHz• 1 MB level 2 cache
Memory	Three GB using two 1-GB ECC DIMMs and two 512-MB ECC DIMMs
Hard disk drives	Four 500-GB Serial ATA disk drives
Power supply	350W
Network I/O	Two 10/100/1000BASE-T gigabit Ethernet ports
System management	Intelligent Platform Management Interface (IPMI) 1.5-compliant service processor module

[FIGURE 1-4](#) shows the front panel components of a 5800 system storage node. Pressing the locator switch on the front of the storage node causes the Locator LEDs on both the front and the back of the node to blink so that you can easily identify a particular node from the front and back of the cabinet. [TABLE 1-2](#) explains the functions and characteristics of the storage node switches and LEDs.

FIGURE 1-4 Storage Node Front Panel Components

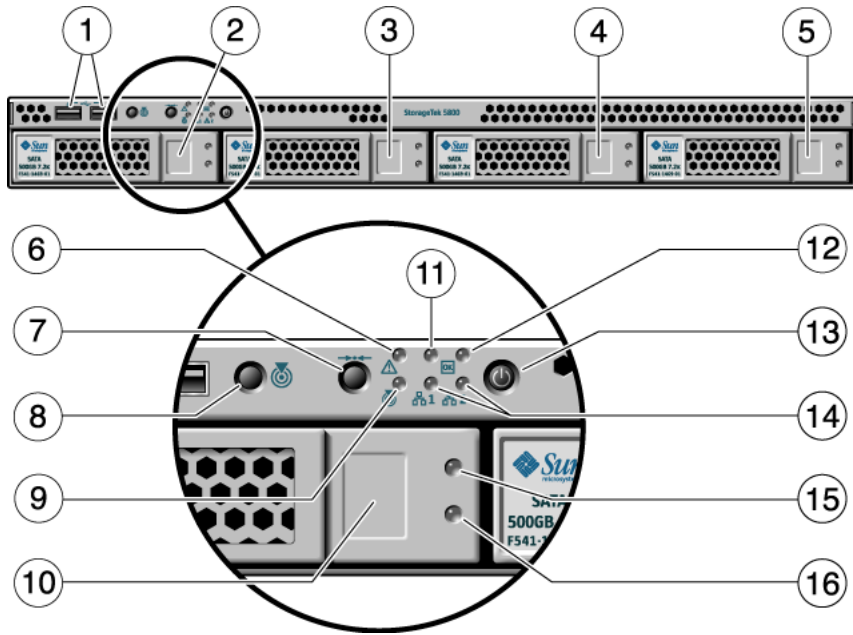


Figure Legend

1	USB ports (not used)	9	Locator LED
2	Drive 0	10	Unlatch level button
3	Drive 1	11	Not used
4	Drive 2	12	Power LED
5	Drive 3	13	Power switch
6	Storage node fault LED	14	Not used
7	Reset switch	15	Drive fault LED
8	Locator switch	16	Drive activity LED

TABLE 1-2 Storage Node Front Panel LED and Switch Descriptions

LED/Switch	Description
Storage node fault LED	Yellow/amber when a fault exists in the system.
Reset switch	Reboots the storage node.
Power LED	Green when the power is on.

TABLE 1-2 Storage Node Front Panel LED and Switch Descriptions (*Continued*)

LED/Switch	Description
Locator LED	White and blinks when you press the locator switch. It also blinks when the service node receives the chassis indicator command to light the LED. Note: There is a corresponding Locator LED on the back of the system.
Drive fault LED	Amber when a drive fault exists. Note: The drive fault LED is activated by the 5800 system software to indicate components in the cell that must be serviced.
Drive activity LED	Solid green when no activity is taking place. When disk access occurs, it momentarily turns off. When access is non-stop, it blinks continuously. When there is no drive present in the bay, it turns off.
Power switch	Provides power to the system.
Locator switch	Causes the Locator LEDs on the front and back to blink, enabling you to locate a storage node in the rear of a fully equipped cabinet.

FIGURE 1-5 shows the back panel components of a 5800 system storage node.

FIGURE 1-5 Storage Node Back Panel Components

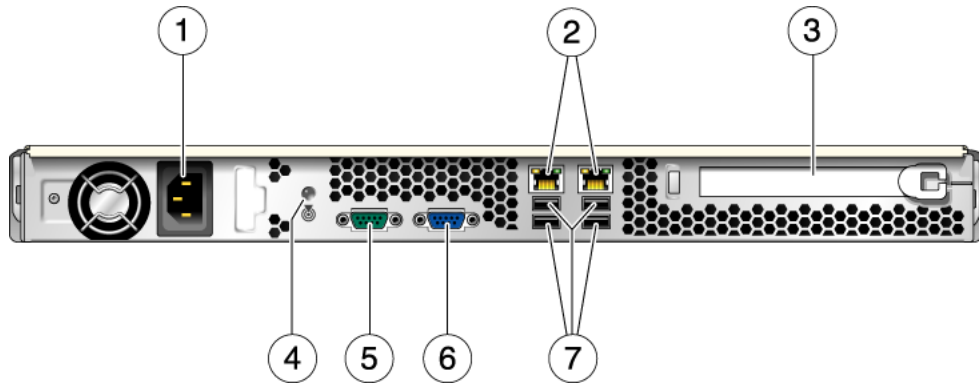


Figure Legend

1	Power connector	5	Serial port
2	Gigabit Ethernet ports	6	VGA port
3	Blank plate	7	USB ports
4	Locator LED		

Service Node

The service node is a Sun Microsystems Sun Fire™ X2100 M2 server with one 250-gigabyte Serial ATA disk drive. The 5800 system uses the service node for initial configuration and troubleshooting, and to upgrade the system software. The system does not use the service node to access the data objects. The key components of a service node are listed in TABLE 1-3.

TABLE 1-3 Service Node Features

Component	Description
CPU	<ul style="list-style-type: none">• One dual-core AMD Opteron processor• Processor frequency: 1.8 GHz• Two 1-MB level 2 cache
Memory	2 GB using four 512MB ECC DIMMs
Media storage	DVD-ROM drive
Hard disk drive	One 250-GB Serial ATA

TABLE 1-3 Service Node Features

Component	Description
Power supply	345W PSU
Network I/O	Four 10/100/1000BASE-T GB Ethernet ports, two Broadcom and two NVidia. (The 5800 system uses the two Broadcom ports.)
System management	IPMI 2.0-compliant service processor module

FIGURE 1-6 shows the front panel of the service node.

FIGURE 1-6 Service Node Front Panel Components

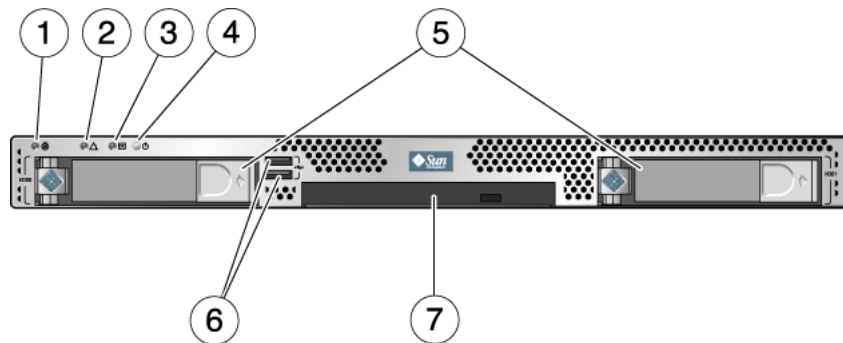


Figure Legend

1	Locator button/LED	5	Hard disk drive bays. The single disk drive is located behind the left drive bay latch.
2	Service indicator LED	6	USB ports
3	Power LED	7	DVD drive
4	Power switch		

FIGURE 1-7 shows the back panel of the service node.

FIGURE 1-7 Service Node Back Panel Components

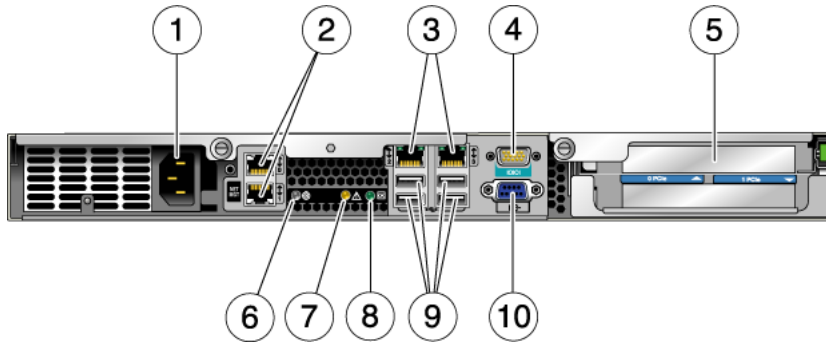


Figure Legend

1	Power connector	6	Locator LED
2	Broadcom Ethernet ports	7	Service indicator LED
3	NVIDIA Ethernet ports	8	Power LED
4	USB ports	9	USB ports
5	Not used	10	Serial port

Gigabit Ethernet Switches

A half- or full-cell 5800 system includes two gigabit Ethernet switches. The switches allow the system to be addressable from a single physical Ethernet connection (with a redundant backup) as two virtual IP (VIP) addresses: one for data and one for administration. The switches also enable load-balancing capabilities for store and retrieve data flows to and from the storage nodes by making use of chipsets that support basic packet header analysis of hash-table based routing information.

FIGURE 1-8 shows the components of the switches. One of the switches is designated as primary and the other as standby. By default, the bottom switch is the active, primary switch, and the top switch is the secondary switch in standby mode. If the primary switch fails, the secondary switch takes control automatically and becomes the primary switch. If the primary switch comes back online, it resumes control.

Storage nodes 1 through 16 are connected to Ethernet ports 1 through 16 of each switch. The service node is connected to port 17 of each switch. The switches are connected to each other for heartbeat communication by ports 23 and 24 of each switch. See **TABLE 1-4** for an explanation of how the primary and secondary switches are connected to the network patch panel.

FIGURE 1-8 Gigabit Ethernet Switch

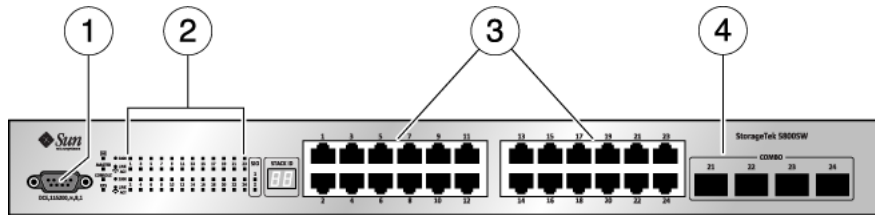


Figure Legend

- 1 Serial port
- 2 Port connection status LEDs
- 3 Ethernet ports
- 4 Not used

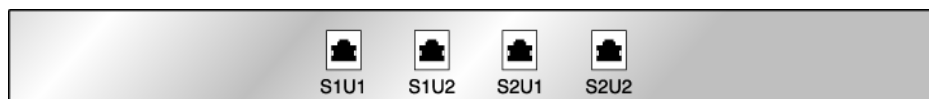
TABLE 1-4 Gigabit Ethernet Switch Connections to the Network Patch Panel

Switch and Ethernet Port	Network Patch Panel Port
Primary - port 21	S1U2 - For Sun service personnel only
Primary - port 22	S1U1
Secondary - port 21	S2U2 - For Sun service personnel only
Secondary - port 22	S2U1

Network Patch Panel

A single network patch panel on the back of the 5800 system provides all the attachment points for the network. [FIGURE 1-9](#) shows the port configuration of the network patch panel.

FIGURE 1-9 Network Patch Panel



The Ethernet ports are designated SxUy, where:

S = Switch
x = Switch number
U = Uplink
y = Port number

S1 is the primary switch and S2 is the secondary. The secondary switch becomes operational when the primary fails. The S1U2 and S2U2 connections are for Sun service personnel only.

5800 System Software

This section describes the 5800 system software. It includes the following topics:

- [“Bundled Software” on page 16](#)
- [“Metadata” on page 17](#)
- [“The Placement Algorithm” on page 17](#)
- [“User Interfaces” on page 18](#)

Bundled Software

You manage and upgrade the 5800 system software as a combined *bundle* rather than as individual components. The software components of the bundle include:

- 5800 system software
- Sun Solaris 10 Operating System (x86 version)
- All Sun Solaris 10 patches
- Basic Input Output System (BIOS)
- Server Management Daughter Card (SMDC) firmware

Even though there are many components, each with their own unique software, BIOS, and firmware, they are upgraded as a whole unit. Even if only one component needs upgrading, the version number of the bundle changes, and you must upgrade the entire bundle.

Metadata

Metadata is information that describes a data object. The 5800 system stores metadata about all data objects in a distributed database. Users can issue queries to search the database and find objects based on the metadata assigned to them. The 5800 system allows two types of metadata: *system* and *extended*.

System Metadata

The 5800 system automatically assigns system metadata to every data object when it is stored on the 5800 system. System metadata includes a unique identifier for each object, called the Object ID or OID. The application programming interface (API) included with the 5800 system can retrieve the object using this OID. System metadata also includes creation time, data length, and data hash.

Extended Metadata

Extended metadata goes beyond the system metadata to further describe each data object. For example, if the data stored on the 5800 system includes medical records, extended metadata attributes might include patient name, date of visit, doctor name, medical record number, and insurance company. Users can issue queries to retrieve data objects using these attributes. For example, a query could retrieve all records (data objects) for a given doctor and a particular insurance company.

The Placement Algorithm

The 5800 system stores data objects across multiple storage nodes and disks using 5+2 encoding. The system can tolerate up to two missing data or parity fragments. After a failure of a disk or a storage node, the system distributes the data and/or parity to other storage nodes and disks. After a rebuild cycle, the system can tolerate another two missing data or parity fragments.

When a data object comes into the system, the gigabit Ethernet switch directs the store request to a storage node, and that node fragments the object and distributes the fragments to different disks in the system. The 5800 system breaks up the data into data and parity chunks. A placement algorithm then decides out of thousands of different layout possibilities where to put the chunks.

User Interfaces

The 5800 system exports two virtual IP (VIP) addresses: one for data processing and one for administrative functions. Your interaction with the system does not require a knowledge of the underlying hardware. Instead, you access it as a single large system.

The 5800 system provides a number of different user interfaces for data processing and management functions, as shown in [TABLE 1-5](#).

TABLE 1-5 5800 Storage System User Interface Functional Capabilities

Interface	Data Processing Capabilities	Administration / Management Capabilities
API	Can perform all store, retrieve, query, and delete functions	None
WebDAV	<ul style="list-style-type: none">• Same capabilities as API except no query function• Can present a virtual directory structure view of data objects according to a defined schema of metadata attributes• Not supported for multi-cell configurations	None
CLI	None, except capability to delete all data and metadata on a full-cell	Can perform most system administration tasks
GUI	None, except capability to delete all data and metadata on a full-cell	Can perform most system administration tasks

Application Program Interface (API)

A Java™ and C language-based application program interface (API) provides the basic store and retrieve commands with additional query semantics.

You perform data processing tasks on the 5800 system using the API or WebDAV, both of which are available through the data VIP address.

The Java and C language APIs enable you to store, retrieve, query, and delete data and metadata through Java and C client libraries. Sample applications and command-line routines that demonstrate the 5800 system's capabilities as well as provide good programming examples are provided in the client software developer's kit (SDK).

The SDK also provides an emulator that can run on the Sun Solaris, Red Hat Enterprise Linux, and Microsoft Windows operating systems. The emulator imitates the behavior of a 5800 system, allowing you to test your software or applications.

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*.

WebDAV

The Web-based Distributed Authoring and Versioning (WebDAV) protocol is a set of extensions to the HTTP/1.1 protocol that allows you to read, add, and delete files on remote web servers. You can set up virtual file system views in the 5800 system that allow you to use WebDAV to browse through data files on the system as though they were stored in a hierarchical path structure.

For example, you could set up a virtual file system view that included medical records as a folder, and included subfolders for a number of different doctors or hospitals. You could then use a web browser to view those files and add and delete new files to the folder and subfolders.

WebDAV is not supported for multi-cell configurations.

CLI and GUI

You perform administrative tasks using either the command-line interface (CLI) or the graphical-user interface (GUI). The remainder of this guide details the administrative tasks that are available and the CLI and GUI procedures to perform each task. For additional information on the GUI, refer to the online Help supplied with the software.

Using the Administrative Interfaces

This chapter describes how to gain administrative access to the 5800 system in order to perform the administrative tasks described in this guide. It includes the following sections:

- [“About Administrative Interfaces” on page 21](#)
 - [“Hive-Level and Cell-Level Functions” on page 22](#)
 - [“Using the CLI” on page 22](#)
 - [“Using the GUI” on page 26](#)
 - [“Multiple Users Accessing the CLI or GUI” on page 33](#)
 - [“Task Overview” on page 34](#)
-

About Administrative Interfaces

You perform administrative tasks on the 5800 system in one of two ways: using a command-line interface (CLI) or using the administrative graphical user interface (GUI). You access the CLI with the `ssh` command, and if you want, you can script CLI commands. You access the GUI using a web browser.

Using either the CLI or the GUI, you can perform administrative tasks such as configuring the system’s IP addresses; specifying which clients are authorized to access data on the system; monitoring individual components including disks and nodes; shutting down, powering down, and rebooting the system; and setting up the schema definitions.

The remaining chapters in this guide provide detailed procedures for using each of these interfaces to perform administrative tasks. For the complete syntax of each of the CLI commands, see the [“Reference: CLI Commands” on page 127](#). For more

information on how to navigate the GUI interface, refer to the online Help supplied with the GUI. For a list of administrative tasks and which interface you can use to perform those tasks, see [“Task Overview” on page 34](#).

Hive-Level and Cell-Level Functions

Most administrative tasks that you perform on the 5800 system affect all cells in a multicell configuration and are therefore considered “hive-level” functions. Examples of hive-level functions include setting the administrative password, specifying which clients are authorized to access the data on the system, and setting up email notifications of system events.

Some administrative tasks affect only one cell in a multicell environment. For example, you specify the administrative IP address, the data IP address, and the default gateway for each cell separately.

For CLI commands that affect only one cell, you must enter the cell ID of the cell you want to administer

To display the cells and cell IDs configured on your system, use the `hiveadm --status` command:

```
ST5800 $ hiveadm --status
There is/are 2 cell(s) in the hive:
- Cell 1: adminVIP = 10.7.224.21, dataVIP = 10.7.224.22
- Cell 5: adminVIP = 10.7.224.101, dataVIP = 10.7.224.102
ST5800 $
```

Using the CLI

This section describes how to access the CLI commands, log out of the CLI system, and execute CLI scripts.

Logging In To the CLI

You log in to the CLI using a secure shell (ssh) connection from any system on the network. A secure shell is included on most UNIX systems, but if you are using another operating system, such as Windows XP, you might have to obtain a third-party ssh client.

Open the secure shell connection to a cell's administrative IP address. (See [“Administrative IP Address” on page 38](#) for more information.)

Note – In a multicell configuration, you must use the administrative IP address of the *master cell*, the cell with the lowest cell ID, to perform administrative tasks on any cell in the system.

The following example shows the command you would enter from a UNIX system to open a secure shell connection to a cell with administrative IP address 10.7.227.101. The default password for the system is admin. (See [“Administrative Password” on page 41](#) for information about changing the password.)

When the connection is open, you will see the ST5800 \$ prompt.

```
% ssh admin@10.7.227.101
password: [admin]
Last login: Wed Feb 7 07:55:32 2007 from 10.7.6.22
Sun StorageTek (TM) 5800 System Management Console
Copyright (C) 2007 Sun Microsystems, Inc.
All rights reserved. Use is subject to license terms.
ST5800 $
```

Logging Out of the CLI

You log out of the CLI by issuing the logout command.

```
ST5800 $ logout
Connection to 10.7.227.101 closed.
%
```

▼ To Create and Execute a Script for CLI Commands

1. Use the `ssh` application on the client system to create public and private key files.

Consult your `ssh` application's documentation for instructions on creating these files. Create the public key without a passphrase. You may be prompted to enter the CLI password (default is `admin`) during key generation.

2. Set the public key on the 5800 system using the `passwd --pubkey` command:

Note – The `--pubkey` option of the `passwd` command is available only in non-interactive mode (that is, you must enter it at the same time that you enter the `ssh` command, as shown in this example). For more information about the `--pubkey` option, see [“The Public Key” on page 42](#).

```
client $ ssh admin@hc1-admin passwd --pubkey < key.pub
```

Substitute the name of the file containing the public key for `key.pub` in the example.

3. Enter the administrative password for the 5800 system.

```
Password: XXXXXXXX
CLI admin: The public key has been changed successfully
client $
```

You can now use `ssh` from your client system to connect to the 5800 system without having to enter a password.

4. Create a script file containing CLI commands using any available editor.

For example, on the client, use any available editor to create the following script file:

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

Substitute the administrative IP address of the cell you want to access for `hc1-admin` in the example.

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

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

6. Run the script.

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

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

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

Getting Help on CLI Commands

Refer to [“Reference: CLI Commands” on page 127](#) for complete syntax and examples of CLI commands. You can also issue the `help` command within the CLI to obtain a list of available CLI commands. Display information on a particular command by issuing the `help command-name` command.

```
ST5800 $ help
Type "help <command>" or "<command> --help" for one of the
commands below
alertcfg      cellcfg      copyright   date
df            help        hiveadm    hivecfg
hwstat        logout      mdconfig   passwd
perfstats     reboot      sensors     shutdown
sysstat       version     wipe
ST5800 $
```

```
ST5800 $ help hwstat
Usage: hwstat [options]
Displays information about the various FRUs in the system. The
component name or FRU ID may be used when specifying the component
to view.

Options:
  -c, --cellid <cellid> Specifies the ID of the cell where the
command should run.
  -f, --FRUID <component> Shows information about component

ST5800 $
```

Using the GUI

This section describes how to launch and log out of the GUI and get help on GUI screens and functions. You launch the GUI either from a web browser or from the command line on any system that is on the same network as the 5800 system. Refer to the *Sun StorageTek 5800 System Release Notes* for information about compatible browsers.

▼ To Launch the GUI from a Browser

Use a web browser from any system on the same network as the 5800 system, as follows:

1. Type **https://** followed by the administrative IP address of the cell and **8090** (the GUI port number) on the browser URL line, as in this example:

```
https://10.7.227.101:8090/
```

Note – In a multicell configuration, you must use the administrative IP address of the *master cell*, the cell with the lowest cell ID, to perform administrative tasks on any cell in the system.

Note – If your web browser does not have the Java plug-in, the system automatically downloads it at this point.

Note – You might see messages about unverified certificates as you start the GUI. Unless you have reason to suspect that another site is impersonating the administrative IP address of the 5800 system, you can ignore these messages.

Note – If the GUI does not start, see [“To Set MIME Associations for the Browser” on page 27](#).

2. Type the administrative password in the Password field.

The default administrative password is `admin`. Passwords are case-sensitive. See [“Administrative Password” on page 41](#) for more information.

3. Click Login.

The main screen of the 5800 system GUI is displayed in [FIGURE 2-1](#).

▼ To Set MIME Associations for the Browser

Your browser must have the correct MIME association to handle `.jnlp` files and invoke Java™ Web Start. Most browsers are configured to handle `.jnlp` files by default. If the GUI does not start, make sure the MIME associations in the browser are configured correctly. You configure MIME associations differently depending on which browser you are using. For example, if you are using a Mozilla browser, you might have to configure the MIME association as follows:

- 1. Select Edit > Preferences from the browser toolbar.**
- 2. Select Navigator > Helper Applications.**
- 3. Click New Type.**
- 4. For Mime Type, enter `application/x-javaws`.**
- 5. For Description, enter something like *Java Web Start*.**
- 6. For Extension, enter `jnlp`.**
- 7. Select Open it with and provide the location of your javaws (for example, `/usr/jdk/instances/jdk1.5.0/bin/javaws`).**

If you don't know where `javaws` is on your system, you might have to search for it.

8. Click OK to close all pop-ups.

▼ To Launch the GUI from the Command Line

Complete the following steps from any system on the same network as the 5800 system, as follows:

1. **Issue the following command from the command line of any system on the same network as the 5800 system:**

```
javaws https://adminIP:8090/admgui.jnlp
```

where *adminIP* is the administrative IP address of the cell. For example, if the administrative IP address of the cell is 10.7.227.101, you enter the following command at the command line:

```
javaws https://10.7.227.101:8090/admgui.jnlp
```

Note – In a multicell configuration, you must use the administrative IP address of the *master cell*, the cell with the lowest cell ID, to perform administrative tasks on any cell in the system.

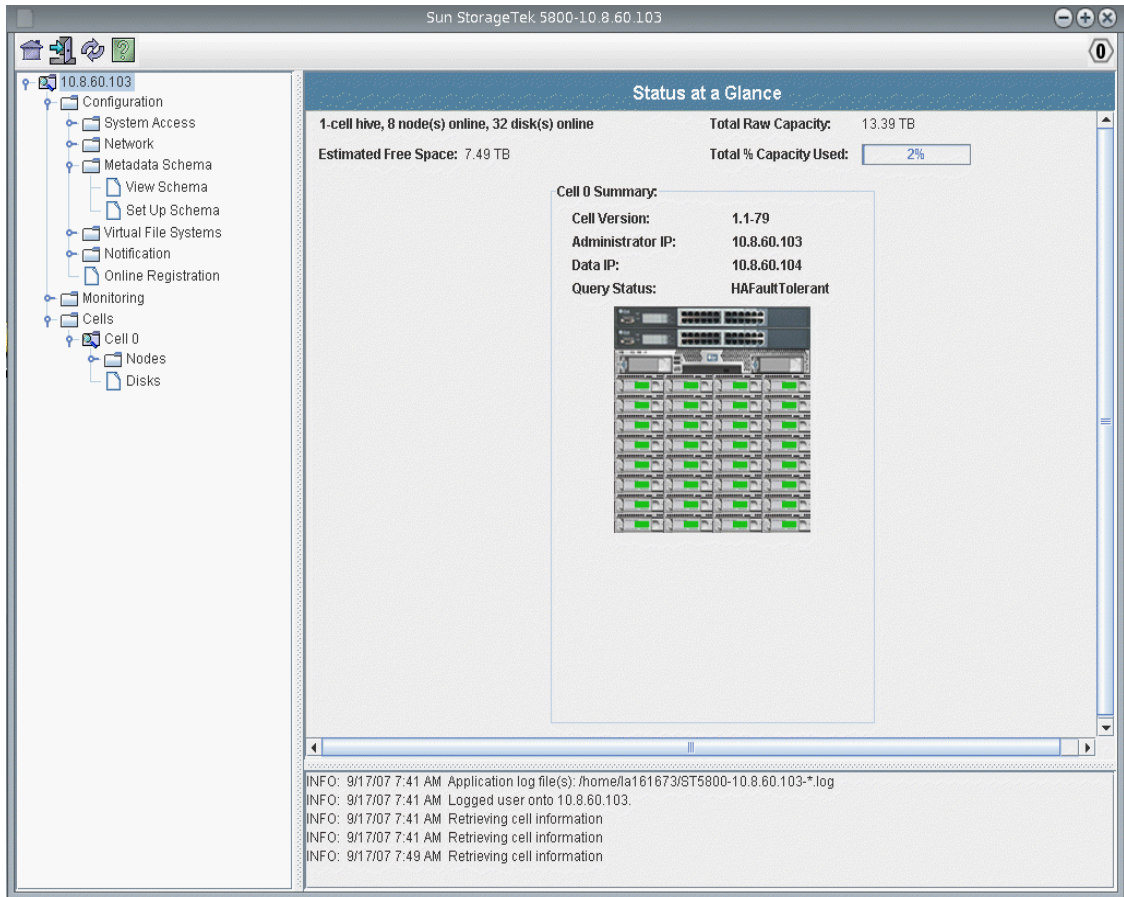
2. **Type the administrative password in the Password field.**

The default administrative password is `admin`. Passwords are case-sensitive. See [“Administrative Password” on page 41](#) for more information.

3. **Click Login.**

The main screen of the 5800 system GUI is displayed in [FIGURE 2-1](#).

FIGURE 2-1 Main GUI Screen



Getting Help on the GUI

To view information about the 5800 system GUI software, click the Help button (shown in [FIGURE 2-2](#)) in the 5800 system GUI toolbar.

FIGURE 2-2 Help Button in Toolbar



The Help panel (shown in [FIGURE 2-3](#)) includes a Navigation pane on the left, and a Topic pane on the right.

To display a Help topic, use the Contents, Index, and Search tabs in the Navigation pane. Click the Search tab and click Tips on Searching to learn about the search feature.

FIGURE 2-3 GUI Help Pane

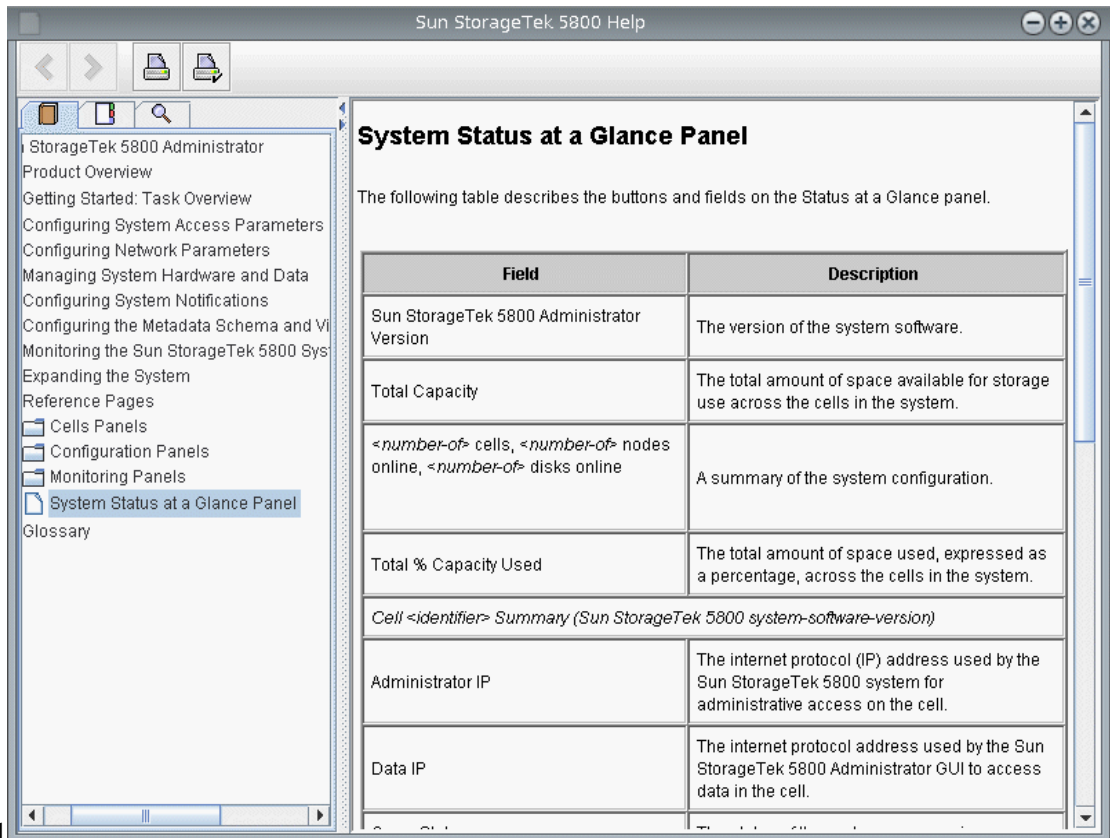








TABLE 2-1 describes the Help tabs.

TABLE 2-1 Help Tabs

Tab	Description
Contents 	Click a folder icon to display subtopics. Click a page icon to display the Help page for that topic in the Topic pane.
Index 	Click an index entry to display the corresponding Help page.
Search 	Type the words for which you want to search and click Search. The Navigation pane displays a list of topics that match your search criteria in order of relevancy. Click a topic link to display the Help page for that topic. Click the Tips on Searching link for information about how to improve your search results. To search for a particular word or phrase within a topic, click in the Topic pane, press Ctrl+F, type the word or phrase for which you are searching, and click Find.

The Help panel icons are described in TABLE 2-2.

TABLE 2-2 Help Icons

Control/Indicator	Description
	Go back to the previous Help topic that you viewed in the current session.
	Go forward to the next Help topic that you viewed in the current session.
	Print the current Help topic.

Logging Out of the GUI

To log out of the 5800 system GUI, click the Log Out button (shown in [FIGURE 2-4](#)) in the 5800 system GUI toolbar.

FIGURE 2-4 Log Out Button in Toolbar



Multiple Users Accessing the CLI or GUI

No more than five users should be simultaneously logged in to the CLI or GUI. The 5800 system allows only one GUI or CLI user at time to perform write operations.

When you select a write operation in the GUI or CLI, the system checks to see if any other CLI or GUI user currently holds write privileges. If not, the system allows you to perform the write operation. You then have write privileges for 15 minutes, or until you log out of the system, whichever occurs first.

Task Overview

TABLE 2-3 lists the administrative tasks available for the 5800 system, whether you can perform the task from the GUI or CLI, and a reference to more information in this guide.

TABLE 2-3 5800 System Administrative Tasks

Task	Available from GUI?	Available from CLI?	See This Section
Configure the administrative IP address	✓	✓	“Administrative IP Address” on page 38
Configure the data IP address	✓	✓	“Data IP Address” on page 39
Configure the service node IP address	✓	✓	“Service Node IP Address” on page 40
Configure the administrative password	✓	✓	“Administrative Password” on page 41
Configure a public key		✓	“The Public Key” on page 42
Configure authorized data subnetworks	✓	✓	“Authorized Subnetworks” on page 44
Configure a gateway	✓	✓	“The Gateway” on page 48
Configure an NTP server	✓	✓	“NTP Servers” on page 49
Check the system time	✓	✓	“NTP Servers” on page 49
Configure DNS	✓	✓	“DNS” on page 50
Register the system	✓		“Registering Using the CLI” on page 64
Start and stop system components	✓	✓	“Starting and Stopping System Components” on page 53
Delete all data from the system	✓	✓	“Deleting All Data From the System” on page 58
Configure email notifications	✓	✓	“Configuring System Notifications” on page 61
Configure the external log host	✓	✓	“Configuring System Notifications” on page 61
Configure the metadata schema	✓	✓	“Configuring Metadata and Virtual File System Views” on page 67

TABLE 2-3 5800 System Administrative Tasks (*Continued*)

Task	Available from GUI?	Available from CLI?	See This Section
Configure virtual file system views	✓	✓	“Configuring Metadata and Virtual File System Views” on page 67
Monitor the system	✓	✓	<ul style="list-style-type: none">• “Monitoring the System Using the CLI” on page 105• “Monitoring the 5800 System Using the GUI” on page 115
Use NDMP to implement disaster recovery	NA	NA	“Implementing Disaster Recovery” on page 121

Configuring System Access Parameters

This chapter describes how to configure system access settings. It contains the following sections:

- [“About System Access” on page 37](#)
- [“Administrative IP Address” on page 38](#)
- [“Data IP Address” on page 39](#)
- [“Service Node IP Address” on page 40](#)
- [“Administrative Password” on page 41](#)
- [“The Public Key” on page 42](#)
- [“Authorized Subnetworks” on page 44](#)

Note – For instructions on accessing the CLI commands and GUI functions described in this chapter, see [“Using the Administrative Interfaces” on page 21](#).

About System Access

The 5800 system exports two IP addresses per cell for system access: one for administrative access and one for data access. To administer and monitor data on each cell in the system, you must configure the administrative and data IP addresses for each cell.

You can also set the service node IP address on each cell, control which clients on the network have access to the data on the system, and change the administrative password and public key.

The following IP addresses are reserved for use by the 5800 system. Do not configure any hosts on your network with these addresses:

- 10.123.0.1
- 10.123.0.2
- 10.123.45.1 through 10.123.45.16

Administrative IP Address

The administrative IP address is the address exported by the 5800 system for administrative access to a cell. You configure the administrative IP address on a per-cell basis.

▼ To Configure the Administrative IP Address Using the CLI

- **Assign an administrative IP address with the command**
`cellcfg --cellid cellid --admin_ip ip_address.`

For example:

```
ST5800 $ cellcfg --cellid 1 --admin_ip 10.7.224.41
```

Note – The system must reboot whenever you change any of the following values: the administrative IP, the data IP, or the service node IP addresses. If you want to change more than one of these values, change them at the same time so that the system doesn't reboot multiple times. Use the format `cellcfg --cellid cellid --admin_ip ip_address --data_ip ip_address --service_node_ip ip_address`. For example:

```
cellcfg --cellid 2 --admin_ip 10.7.224.41 --data_ip 10.7.224.42  
--service_node_ip 10.7.224.40.
```

▼ To Configure the Administrative IP Address Using the GUI

1. From the navigation panel, choose **Configuration > System Access**.

2. Click Configure Cell IPs.
3. Choose the cell for which you want to configure an administrative IP address.
4. Type the administrative IP address in the Administrative IP Address box.
5. (Optional) If you want to configure the data IP address or service node IP address, change those settings now so that the system only reboots once for the changes to take effect.
6. Click Apply.
7. Repeat steps 1–5 for each cell for which you want to configure the administrative IP address.

Data IP Address

The data IP address is the address exported by the 5800 system for access to the data stored on a cell. You configure the data IP address on a per-cell basis.

▼ To Configure the Data IP Address Using the CLI

- **Assign a data IP address with the command**
`cellcfg --cellid cellid --data_ip ip_address.`

For example:

```
ST5800 $ cellcfg --cellid 2 --data_ip 10.7.224.42
```

Note – The system must reboot whenever you change any of the following values: the administrative IP, the data IP, or the service node IP addresses. If you want to change more than one of these values, change them at the same time so that the system doesn't reboot multiple times. Use the format `cellcfg --cellid cellid --admin_ip ip_address --data_ip ip_address --service_node_ip ip_address`. For example:

```
cellcfg --cellid 2 --admin_ip 10.7.224.41 --data_ip 10.7.224.42  
--service_node_ip 10.7.224.40.
```

▼ To Configure the Data IP Address Using the GUI

1. From the navigation panel, choose **Configuration > System Access**.
2. Click **Configure Cell IPs**.
3. Choose the cell for which you want to configure the data IP address.
4. Type the data IP address in the **Data IP Address** box.
5. (Optional) If you want to configure the administrative IP address or service node IP address, change those settings now so that the system only reboots once for the changes to take effect.
6. Click **Apply**.
7. Repeat steps 1–5 for each cell for which you want to configure the data IP address.

Service Node IP Address

The service node IP address is the address of the server for the cell. The 5800 system uses the service node for configuration, troubleshooting, and upgrading the system software. If there is a conflict between the default IP address on the service node (10.7.227.100) and an address on your network, you can set a new address for the service node. You configure the service node IP address on a per-cell basis.

▼ To Configure the Service Node IP Address Using the CLI

- **Assign the service node IP with the command**
`cellcfg --cellid cellid --service_node_ip ip_address.`
For example:

```
ST5800 $ cellcfg --cellid 2 --service_node_ip 10.7.224.40
```

Note – The system must reboot whenever you change any of the following values: the administrative IP, the data IP, or the service node IP addresses. If you want to change more than one of these values, change them at the same time so that the system doesn't reboot multiple times. Use the format `cellcfg --cellid cellid --admin_ip ip_address --data_ip ip_address --service_node_ip ip_address`. For example:

```
cellcfg --cellid 2 --admin_ip 10.7.224.41 data_ip 10.7.224.42 -  
-service_node_ip 10.7.224.40.
```

▼ To Configure the Service Node IP Address Using the GUI

1. From the navigation panel, choose **Configuration > System Access**.
2. Click **Configure Cell IPs**.
3. Choose the cell for which you want to configure the service node IP address.
4. Type the service node IP address in the **Service Node IP Address** box.
5. (Optional) If you want to configure the administrative IP address or data IP address, change those settings now so that the system only reboots once for the changes to take effect.
6. Click **Apply**.
7. Repeat steps 1–5 for each cell for which you want to configure a service node IP address.

Administrative Password

The administrative password allows you to access the 5800 system CLI commands and also to perform configuration and administrative tasks using the GUI. The default password is `admin`. Passwords are case-sensitive.

You set the administrative password on a per-hive basis.

▼ To Configure the Administrative Password Using the CLI

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

For example:

```
ST5800 $ passwd
Enter current password:XXXXX
Enter new password:XXXXXX
Re-enter new password:XXXXXX
CLI admin: The admin password has been changed successfully.
ST5800 $
```

▼ To Configure the Administrative Password Using the GUI

1. From the navigation panel, choose **Configuration > System Access**.
2. Click **Change Admin Password**.
3. Type the current password in the **Current Password** field. (If there is no current password, leave this field blank.)
4. Type the password you would like to use in the **New Password** field.
5. Type the new password again in the **Reenter Password** field.
6. Click **Apply**.

The Public Key

A public key allows you to log in to the 5800 system from client systems carrying the private version of the key without using a password. You might want to use this feature so that you can execute scripts of CLI commands from a specific client. See [“To Create and Execute a Script for CLI Commands” on page 24](#) for more information about scripting CLI commands.

Use the ssh application on your client to create public and private key files. (See the documentation for your ssh application for information about creating these files.) Create the public key without a passphrase.

Once you have configured the public key file on the 5800 system, you can log in from any client with the private version of that key without being prompted for a password. If you wish to return to interactive logins, remove the private key from the client, or configure a new public key on the 5800 system.

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

Note – The `--pubkey` option is available only in non-interactive mode (that is, you must enter it at the same time that you enter the ssh command, as in **ssh admin@10.7.227.101 passwd --pubkey < key.pub**).

You set the public key on a per-hive basis.

▼ To Configure a Public Key Using the CLI

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

For example:

```
client $ ssh admin@admin_IP passwd --pubkey < key.pub
Password:
```

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

2. **Enter the administrative password for the 5800 system.**

For example:

```
Password: XXXXXX
CLI admin: The public key has been changed successfully
client $
```

3. Verify password-free login.

For example:

```
client $ ssh admin@admin_IP
Sun StorageTek (TM) 5800 System Management Console
Copyright (C) 2007 Sun Microsystems, Inc.
All rights reserved. Use is subject to license terms.
ST5800 $
```

Configuring a Public Key Using the GUI

This function is not available from the GUI.

Authorized Subnetworks

By default, the system allows any client on the network to access the data stored on the 5800 system. Using the authorized subnetworks feature, you can control which clients can access the data by specifying a list of authorized subnetworks. Only clients running on the authorized subnetworks can access the data stored on the 5800 system. You set the authorized subnetworks on a per-hive basis.

For example, if you specify an authorized subnetwork of 192.37.54.0/24, all clients running on the 192.37.54.0/24 subnetwork will be allowed to access data on the 5800 system. You may specify a single client as an authorized “subnetwork” consisting of one host. For example, to allow the client with IP address 172.168.20.35 to access the system, specify 172.168.20.35 as an authorized subnetwork.

To ensure optimal performance, there are limitations on the number of authorized subnetworks you can specify. The maximum number of authorized subnetworks allowed is five.

Note – If your configuration requires that you specify more than five authorized subnetworks, consult Sun services for assistance.

▼ To Configure Authorized Subnetworks Using the CLI

1. **Configure authorized subnetworks using the command**
`hivecfg --authorized_clients ip_addresses.`

For example:

```
ST5800 $ hivecfg --authorized_clients  
10.45.1.19,192.18.7.2,34.7.7.101
```


Note – If DNS is enabled on your system, you may specify host names instead of IP addresses.

2. **To reset this property and allow all clients to access data, use the command**
`hivecfg --authorized_clients all.`

For example:

```
ST5800 $ hivecfg --authorized_clients all
```

▼ To Configure Authorized Subnetworks Using the GUI

1. From the navigation panel, choose **Configuration > System Access**.
2. Click **Authorize Data Clients**.
3. Click the **Add** button .
A new row is displayed in the table.
4. (Optional) If DNS is enabled and you want to enter host names instead of IP addresses, select **Host Name**.
5. Type the host name or Internet Protocol (IP) address and subnet mask for the subnetwork.
6. Click **Apply**.
7. Repeat Steps 3–6 for each client that you want to authorize.

Configuring Network Parameters

This chapter describes how to configure system parameters. It contains the following sections:

- [“About Network Parameters” on page 47](#)
- [“The Gateway” on page 48](#)
- [“NTP Servers” on page 49](#)
- [“DNS” on page 50](#)

Note – For instructions on accessing the CLI commands and GUI functions described in this chapter, see [“Using the Administrative Interfaces” on page 21](#).

About Network Parameters

You must configure a gateway from the 5800 system to the network so that the 5800 system is available on the network. In addition, you must configure Network Time Protocol (NTP) servers to ensure that the computer clocks in the system are coordinated.

You can enable DNS for the system, although this is not required.

The Gateway

The gateway is the router that connects the local subnet on which the 5800 system resides to the larger network. You must configure a default gateway for each 5800 system cell, to enable information about the system to be available on the network. You configure the gateway on a per-cell basis.



Caution – The 5800 system will not boot up correctly if the gateway address that you configure is not a valid IP address on the same network as the 5800 system service node IP address. (See [“Service Node IP Address”](#) on page 40.) The system can boot up if the gateway you configure is down, or even if the gateway is not an actual “live” machine, but the system cannot boot if the gateway IP address is invalid or is not reachable from the service node.

▼ To Configure a Gateway Using the CLI

- **Configure the default gateway using the command**

```
cellcfg --cellid cellid --gateway ip_address --subnet subnet_mask.
```

For example:

```
ST5800 $ cellcfg --cellid 1 --gateway 10.7.227.254 --subnet 255.255.255.0
```

▼ To Configure a Gateway Using the GUI

1. From the navigation panel, choose **Configuration > Network**.
2. Click **Configure Gateway**.
3. From the **Cell** drop-down menu, choose the cell for which you want to configure the gateway.
4. Type the gateway address.
5. Type the subnet mask address.
6. Click **Apply**.
7. Repeat steps 1–6 for each cell that needs a network gateway configured.

NTP Servers

A Network Time Protocol (NTP) server synchronizes the clocks on systems on the network and ensures that the timestamps indicating when data is stored or deleted are accurate.

For 5800 system operation, you must specify at least one external NTP server. After you configure the NTP servers, you can check to see the current system time.

You configure NTP servers on a per-hive basis.



▼ To Configure an NTP Server Using the CLI

- To configure the external NTP servers, use the command `hivecfg --ntp_server ip_addresses`.

For example:

```
ST5800 $ hivecfg --ntp_server 10.7.224.30,10.7.224.40
```

▼ To Configure an NTP Server Using the GUI

1. From the navigation panel, choose **Configuration > Network**.
2. Click **Set NTP Server**.
3. To add a new NTP server IP address or host name:
 - a. Click the **Add** button  .
A new row is displayed in the table.
 - b. (Optional) If **Domain Name Service (DNS)** is enabled and you want to enter host names instead of IP addresses, select **Host Name**.
 - c. Type the NTP server host name or IP address in the newly added text fields of the table.
4. To remove an NTP server, select the host name or IP address in the table and click the **Remove** button  .
5. Click **Apply**.

▼ To Check the System Time Using the CLI

- Check the system time using the command `date`.

For example:

```
ST5800 $ date
Thu Jun 28 12:43:17 UTC 2007
```

▼ To Check the System Time Using the GUI

To check the time and date for a cell:

- From the navigation panel, choose **Cells > Cell <identifier>**.

The Cell Summary panel is displayed, listing the System Time.

DNS

The Domain Name Service (DNS) is a service that the 5800 system may use to translate names (such as the NTP server name) to Internet Protocol (IP) addresses.

You configure DNS on a per-hive basis.

▼ To Configure DNS Using the CLI

- Use the command `hivecfg --set` to set the DNS parameters, as follows:
 - **DNS:** `y` enables DNS, `n` disables DNS
 - **Domain Name:** DNS domain name to be used by the 5800 system
 - **DNS Search:** Other domains the system should search if the specified domain name does not result in a valid IP address.
 - **Primary DNS Server:** The server that the system should use first to translate domain names.


- **Secondary DNS Server:** The server that the system should use to translate domain names if the primary server is unavailable.

For example:

```
ST5800 $ hivecfg --set
Enter new value, or hit <enter> to leave the value unchanged:
[multiple values need to be comma separated]

NTP Server [129.145.155.32,129.146.17.39]:
SMTP Server [129.147.62.198]:
SMTP Port [25]:
Authorized Clients [all]:
External Logger [10.7.224.10]:
DNS [y or n] [y]: y
Domain Name [sfbay.sun.com]: sfran.sun.com
DNS Search [sfbay.sun.com,sun.com,eng.sun.com]: sfran.sun.com,sun.com,eng.sun.com
Primary DNS Server [10.8.11.110]: 10.8.11.110
Secondary DNS Server [10.8.11.82]: 10.8.11.82
```

▼ To Configure DNS Using the GUI

1. From the navigation panel, choose Configuration > Network.
2. Click Set Up DNS.
3. Select the Enable DNS check box.
4. In the Domain Name field, type the domain name to be used by the system.
5. In the Primary Server field, type the IP address of the first server the system should use to translate domain names.
6. In the Secondary Server field, type the IP address of the backup server the system should use if the primary server is unable to translate.
7. Click the Add button  to add DNS suffixes to the DNS Suffix Search List. These are additional domains the system should search if the specified Domain Name does not result in a valid IP address.
8. Click Apply.

Managing System Hardware and Data

This chapter describes how to start and stop hives, cells, and nodes, as well as how to remove all data from the 5800 system. It contains the following sections:

- [“Starting and Stopping System Components” on page 53](#)
- [“System Performance and Capacity Impact after Disks or Nodes are Offline” on page 56](#)
- [“Recovering From a Power Failure” on page 57](#)
- [“Deleting All Data From the System” on page 58](#)

Note – For instructions on accessing the CLI commands and GUI functions described in this chapter, see [“Using the Administrative Interfaces” on page 21](#).

Starting and Stopping System Components

To perform administrative actions on the hardware, you might need to shut down or reboot cells.

Caution – For best results, before shutting down or rebooting a cell, be sure that any applications storing or retrieving data to or from that cell are also shut down until you have completed the maintenance action on the cell.

Caution – After you reboot a cell, make sure that the query engine status, as reported by the `sysstat` command is `HAFaultTolerant` before resuming applications that are storing or retrieving data to or from the cell. See “[sysstat](#)” on [page 157](#) for more information on the `sysstat` command.

▼ To Shut Down a Cell Using the CLI

- **Shut down a cell with the command** `shutdown --cellid cellid`.

For example:

```
ST5800 $ shutdown --cellid 1
shutdown? [y/N]: n
ST5800 $ shutdown --cellid 1
shutdown? [y/N]: y
Connection to hcl-admin closed.
```

Note – If you want to completely power down a cell (for example, so that you can move a rack), issue the `shutdown --all` command, which shuts down the service node as well as all the storage nodes in the system. Then, switch all power switches on the front of the rack to the off or 0 position.

▼ To Shut Down a Cell Using the GUI

1. **From the navigational panel, choose Cells > Cell <*identifier*>**.
The Cell Summary panel is displayed.
2. **From the Cell Operations drop-down list box, choose Shutdown Cell.**
3. **Click Apply.**
A confirmation message asks if you want to continue with shutting down the cell and if you want to shutdown the service node as part of the shutdown process.
4. **Select the Shutdown service node checkbox to shut down the service node as part of the shutdown process.**
5. **Click Yes to begin the shutdown process.**

▼ To Reboot a Cell Using the CLI

- **Reboot a cell with the command** `reboot --cellid cellid`.

For example:

```
ST5800 $ reboot --cellid 1
Reboot? [y/N]: n
ST5800 $ reboot
Reboot? [y/N]: y
Connection to hcl-admin closed.
```

Note – If you want to reboot the switches and service node along with the storage nodes on the cell, issue the `reboot cellid cellid --all` command.

▼ To Reboot a Cell Using the GUI

1. **From the navigational panel, choose** `Cells > Cell <identifier>`.

The Cell Summary panel is displayed.

2. **From the Cell Operations drop-down list box, choose** `Reboot Cell`.

3. **Click Apply.**

A confirmation message asks if you want to continue with rebooting the cell and if you want to reboot the service node and switches as part of the reboot process.

4. **Select the Reboot service node and switches checkbox to reboot the service node and switches as part of the reboot process.**

5. **Click Yes to begin the reboot process.**

▼ To Power Up a Cell

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 the 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 5800 system is operational using the `hwstat` and `sysstat` commands. (For more information, see “`hwstat`” on page 142 and “`sysstat`” on page 157.)

System Performance and Capacity Impact after Disks or Nodes are Offline

The 5800 system includes extensive healing capabilities that allow the system to recover from failed disks or nodes. This healing activity may affect system performance and capacity, as described in this section.

If disks fail and are replaced, or nodes go offline and then back online, you may notice that the amount of space utilized on the system changes. (Use the `df` command to display space utilization on the system.)

If a disk goes offline, or if a disk that was previously offline comes back online, the resulting healing activity will affect performance of input and output operations to the 5800 system. Performance of these operations may decrease by approximately 30% for the duration of a healing cycle. The `sysstat` command displays the status of a healing cycle as the Data Reliability Check. For example:

```
ST5800 $ sysstat
Cell 0: Online. Estimated Free Space: 7T
8 nodes online, 32 disks online.
Data VIP 10.8.60.104, Admin VIP 10.8.60.103
Data services Online, Query Engine Status: HAFaultTolerant
Data Integrity check last completed at Fri Aug 03 19:51:50 UTC 2007
Data Reliability check last completed at Tue Aug 07 07:52:45 UTC 2007
Query Integrity check last completed at Tue Aug 07 07:52:45 UTC 2007
NDMP status: Backup ready.
ST5800 $
```

If a disk fails or is replaced, a healing cycle is completed when the last completed date for the Data Reliability Check reflects a date and time after the failure or replacement occurred.

The healing cycle can take 12 hours for a single disk failure or up to 36 hours for a node failure. During this period, the system is less fault tolerant than usual. While normally the system can sustain simultaneous failures of any two disks without losing data, during a healing cycle, the system can tolerate the failure of only one additional disk (other than the one for which the system is healing).

If two or more additional disks fail while the system is in the healing cycle caused by the original disk failure, some data may be lost. (The likelihood of so many failures within such a short time period is extremely low, however.)

Note – For best performance, avoid taking disks or nodes offline during a healing cycle as it may create the appearance of data loss.

Recovering From a Power Failure

When power is restored after a power failure, the 5800 system becomes operational automatically without administrator intervention.

Note – You might have to push the power button on the service node to resume power to that node.

It takes approximately two hours from the time that power is restored until the disks come back online and data services are available. Use the `hwstat` command to verify that all nodes and disks are online. See [“hwstat” on page 142](#) for more information about the `hwstat` command.

After the disks are back online, the query engine is repopulated, which requires a minimum of 12 hours. During repopulation, queries to the data stored on the system might return incomplete results. When the `sysstat` command returns a status of `Query Integrity Established`, you can be sure that queries are now returning complete results. (See [“sysstat” on page 157](#) for more information about the `sysstat` command.)

Data Availability After Power Loss

No data loss should occur as a result of a power failure. Any client store operations that were in progress at the time of power failure will have failed, but any stored data for which the client received an OID remains securely stored on the 5800 system.

In very rare instances, however, individual fragments of stored objects may become unavailable after the system recovers from the power failure. If three fragments of the same object become unavailable, the system will return an `ArchiveException`

“Error opening fragments for oid” error when a client tries to retrieve the object. In this case, contact Sun service for assistance in restoring the object that has become unavailable.

To determine if any objects have become unavailable as a result of the power loss, wait approximately 12 hours after power is restored, and issue the `sysstat` command to see if the Data Reliability Check has completed. If the Data Reliability Check is listed as *not completed since boot*, wait a few more hours and check `sysstat` again.

When `sysstat` indicates that the Data Reliability Check has completed, check the external log messages for `RecoverLostFragments` warnings and errors such as the following:

```
Sep  4 21:24:37 10.7.224.101 java: [local1.warning] java[1228]:
[ID 702911 local1.warning] 286 EXT_WARNING
[MgmtServer.monitorDataDoctor] (296.1) Healing Task
RecoverLostFragments completed with 10 errors: This may indicate a
potential serious problem and should be escalated to a Service
Technician.
```

If you see an error of this type, wait approximately 12 more hours for another healing cycle to be completed. (To determine when a healing cycle has completed, issue the `sysstat` command and check the timestamp for Data Reliability Check.) Then, check the log messages again for `RecoverLostFragments` warnings or errors issued around the time that the most recent Data Reliability Check was completed.

If the system consistently issues `RecoverLostFragments` errors and warnings at the end of every healing cycle, contact Sun service, since there may be some risk of data being unavailable.

Deleting All Data From the System

You can delete (“wipe”) all data stored on a 5800 system hive. When you perform the wipe operation, all user data is destroyed. The system resets the metadata schema file to the original factory settings, while other settings (such as network settings and passwords) are unaffected.

Note – The option to wipe data from a single cell is not available in a multicell configuration; in a multicell configuration, you must wipe data from all cells simultaneously.

Caution – When you wipe data from the system, the metadata schema file is also reset back to the original factory settings. If you want to save your metadata schema file, be sure to back it up before wiping the data.

▼ To Delete All Data Using the CLI

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

For example:

```
ST5800 $ wipe
Destroy all data and clear the metadata schema? [y/N]: y
```

▼ To Delete All Data Using the GUI

1. **From the navigational panel, choose `Cells > Cell <identifier>`.**
The Cell Summary panel is displayed.
2. **From the Cell Operations drop-down list box, choose `Wipe Cell (or Wipe All Cells for a multi-cell system)`.**
3. **Click `Apply`.**
A confirmation message asks if you want to continue with removing data and metadata from all cells.
4. **Click `Yes` to begin the wipe process.**

Note – In a multicell configuration, you cannot wipe data from a single cell; you must wipe all cells simultaneously.

Configuring System Notifications and Online Registration

This chapter describes how to configure system notifications. It contains the following sections:

- [“Configuring System Notifications” on page 61](#)
- [“Registering Online” on page 64](#)

Note – For instructions on accessing the CLI commands and GUI functions described in this chapter, see [“Using the Administrative Interfaces” on page 21](#).

Configuring System Notifications

If you configure email notifications, the 5800 system sends email alerts for conditions such as the following:

- Enabling or disabling of a disk or node
- System reboot
- System shutdown
- Administrator password change
- Switch failover
- Node failure
- Disk added or removed
- System has reached capacity (any disk has reached 80% raw utilization) and cannot accept new objects. (See [“Obtaining Disk Status” on page 111](#) for information about using the `df -p` command to monitor raw utilization on the disks.)

- Data or administrative IP address change
- System “wiped” of all data
- System upgrade
- Schema changes

To configure the 5800 system to send email notifications, you must specify a Simple Mail Transfer Protocol (SMTP) server and the email address of the recipient.

You can also specify an external logging host to which the 5800 system sends detailed log messages for debugging purposes.

You configure the email notifications and the external logging host on a per-hive basis.

Note – Timestamps on the email notifications are always reported in UTC. Timestamps on log messages in the external logging host reflect whatever time zone the system clock on external host is using.

▼ To Configure Email Notifications Using the CLI

1. **Specify the SMTP server and port number with the command**

```
hivecfg --smtp_server ip_address --smtp_port ip_address.
```

For example:

```
ST5800 $ hivecfg --smtp_server 10.7.224.15 --smtp_port 25
```



Note – The SMTP port number is not a required entry.

2. **To add or delete an email address (either To: or cc:) from the specified list type use the commands `alertcfg add to addresses` or `alertcfg del cc addresses`.**

For example:

```
ST5800 $ alertcfg add to alerts@samplecompany.com  
ST5800 $ alertcfg del cc admin@samplecompany.com
```

▼ To Configure Email Notifications Using the GUI

1. From the navigation panel, choose **Configuration > Notification**.
2. Click **Set Up Email**.
3. (Optional) If DNS is enabled and you want to enter host names instead of IP addresses, select **Host Name**.
4. Type the IP address or host name of the SMTP server that you want the 5800 system to use.
5. Type the port number of the SMTP server.
6. Perform one of the following:
 - To add an email recipient:
 - i. Click the **Add button**  .
A new row is displayed in the table.
 - ii. In the **Type** field, choose the type of recipient that you want: **To** or **Cc**.
 - iii. In the **Address** field, type the full email address of the recipient.
 - To remove an email recipient:
 - i. Click the **Remove button**  .
 - ii. At the prompt, confirm the removal.
7. Click **Apply**.

▼ To Configure an External Log Host Using the CLI

Note – In order to receive log messages, the external logging host must be configured so that syslog is set to “listen” to the network. The procedure to configure syslog to listen to the network differs depending on which operating system is running on the external logging host. On systems running Solaris 10, syslog is set to listen to the network by default.

- To set an external log host for the 5800 system, use the command `hivecfg --external_logger ip_address`.

For example:

```
ST5800 $ hivecfg --external_logger 10.7.224.15
```

▼ To Configure an External Log Host Using the GUI

1. From the navigation panel, choose **Configuration > Notification**.
2. Click **Set Logging Host**.
3. (Optional) If DNS is enabled and you want to enter host names instead of IP addresses, select **Host Name**.
4. Type the IP address or host name of the host to which the email notifications log will be sent.
5. Click **Apply**.

Registering Online

You can register information about you and your 5800 system with Sun Microsystems. (For this version of the 5800 system, registration just involves sending Sun your user information; Sun does no monitoring or transfer of telemetry.)

Registering Using the CLI

This function is not available from the CLI.

▼ To Register Using the GUI

1. From the navigation panel, choose **Configuration > Online Registration**.
2. The **Online Registration** panel is displayed.

3. Read the disclaimer and select the Agree check box.
4. In the Sun Account section, type your Sun account number and password.
5. If you do not know your account number and password, click the link at the bottom of the panel.
6. Click the Proxy Server tab and type your proxy server name, port, user name, and user name password.

Note – For registration to be successful, the authorized client setting must be at its default (“all” indicating that all clients can access the system), or the proxy server specified must be included in the authorized client list. See [“Authorized Subnetworks” on page 44](#) for more information about authorized clients.

7. Click Apply.

Configuring Metadata and Virtual File System Views

This chapter describes how to modify the default schema file to add metadata specific to your applications. This chapter also describes how to modify the default schema file to configure virtual file system views that allow users to browse data objects as though they were stored in a traditional hierarchical file structure.

This chapter contains the following sections:

- [“The Metadata Schema” on page 67](#)
- [“Virtual File System Views” on page 87](#)
- [“Summary of Metadata Schema Elements” on page 94](#)
- [“Configuring the Metadata Schema Using the CLI” on page 95](#)
- [“Configuring the Metadata Schema Using the GUI” on page 96](#)
- [“Configuring Virtual File System Views Using the GUI” on page 101](#)

Note – For instructions on accessing the CLI commands and GUI functions described in this chapter, see [“Using the Administrative Interfaces” on page 21](#).

The Metadata Schema

The metadata schema specifies the metadata attributes that can be stored with objects in the 5800 system. The system comes preconfigured with a default metadata schema, which you can modify to specify metadata appropriate to your applications.

The following sections describe the metadata schema file and its components.

Metadata Schema File

You specify what metadata the data objects in your system include and how that metadata is structured using the schema file. You also configure virtual views using the schema file. A predefined schema file, which contains the minimum set of attributes, is included with the 5800 system. You modify that schema file to add the extended metadata and file system views applicable to your configuration.

Schema File Structure

The schema file for the 5800 system is a standard XML file with the general format shown in [CODE EXAMPLE 7-1](#). See [CODE EXAMPLE 7-2](#) for an example of a schema file.

CODE EXAMPLE 7-1 General Structure of Schema File

```
<?xml version="1.0" encoding="UTF-8"?>
<metadataConfig>
    <schema>
        Schema definition
    </schema>
    <fsViews>
        File system views specification
    </fsViews>
    <tables>
        Tables definition
    </tables>
</metadataConfig>
```

Schema File DTD

The Document Type Definition (DTD), which defines the structure of a the schema file, is shown in [CODE EXAMPLE 7-2](#).

CODE EXAMPLE 7-2 DTD for Schema File

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Sun StorageTek 5800 Metadata Configuration Rules. (Sun
Microsystems, Inc.) -->
<!ELEMENT metadataConfig (schema, fsViews?, tables?)>
<!ELEMENT schema (namespace*, field*)>
<!ELEMENT namespace (namespace*, field*)>
<!ELEMENT field EMPTY>
<!ELEMENT fsViews (fsView*)>
<!ELEMENT fsView (attribute+)>
<!ELEMENT attribute EMPTY>
<!ELEMENT tables (table*)>
<!ELEMENT table (column+)>
<!ELEMENT column EMPTY>
<!ATTLIST namespace
  name CDATA #REQUIRED
  writable (true | false) "true"
  extensible (true | false) "true"
>
<!ATTLIST field
  name CDATA #REQUIRED
  type (long | double | string | char | binary | date | time |
timestamp | objectid) #REQUIRED
  length CDATA #IMPLIED
  queryable (true | false) "true"
>
<!ATTLIST fsView
  name CDATA #REQUIRED
  filename CDATA #REQUIRED
  namespace CDATA #IMPLIED
  readonly (true | false) "false"
  filesonlyatleaflevel (true | false) "true"
  fsattrs (true | false) "false"
>
<!ATTLIST attribute
  name CDATA #REQUIRED
>
<!ATTLIST table
  name CDATA #REQUIRED
>
<!ATTLIST column
  name CDATA #REQUIRED
```

Example Schema File

[CODE EXAMPLE 7-3](#) shows an example of a schema file for a system storing MP3 music files.

CODE EXAMPLE 7-3 Example Schema File

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Example of schema configuration file for a system storing MP3
music files.-->
<metadataConfig>
  <schema>
    <namespace name="mp3" writable="true" extensible="true">
      <field name="artist" type="string" length="128"
        queryable="true" />
      <field name="album" type="string" length="128"
        queryable="true" />
      <field name="title" type="string" length="128"
        queryable="true" />
      <field name="type" type="string" length="128"
        queryable="true" />
      <field name="year" type="long" queryable="true"/>
    </namespace>
  </schema>
  <fsViews>
    <fsView name="byArtist" namespace="mp3"
      filename="{title}.{type}" fsattrs="true"
      filesonlyatleaflevel="true">
      <attribute name="artist"/>
      <attribute name="album"/>
    </fsView>
    <fsView name="byAlbum" filename="{mp3.title}.mp3"
      readonly="true" fsattrs="true">
      <attribute name="mp3.album"/>
    </fsView>
  </fsViews>
  <tables>
    <table name="mp3">
      <column name="mp3.artist"/>
      <column name="mp3.album"/>
      <column name="mp3.title"/>
      <column name="mp3.year"/>
    </table>
  </tables>
</metadataConfig>
```

Metadata

Metadata is information that describes a data object. The 5800 system stores metadata about all data objects in a distributed database. Users can issue queries to search the database and find objects based on the metadata assigned to them. The 5800 system allows two types of metadata: system and extended.

System Metadata

The 5800 system automatically assigns system metadata to every data object when it is stored on the 5800 system. System metadata includes a unique identifier for each object, called the Object ID or OID. The application programming interface (API) included with the 5800 system can retrieve the object using this OID. System metadata also includes creation time, data length, and data hash.

Extended Metadata

Extended metadata goes beyond the system metadata to further describe each data object. For example, if the data stored on the 5800 system includes medical records, extended metadata attributes might include patient name, date of visit, doctor name, medical record number, and insurance company. Users can issue queries to retrieve data objects using these attributes. For example, a query could retrieve all records (data objects) for a given doctor and a particular insurance company.

Metadata Types

The 5800 system supports metadata as sets of typed name-value pairs. [TABLE 7-1](#) lists the supported metadata types.

TABLE 7-1 Supported Metadata Types

Valid Types	Description
Long	64 bits Maximum Value: -9223372036854775808 Minimum Value: 9223372036854775807
Double	64 bits Maximum Value: 1.7976931348623157E308 Minimum Positive Value: 4.9E-324
String	A string of characters from the Basic Multilingual Plane of unicode values, excluding the null character (0). Characters from the range of Unicode Surrogates (D800-DFFF) are not supported. Length can be 0 to 4000 unicode characters.
Char	A string of eight-bit characters in the ISO-8859-1 (Latin-1) character set, excluding the null character (0). Length can be 0 to 8000 Latin-1 characters.
Binary	A string of bytes from the range 00 to FF. Length can be 0 to 8000 bytes.
Date	Corresponds to JDBC™ SQL DATE type. Year/Month/Day
Time	Corresponds to JDBC SQL TIME type with precision 0 (seconds past midnight)
Timestamp	Corresponds to JDBC SQL TIMESTAMP type with precision 3 (absolute Year/Month/Day/Hour/Minute/Second/Millisecond)
ObjectID	Binary that specifies that OID of the data

Namespaces

You can group metadata into *namespaces*, or collections of metadata names, identified by a string. Namespaces are essentially directories of metadata names. Just as directories may include subdirectories, namespaces may include *subnamespaces* or namespaces within namespaces. You can have as many namespaces as desired in the 5800 system metadata schema. There is also no limit on the number of subnamespaces within a given namespace.

The full name of an attribute is the name of its namespace, followed by a dot, followed by the attribute name. For example, the attribute name `yoyodyne.widget.oscillation.overthrunder` represents an attribute whose

name is `overthruster`, which is grouped within the subnamespace `oscillation`, which is part of the subnamespace `widget`, which in turn is part of the namespace `yoyodyne`.

Writable and Extensible Namespaces

When defining a namespace in the metadata schema, you can define two optional properties:

- **Writable**

If a namespace is *writable*, you can specify any field in the namespace when you store an object. If a namespace is non-writable, it is read-only, and you cannot specify any of the fields. The `system` namespace, for example, is non-writable (read-only). If a namespace is non-writable, any subnamespaces you add will also be non-writable.

- **Extensible**

By default, namespaces are *extensible*, which means that you can add attributes or subdomains to the namespace. You can change a namespace from extensible to non-extensible, but the reverse is not true.

Reserved Namespaces

The 5800 system reserves a namespace called `system` for metadata created by the 5800 system itself and a namespace called `filesystem` to specify how the file system layer presents files. For example, the `system` namespace includes the creation time for an object and the `filesystem` namespace includes the object's user identifier (UID) and group identifier.

[TABLE 7-2](#) lists the namespaces that 5800 system reserves.

TABLE 7-2 Reserved Domains

Name	Writable	Extensible
system	false	false
filesystem	true	false

The system Namespace

TABLE 7-3 lists the contents of the reserved `system` namespace.

TABLE 7-3 `system` Namespace Contents

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

The filesystem Namespace

TABLE 7-4 lists the contents of the reserved `filesystem` namespace.

TABLE 7-4 `filesystem` Namespace Contents

Attribute Name	Definition
<code>filesystem.uid</code>	Owner ID
<code>filesystem.gid</code>	Group ID
<code>filesystem.mode</code>	File mode (permissions, etc.)
<code>filesystem.mtime</code>	Last modification time.
<code>filesystem.mimetype</code>	MIME type

Fully Qualified Names

Applications must always use the fully qualified name of the attribute when storing metadata or querying. The fully qualified name includes all the enclosed namespace names from the broadest to the narrowest, separated by dots, followed by the attribute name itself, as in `namespace.subnamespace.fieldName`.

Planning Namespaces

You might want to use the name of your organization or company as the top level namespace, and something like project names as subnamespaces. For example, an organization named Yoyodyne, Inc. might set up their namespaces and subnamespaces as follows:

```
<namespace name="yoyodyne">
  <namespace name="widget">
    <namespace name="oscillation">
      <attribute name="overthruster" />
      ...
    </namespace>
  </namespace>
  <namespace name="lectroid">
    <attribute name="type"/>
    ...
  </namespace>
</namespace>
```

Tables and Columns

You partition the metadata schema into tables and specify each metadata field as a column within a particular table. Objects stored in the 5800 system become rows in one or more tables, depending on which metadata fields are associated with that data.

It is strongly recommended that all fields in a query come from the same table, since queries might fail if they include fields from different tables. The largest supported query string is 8080 bytes. The combined size of all query literals and parameters is also limited to 8080 bytes. If you must use queries that include fields from more than one table, make sure that no more than one query that references multiple tables is running at a time. Refer to the *Sun StorageTek 5800 System Client API Reference Guide* for complete information about query sizes and limitations.

Table Example

Suppose you specify columns in a table *reference* in the metadata schema as follows:

```
<table name="reference" />
  <column name="mp3.artist" />
  <column name="mp3.album" />
  <column name="mp3.title" />
  <column name="dates.year" />
</table>
```

The reference table you create would have the logical layout shown in [TABLE 7-5](#).

TABLE 7-5 Example Table (*reference* Table)

OID	mp3.artist	mp3.album	mp3.title	dates.year
Object1	Benny Goodman	The Very Best of Benny Goodman	St. Louis Blues	2000
Object2	Rod Stewart	The Very Best of Rod Stewart	Maggie May	2001
Object3	Bing Crosby	Null	I'm Dreaming of a White Christmas	Null

When an object is stored in the 5800 system that has any of the specified metadata attributes associated with it (mp3.artist, mp3.album, mp3.title, or dates.year), that object OID is listed as a row in the reference table and the values of the attributes are listed in the column that corresponds to that attribute. If no value is assigned to an attribute for that object, no value is listed in the corresponding column.

If the object has other metadata associated with it, that object will also be stored in whichever tables include that other metadata as columns.

The length Attribute for Fields

You specify a length attribute for fields of type string, binary, and char. The length attribute is important because there are limits to the number of bytes that each table row and each index can store. See ["Planning Tables" on page 79](#) and ["Planning Indexes" on page 82](#) for more information.

Note – The 5800 system emulator supports the same field length as that supported by the 5800 system, within the specified limits.

Trying to store a string, binary, or char value that is longer than the specified field length will result in an error message.

Planning Tables

It is strongly recommended that you store metadata attributes that occur together in queries in the same table, since queries that include fields from different tables may fail. Pay close attention to which metadata attributes occur together in your data, especially if those attributes are used in queries, and group those fields together into the same table.

Conversely, avoid putting fields into the same table that do not occur together in queries, since doing so wastes space and degrades query performance.

Planning Table Rows

When planning tables, be aware that the maximum number of bytes allowed for any single row in the table is 8080.

You might want to specify as small a length value as possible for each field (column) in the table so that you can fit as many columns as possible in the table and any single row will not exceed the 8080-byte limit.

[TABLE 7-6](#) lists the number of bytes that each element in a column consumes. The total amount of space consumed by all the columns in a table cannot exceed 8080 bytes.

TABLE 7-6 Number of Bytes Used By Each Column in a Table Row Definition

Element	Space Consumed
System overhead	78 bytes per table row.
Column (Field)	<p>Each column (or field) in a table row uses 2 bytes per column for overhead, plus the number of bytes in the field. The number of bytes for each field type is as follows:</p> <ul style="list-style-type: none"> • string - twice the length of the field • long - 8 bytes • double - 8 bytes • timestamp - 8 bytes • date - 4 bytes • time - 4 bytes • char - length of the field • binary - length of the field • objectID - 30 bytes <p>For example, a field of type string with length 80 consumes 2 bytes for overhead plus 160 bytes for the field length, for a total of 162 bytes. A field of type date consumes 2 bytes for overhead plus 4 bytes for field length, for a total of 6 bytes.</p>

Planning Table Rows Example

Suppose the fields listed in [TABLE 7-7](#) commonly occur together and will be used together in queries. (Three of these fields are in the namespace *mp3* and one is in the namespace *dates*.)

TABLE 7-7 Examples of Fields to Be Added to a Table

Field	Type	Length Setting	Bytes Required
<code>mp3.artist</code>	string	256	512
<code>mp3.album</code>	string	256	512
<code>mp3.title</code>	string	512	1024
<code>dates.year</code>	long	NA	8

Include each of these fields as columns in the same table, called, for example, *reference*. The maximum number of bytes allowed in any row in the table is 8080. When planning the *reference* table, calculate the total number of bytes used by all the columns combined, to make sure it is less than 8080, as follows:

78 (for system overhead) +
8 (2 per column for column overhead) +
512 (for mp3.artist) +
512 (for mp3.album) +
1024 (for mp3.title) +
8 (for dates.year)

= 2142 bytes total

Since 2142 bytes is less than 8080 bytes, the total combined size of all columns is acceptable.

Planning Tables Checklist

For best results, keep in mind the following considerations when planning tables:

- The maximum number of tables allowed in the schema is 1000. Each table and each index counts against this maximum. There are also some system tables that count against this maximum.
- If you do not specify a field as a column in a table, that field is stored in its own separate table, as the single column in that table.
- The maximum number of bytes allowed for any single row in a table is 8080.
- Each table can include no more than 254 fields.
- Each metadata field can be a column in *only* one table.
- When specifying a field as a column in a table, be sure to use the field's fully qualified name (`namespace.field`).
- Tables might contain fields from more than one namespace.
- If you want to assign a field as a column in a table, you must do so when you first add the field to the schema definition.
- Once you have assigned a field to a table, you cannot move that field to another table, or move it to a different column within the table.
- You cannot add or remove columns from a table once you have defined the table.

Indexes

The system creates indexes on metadata fields to allow those fields to be queried more efficiently. You use virtual file system views to specify the content of the indexes the system creates and to maximize query performance.

Note – You also configure virtual file system views that have nothing to do with indexes. See [“Virtual File System Views” on page 87](#) for information about virtual file system views.

For each virtual file system view you create, the system creates an index of up to 15 fields, as long as those fields all come from the same table.

Note – Each virtual file system view consumes more system resources than in previous versions of the 5800 system. For best performance, create only virtual file system views that are required for an application or to define an index to speed up queries.

Planning Indexes

For virtual file system views that you create in order to specify indexes that improve query performance, follow these guidelines:

- Include only fields from the same table. All entries in an index must be from the same table.

- Make sure that the index requires less than 1024 bytes of space. [TABLE 7-8](#) lists the number of bytes required by each element of the index entry.

TABLE 7-8 Number of Bytes Used By Each Element in Index

Element	Space Consumed
System overhead	78 bytes per index entry.
Field	<p>Each field in an index entry uses 2 bytes for overhead, plus the number of bytes in the field.</p> <p>The number of bytes for each field type is as follows:</p> <ul style="list-style-type: none"> • string - twice the length of the field • long - 8 bytes • double - 8 bytes • timestamp - 8 bytes • date - 4 bytes • time - 4 bytes • char - length of the field • binary - length of the field • objectID - 30 bytes <p>For example, a field of type string with length 80 consumes 2 bytes for overhead plus 160 bytes for the field length, for a total of 162 bytes. A field of type long consumes 2 bytes for overhead plus 8 bytes for field length, for a total of 10 bytes.</p>

Planning Indexes Examples

This section includes two different examples of how you might go about planning for indexes.

Example 1

Suppose you want to have a query on the fields listed in [TABLE 7-9](#):

TABLE 7-9 Example of Fields to Be Added to a Table

Field	Type	Length Setting	Bytes Required
book.author	string	50	100
book.series	string	50	100
book.title	string	50	100
dates.year	long	NA	8

To maximize query performance, you include each of these fields as columns within the same table, called *books*. To maximize performance even further, you create a virtual file system view called, for example, *bookview*, that includes these fields and no others so that an index is created on these fields for querying.

Since all of the fields are from the same table, the system creates an index that includes all of these fields, as long as the total number of bytes required for the index does not exceed 1024. Calculate the number of bytes required for the index as follows:

78	(for system overhead) +
8	(2 per column for column overhead) +
100	(for <code>book.author</code>) +
100	(for <code>book.series</code>) +
100	(for <code>book.title</code>) +
8	(for <code>dates.year</code>)

= 394 bytes total

Since 394 is less than 1024, the system indexes all of the fields, allowing them to be queried at maximum performance.

If you calculate that the fields in a query cannot be indexed because they require too much space, you might want to reduce the length specified for each field. Alternatively, you might want to define a virtual file system view with a smaller set of fields. An index of a subset of fields in the query might still help to speed up query performance.

Example 2

Suppose your system is configured with the schema file shown in [CODE EXAMPLE 7-4](#)

CODE EXAMPLE 7-4 Example Schema File for Index Planning:

```
<?xml version="1.0" encoding="UTF-8"?>
  <!--Using fsView to Create Index on Commonly Searched Fields -->
  <metadataConfig>
    <schema>
      <namespace name="MyTube" writable="true" extensible="true">
        <field name="Title" type="string" length="38"
          queryable="true"/>
        <field name="keywords" type="string" length="120"
          queryable="true"/>
        <field name="owner" type="string" length="25"
          queryable="true"/>
        <field name="format" type="long" queryable="false"/>
        <field name="date" type="string" length="12"
          queryable="true"/>
      </namespace>
    </schema>
    <tables>
      <table name="videos">
        <column name="MyTube.Title"/>
        <column name="MyTube.keywords"/>
        <column name="MyTube.owner"/>
        <column name="MyTube.format"/>
        <column name="MyTube.date"/>
      </table>
    </tables>
  </metadataConfig>
```

If you know that users are likely to do searches on the owner, date, and keywords fields, you could create an index called `key_owner_index` on those fields using the `fsView` tag, as shown in the schema file example shown in [CODE EXAMPLE 7-5](#). (Since keyword is included as the filename property, it is automatically included as an attribute of the `fsView` and therefore included in the index.)

CODE EXAMPLE 7-5 Using fsView to Create Index on Commonly Searched Fields

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Using fsView to Create Index on Commonly Searched Fields-->
<metadataConfig>
  <schema>
    <namespace name="MyTube" writable="true" extensible="true">
      <field name="Title" type="string" length="38"
        queryable="true"/>
      <field name="keywords" type="string" length="120"
        queryable="true"/>
      <field name="owner" type="string" length="25"
        queryable="true"/>
      <field name="format" type="long" queryable="false"/>
      <field name="date" type="string" length="12"
        queryable="true"/>
    </namespace>
  </schema>
  <fsViews>
    <fsView name="key_owner_index" namespace="MyTube"
      filename="${keywords}">
      <attribute name="owner" />
      <attribute name="date" />
    </fsView>
  </fsViews>
  <tables>
    <table name="videos">
      <column name="MyTube.Title" />
      <column name="MyTube.keywords" />
      <column name="MyTube.owner" />
      <column name="MyTube.format" />
      <column name="MyTube.date" />
    </table>
  </tables>
</metadataConfig>
```

Users in this example might also commonly search on just the owner and keyword fields, and also sometimes on owner, keyword, and title. The system cannot process queries that do not exactly match an existing index as quickly as queries that do, but if the query fields are almost the same as the fields in the indexes, the performance might still be acceptable.

You should test the queries on your system to see if additional indexes are required to speed query performance.

Excluding Attributes from Indexes and Queries

By setting `queryable = false`, you can exclude that field from the metadata that is indexed and available for queries. You might want to exclude a field from indexes, for example, if you will only access that field through the `retrieveMetadata` example application, and never through queries.

Planning Tables and Indexes Checklist

To maximize the performance of queries, keep in mind these considerations when planning tables and indexes:

- Plan tables that include metadata fields that commonly occur together in queries in the same tables and separate metadata fields that do not occur together into separate tables. Take into account the space limitations on tables, as described in [“Planning Tables” on page 79](#).
- Determine which fields should be in which indexes. Each query uses at most one index, so the fields in indexes should match the fields in queries as much as possible. Take into account the limitations described in [“Planning Indexes” on page 82](#).
- Define virtual file system views whose purpose is to create the indexes you want. As more indexes are created, the process of storing data in the 5800 system might become slower, so limit the number of virtual file system views you configure to those required for these indexes and those required for applications.

Virtual File System Views

The 5800 system stores data as discrete objects that users retrieve through queries on object identifiers and/or metadata. The data is not stored in the hierarchical structure typical of file systems, which contain directories, subdirectories, and files.

However, you can set up a *virtual view* of the data, which presents the data objects in a hierarchical structure that mimics a file system. For example, for a 5800 system that stores MP3 files, you could set up a virtual view with a directory for the artist, a subdirectory for the album, and file names based on the title of the music files.

Users access the file system views of the data using a browser and the Web-based Distributed Authoring and Versioning (WebDAV) protocol.

WebDAV

You access virtual file system views of the data through the Web-based Distributed Authoring and Versioning (WebDAV) protocol, a set of extensions to the HTTP/1.1 protocol that allows you to read, add, and delete files on remote web servers.

WebDAV is not supported for multi-cell configurations.

Note – Virtual file system views are available to browse whenever the Status at a Glance panel in the GUI or the `sysstat` CLI command shows that the Query Engine status is `HAFaultTolerant`. See [“Monitoring the System” on page 115](#) or [“Obtaining System Status” on page 105](#) for more information.

Using WebDAV to Browse Virtual File System Views

To access the virtual file system views using WebDAV, type the following in the browser’s address page:

```
http://<data_VIP>:8080/webdav
```

where `data_VIP` is the data VIP address of the 5800 system. See [“Data IP Address” on page 39](#) for information about the data VIP address.

WebDAV Example

The following example shows a WebDAV screen that might be displayed in a user’s browser. It lists the virtual file system views defined on that system.

```
.  
..  
byArtist  
byAlbum  
byYear
```

Clicking the links on this page allows users to browse objects as though they were arranged in a file system structure.

For example, suppose you have defined a virtual file system view `byArtist` that includes the subdirectories `artist` and `album` (in that order). You have indicated in the virtual file system view definition that the files should be named by track number (`tracknum`). Clicking `byArtist` in the browser would yield the list of artists, as follows:

```
.  
..  
Beatles  
Madonna  
Prince  
Rush
```

Clicking `Rush` would list the album names, as follows:

```
.  
..  
2112  
Signals
```

Clicking `Signals` would list the album's track numbers, as follows:

```
.  
..  
1  
2  
3  
4  
5  
6  
7  
8
```

Clicking the link for `1` would allow users to access the data object on the 5800 system associated with track `1` of the `Rush` album `Signals`.

Note – The procedure users follow to add or delete files from the virtual file system view using WebDAV varies depending by browser. Consult the browser's documentation or online help for information.

Metadata Attributes and WebDAV Properties

Each file in the 5800 system virtual view appears as a file in the file system exported to WebDAV. The file attributes (stat data) are exported as WebDAV properties.

[TABLE 7-10](#) lists the WebDAV property names and corresponding system metadata attributes. These attributes are regular metadata values accessible through API queries.

TABLE 7-10 WebDAV Property Names and System Metadata Attributes

	WebDAV Property	Metadata Attribute	Description
Predefined 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
Properties specific to 5800 storage system	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 unsigned 64-bit integers, while the creationdate property is returned as an ISO 8601 localized string. The getlastmodified property is a string similar to the output of date(1) (for example, Mon Apr 9 17:57:11 UTC 2007).

Including Additional File Attributes in Virtual Views

As described in [“Metadata Attributes and WebDAV Properties”](#) on page 90, the 5800 system exports a number of file attributes as part of a virtual view. In addition to those attributes that are always exported, you can choose to have the remaining attributes in the filesystem namespace (filesystem.mimetype and filesystem.mtime) exported with the files.

If you choose this option, the WebDAV browser uses the `filesystem.mimetype` attribute as Content-type in the HTTP header. With Content-type supplied in the HTTP header, when a user clicks on a link to download the file, WebDAV opens the appropriate program. Without Content-type in the HTTP header, the WebDAV browser does not know the file's type and simply prompts the user to save the file to disk.

If you are using the CLI to configure virtual views, select this option by setting `fsattrs` to `true` in the schema file, as shown in the [“Example Schema File” on page 71](#).

If you are using the GUI to configure virtual views, select this option by selecting the Include Extended File System Fields checkbox on the Setup Virtual File Systems panel. See [“Configuring Virtual File System Views Using the GUI” on page 101](#) for more information on using the GUI to configure virtual views.

Note – Choosing this option to retrieve the additional filesystem attributes requires an additional query to the 5800 system and therefore might negatively affect system performance.

Directory Structure in a Virtual File System View

You can use the `filesonlyatleaflevel` attribute to control which objects are displayed as part of a virtual file system view.

If you keep the `filesonlyatleaflevel` attribute at its default of `true`, an object is displayed as part of the virtual file system view only if it has metadata values stored in the 5800 system for all fields specified in the attribute list for the virtual file system view and also in the `filename` description.

For example, suppose you have set up a virtual view called `byArtist` as follows:

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

In this case, only objects with metadata values for `title`, `type`, `artist`, and `album` will appear in the virtual file system view. For example, the three objects shown here are stored with metadata values for `title`, `type`, `artist`, and `album`, and therefore appear at the bottom (or “leaf”) level of the directory in the virtual file system view.

```
beatles
  abbey_road
    something.mp3
    because.mp3
    come_together.mp3
```

An object that has metadata values for `title` and `artist`, but not for `type`, or `album`, simply would not appear in the view.

If you set the `filesonlyatleaflevel` attribute to `false`, any object that has metadata values for all fields specified in the `filename` description, as well as metadata values for a subset of the fields in the attribute list, appears in the virtual file system view, at the upper levels of the directory (not at the “leaf-level”).

Note – To appear in an upper-level directory in the structure, all of the object’s attributes at higher levels must have values and all attributes at lower levels must not have values. All attributes in the virtual file system view must be defined for the object.

For example, in the preceding example, if the `filesonlyatleaflevel` attribute were set to `false`, an object with metadata values for `title`, `type`, and `artist`, but not `album`, would appear in the virtual file system view as the song “Shattered” by the Rolling Stones appears here:

```
beatles
  abbey_road
    something.mp3
    because.mp3
    come_together.mp3
  rolling_stones
```

Note – All attributes in a virtual file system view for which you have specified `filesonlyatleaflevel = false` must be in the same table. See [“Tables and Columns” on page 77](#) for more information about tables.

Virtual File System Views in the Schema File

The `fsView` section of the schema file determines the virtual file system views that users can browse using WebDAV. See [“The Metadata Schema” on page 67](#) for more information about virtual file system views.

Note the following for `fsViews` in the schema file:

- Each `fsView` entry specifies a new virtual file system view.
- Within each `fsView` entry:
 - `name` is the top-level directory name of the virtual file system view.
 - `namespace` defines a schema namespace prefix for all of the names used (that is, `title` will be parsed as `mp3.title`). The `namespace` field is optional. If you do not specify the `namespace` field, you must use fully qualified names for all attributes in the file system view (in other words, you must use `mp3.title`, not `title`). You can use attributes from different namespaces within the same file system view.
 - `filename` defines the form of the files that are exposed by the file system in that view.
- Each `fsView` entry contains an ordered list of attributes that defines the view. The first attribute in the list represents the highest directory in the file system view, the next attribute in the list represents a subdirectory within that highest directory, the next attribute in the list represents the next subdirectory down, and so on.
- The `readonly` attribute for `fsView` entries allows you to specify that users accessing data through WebDAV cannot create or delete objects in the file system view.

Note – All attributes in the `system` namespace are read only. If you include a system attribute in an `fsView` entry, that entire entry is automatically read only.

Summary of Metadata Schema Elements

The following table summarizes the purpose and meaning of the fields you must specify and plan for when configuring the metadata schema:

TABLE 7-11 Metadata Schema Fields

Element	Purpose	For more information...
Metadata attribute	Describes something about an object. For example, in a patient record, the metadata attribute <code>doctor</code> might specify the patient's doctor's name. The metadata attribute <code>insurance</code> might specify the patient's insurance company.	"Metadata" on page 73
Namespace	Organizes metadata names into collections of names, similar to directories.	"Namespaces" on page 74
Table	Uses rows and columns to group metadata attributes that commonly occur together into a single group.	"Tables and Columns" on page 77
Index	Mechanism that allows the system to query metadata fields. Each virtual file system view created becomes an index. You can use virtual file system views to control the content of the indexes the system creates and maximize query performance.	"Indexes" on page 82
Virtual File System Views	Allow you to view files using WebDAV in a hierarchical structure that mimics a file system. Each view you create also becomes an index, so even if you do not plan on using WebDAV to browse files, you will want to create views in order to specify indexes to maximize query performance.	"Virtual File System Views" on page 87

Configuring the Metadata Schema Using the CLI

Note – Before configuring the metadata schema, make sure that the query database is online by issuing the `sysstat` command and checking that the “Query Engine Status” indicates *HAFaultTolerant*. See “[sysstat](#)” on page 157 for more information about the `sysstat` command.

▼ To Modify the Schema File Using the CLI

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

A schema overlay is an XML file that follows the specification shown in “[Schema File DTD](#)” on page 69. It contains only the new namespaces and fields that you wish to add.

If you wish, you can use `mdconfig` followed by the `-t` or `--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. To perform a validation on the `overlay.xml` file, use the command `mdconfig` followed by the `-p` or `--parse` option.

Note – You can use `ssh` to log in to the 5800 system and issue CLI commands such as `mdconfig` simultaneously by typing the `ssh` command and the CLI command on the same line. That method of issuing commands is useful in this procedure and is shown in the following examples. You may have to enter the administrative password for the 5800 system before the command takes effect.

For example, to validate the local `overlay.xml` file, type the following command from the system on the network where the `overlay.xml` file is stored:

```
$ cat overlay.xml | ssh admin@admin_IP mdconfig --parse
```

Once you are satisfied with the overlay, you must commit it so the 5800 system can execute it.

- To commit the `overlay.xml` file, use the command `mdconfig` followed by the `-a` or `--apply` option.**

For example, to continue the previous example, enter the following command from the system on the network where the `overlay.xml` file is stored:

```
$ cat overlay.xml | ssh admin@admin_IP mdconfig --apply
```

Note – The `--apply` option runs a validation before performing the commit operation. If the XML syntax is not correct, the system returns an error.

If the system is under heavy load, you might see the following error message when you issue the `mdconfig --apply` command:

```
Timed out waiting for the state machine.
```

This message indicates that the new schema definition file has been committed to the system, but not all of the tables have been created.

In this case, reduce the load on the system if possible, and use the command `mdconfig --retry` to finish the table creation:

```
$ ssh admin@admin_IP mdconfig --retry
```

When you issue the `mdconfig --retry` command, the system finishes creating any tables that were not completed during the `mdconfig -a` operation. Tables that had already been created are not affected. You might have to issue the `mdconfig --retry` command several times before all tables are created.

Configuring the Metadata Schema Using the GUI

This section includes procedures for using the GUI to display the current metadata schema and make changes to the schema.

▼ To Display the Current Metadata Schema

- **From the navigational panel, choose Configuration > Metadata Schema > View Schema.**

The View Schema panel is displayed, listing the namespaces and tables that are configured in the schema.

▼ To Display the Fields in a Namespace

1. **From the navigational panel, choose Configuration > Metadata Schema > View Schema.**

The View Schema panel is displayed, listing the namespaces and tables that are configured in the schema.

2. **In the Namespaces section, select the namespace for which you want to display fields.**

The fields are listed in the Fields for Selected Namespace section.

▼ To Display the Fields in a Table

1. **From the navigational panel, choose Configuration > Metadata Schema > View Schema.**

The View Schema panel is displayed, listing the namespaces and tables that are configured in the schema.

2. **In the Tables section, select the table for which you want to display fields.**

The fields are listed in the Columns for Selected Table section.

▼ To Change the Metadata Schema

Note – Before making changes to the metadata schema, make sure that the query database is online by checking that the Status At A Glance panel indicates that “Query Engine Status” is *HAFaultTolerant*.

1. **From the navigation panel, choose Configuration > Metadata Schema.**
2. **Click Set Up Schema.**

The Set Up Schema panel is displayed.

3. **Create namespaces as described in “Creating Namespaces” on page 98. For information about namespaces, see “Namespaces” on page 74.**
4. **Create tables as described in [Creating Tables](#). For information about planning tables, see “Planning Tables” on page 79.**
5. **Click Apply.**

Creating Namespaces

You cannot delete a namespace from the schema. Once a namespace is created, you can only add fields to the namespace, assuming the namespace is extensible. Therefore, review the following information before creating namespaces and namespace fields:

- [“Namespaces” on page 74](#)
- [“Planning Namespaces” on page 77](#)

▼ To Create Namespaces

1. **From the navigational panel, choose Configuration > Metadata Schema.**
2. **Click Set Up Schema.**

The Set Up Schema panel is displayed.

3. **Click the Add button  next to the Namespaces box.**

The Add Namespace panel is displayed.

4. **Type the namespace name.**
5. **Choose the parent namespace from the Parent Namespace drop-down menu.**

Note – Choosing <root> as the parent namespace, selecting the Is Extensible check box, and applying your changes causes this namespace to become a parent namespace.

6. **Define whether the namespace will be writable and/or extensible by clearing or selecting the appropriate check boxes.**

7. **Click the Add button  next to the Fields box.**

Columns are displayed in the Fields box.

8. Specify the following:

- Name - Type the name of the field.
- Data Type - Choose the data type of field.
- Length - Specify the length that the field can be.
- Is Queryable - Clear this check box if you do not want a user to search on this field.

9. Click OK.

The Create Namespace panel is closed, and the newly created namespace and its fields are displayed on the Set Up Schema panel.

10. Create tables for the fields in the namespace as described in [“Creating Tables” on page 99](#).

11. Click Apply.

Creating Tables


You cannot delete a table from the schema. Therefore, review the following information before creating tables:

- [“Configuring the Metadata Schema Using the GUI” on page 96](#)
- [“Planning Tables” on page 79](#)
- [“Planning Tables Checklist” on page 81](#)

▼ To Create Tables

- 1. From the navigation panel, go to Configuration > Metadata Schema.**
- 2. Click Set Up Schema.**


The Set Up Schema panel is displayed.

- 3. Create namespaces as described in [“Creating Namespaces” on page 98](#).**
- 4. Click the Add button  next to the Tables box.**

The Create File System Table panel is displayed.

- 5. Type the table name. For information on planning tables, see [“Planning Tables” on page 79](#).**
- 6. Choose the namespace that contains the fields you want to include in the table.**

The available fields from the namespace are displayed in the Available Fields box.

7. Select the fields that you want included in the table and click the Move Right button  to move the fields to the Selected Fields box.
8. Perform Steps 5 and 6 for all fields that you want to include in the table.
9. Click OK.

The Create Filesystem Table panel is closed and the newly created table is displayed on the Set Up Schema panel.

10. Click Apply.

▼ To Add Fields to an Existing Namespace


Note – You can only add fields to existing namespaces if they are extensible.

1. From the navigation panel, go to Configuration > Metadata Schema.
2. Click Set Up Schema.

The Set Up Schema panel is displayed.

3. Make sure the Show New/Modified Namespaces Only check box is not selected, so that all existing namespaces are displayed on the panel.
4. Select the namespace to which you want to add fields.


The namespace fields are displayed in the Fields for Selected Namespace box.

5. Click the Add button  next to the Fields for Selected Namespace box.

The Add Namespace Fields panel is displayed.

6. Specify the following for this field:

- Name - Type the name of the field.
- Data Type - Choose the data type of field.
- Length - Specify the length that the field can be.
- Is Queryable - Clear this check box if you do not want a user to search on this field.

7. If you want to add another new field, click the Add button  and repeat Steps 5 and 6.

8. Click OK.

The panel is closed and you are returned to the Set Up Schema Panel.

9. Click Apply.

Configuring Virtual File System Views Using the GUI

This section includes procedures for displaying the currently configured virtual file system views, creating new views, and browsing the views.

▼ To Display the Current Virtual File System Views

- From the navigational panel, choose **Configuration > Virtual File Systems > View Virtual File Systems**.

The View Virtual File Systems Views panel is displayed, listing the views that are defined in the system.

▼ To Display the Fields in a View

1. From the navigational panel, choose **Configuration > Virtual File Systems > View Virtual File Systems**.

The View Virtual File Systems Views panel is displayed, listing the views that are defined in the system.

2. In the Views section, select the view for which you want to display fields.

The fields are listed in the Fields for Selected View section.

▼ To Create New Virtual File System Views

1. From the navigation panel, go to **Configuration > Virtual File Systems**.
2. Click **Set Up Virtual File Systems**.

The Set Up Virtual File Systems panel is displayed.


3. Type the view name.

4. If you do not want users who browse this view to be able to add or delete objects, select the Read-Only check box.
5. If you want users who browse this view to see only files for which there are attributes at every level in the hierarchy, select the Files Only at Leaf Level check box.

If you want users to see files at higher levels in the hierarchy if there are no attributes at the lower levels, do not select this checkbox. See [“Directory Structure in a Virtual File System View” on page 91](#) for more information.

6. If you want to include the `filesystem.mimetype` and `filesystem.mtime` attributes for each file as part of the virtual view, select the Include Extended File System Fields check box.

See [“Including Additional File Attributes in Virtual Views” on page 90](#) for more information.

7. In the Available Fields box, select the fields that you want in the view and click the Move Right button  to move the fields to the Selected Fields box.

Note – Fields you select will appear in the virtual view as directories and subdirectories, with the first field selected as the top-level directory and successive fields as subdirectories, in the order you select the fields.

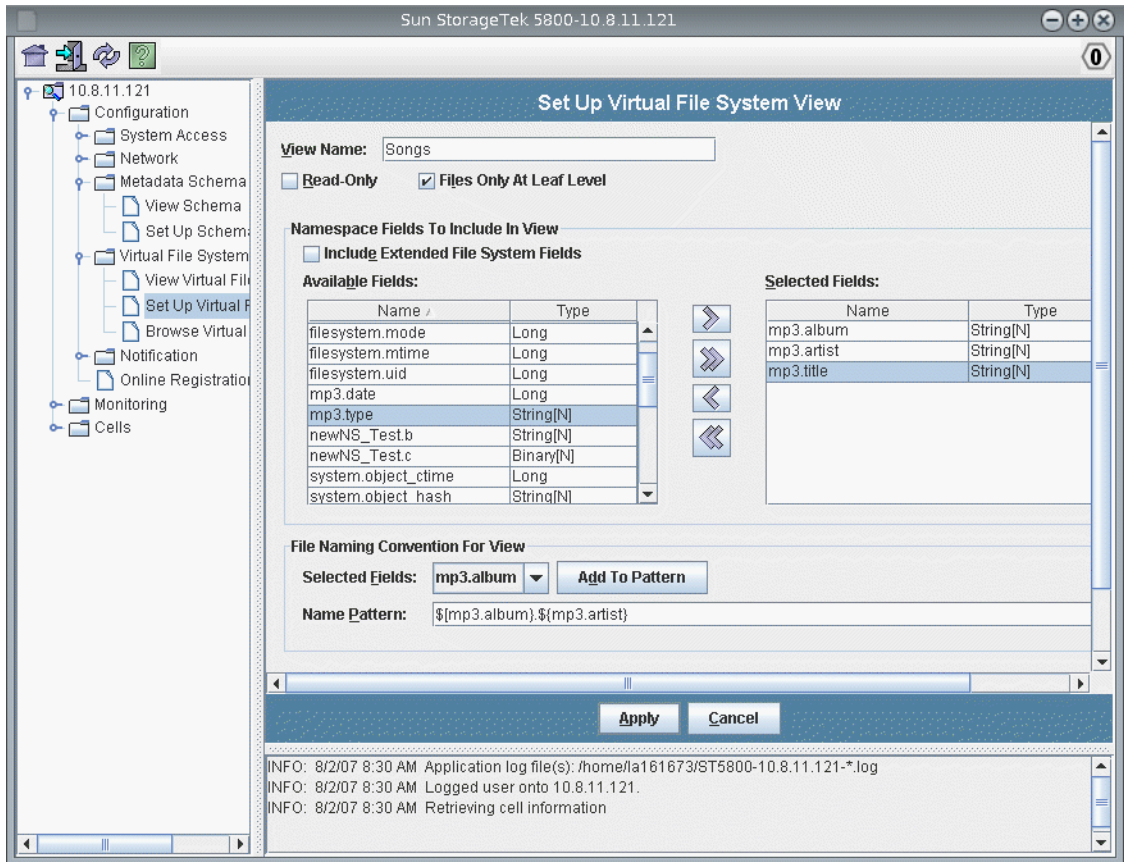
8. In the File Naming Convention For View section, choose a field from the Selected Fields drop-down menu and click Add To Pattern.

The fields you select are displayed in the Name Pattern field. This pattern specifies what the names of the objects will be in the virtual view.

9. Click Apply.

For example, you might set up a virtual file system view named *Songs*, as shown in . Users connecting to the 5800 system using WebDAV would see a virtual file system view that displayed song files in a hierarchy with album as the main folder, and artist and title as subfolders.

FIGURE 7-1 Virtual File System View Configuration Example



▼ To Preview Virtual File System Views

1. From the navigation panel, choose Configuration > Virtual File Systems.
2. Click **Browse Virtual File Systems**.

The virtual file systems configured on your system are displayed, as a user accessing the system using WebDAV would see them.

Monitoring the System Using the CLI

This chapter describes the CLI commands you can use to monitor the 5800 system. For information on using the GUI to monitor the system, see [“Monitoring the 5800 System Using the GUI”](#) on page 115.

This chapter contains the following sections:

- [“Obtaining System Status”](#) on page 105
- [“Displaying Performance Statistics”](#) on page 108
- [“Viewing the System Software Version”](#) on page 109
- [“Obtaining FRU Listings”](#) on page 110
- [“Obtaining Disk Status”](#) on page 111
- [“Obtaining Voltage, Temperature, and Fan Speed Data”](#) on page 113

Note – For instructions on accessing the CLI commands described in this chapter, see [“Using the Administrative Interfaces”](#) on page 21.

Obtaining System Status

- Obtain basic system state information with the command `sysstat`.

This command provides an estimate of free space in the system that is available for data storage. For a detailed breakdown of space usage per disk, refer to the `df` command described in [“Obtaining Disk Status”](#) on page 111.

For example:

```
ST5800 $ sysstat
Cell 0: Online. Estimated Free Space: 7.49 TB
8 nodes online, 32 disks online.
Data VIP 10.8.60.104, Admin VIP 10.8.60.103
Data services Online, Query Engine Status: HAFaultTolerant
Data Integrity check not completed since boot
Data Reliability check last completed at Wed Sep 05 07:12:43 UTC 2007
Query Integrity established as of Wed Sep 05 01:31:20 UTC 2007
NDMP status: Backup ready.
ST5800 $
```

The output that the `sysstat` command produces is explained below. Data reported is for all online disks in the entire system.

- Data services Online indicates that the system is available to read and write to via the API, while Data services Offline means that the system is not available to read and write to via the API.
- Query Engine Status reports the states of the query engine, as follows:
 - HAFaultTolerant - Query services are available and highly fault tolerant.
 - FaultTolerant - Query services are available, but not as fault tolerant as in the HAFaultTolerant state.
 - Operational - Query services are available, but not fault tolerant.
 - Starting - The query engine is starting up. This process may include creating the query database or recreating the connection to the database. Query services are not available during this process.
 - Unknown - The query engine is in an undetermined state. This may be because it is too early in the starting up process to establish a connection to the query engine, or the query engine is in the process of restarting.
 - Stopped - The query engine is stopped; query services are not available.
 - Unavailable - The query engine is not returning any status at this time, probably because it is in a transitional state; query services may not be available.
 - Nonoperational - The query engine is corrupted; no query services will be available until the system has completed recreating the engine.
- Data Integrity check indicates when the system last completed checking each fragment on the system for integrity against bit rot. Each cycle of this testing might take up to one week to complete, so the check will be listed as not complete for the first week after a system reboot.

- **Data Reliability check** indicates when the system last completed a full cycle of testing to detect and recover any missing fragments, indicating that the system has full reliability. Each cycle of this testing takes approximately 12 hours to complete, so the check will be listed as `not complete` for the first 12 hours after a system reboot.
- **Query Integrity established** provides assurance that a query of data stored on the 5800 system will accurately reflect the contents of the object archive. Exceptions would be data that was stored or deleted from the 5800 system while the query was in progress, as well as objects that were stored after the query integrity time and for which the store operation returned the special error status `isIndexed=false` to the storing application.
- **NDMP Status check** indicates the status of the Network Data Management Protocol (NDMP), which allows you to back-up the data stored on the system to tape and restore that data in the event of catastrophic system loss. This check indicates whether the data has been backed up and is available for restoration and also whether the backup or restore is in progress.
- **Obtain extended system state with the command `sysstat --verbose`.**

Verbose output includes the online/offline status of each node and disk in the system. 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 110](#).

For example:

```
ST5800 $ sysstat --verbose
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]
.
.
.
```

Note – If a disk is listed as off line, the disk should be replaced.

Displaying Performance Statistics

- Display real-time performance metrics about throughput and operations using the command `perfstats`.

The metrics displayed reflect activity during a specified time interval (default is 15 seconds). There is a delay between the time an action occurred on the system and the time it is displayed by the `perfstats` command.

For example:

```
ST5800 $ perfstats

Cell Performance Statistics:

                # Ops          Avg          Avg
                -----          Op/sec       KB/sec
Add MD:                0          0.00          0.00
Store:                  0          0.00          0.00
Retrieve:                1          0.20          0.15
Retrieve MD:            0          0.00          0.00
Delete:                 0          0.00           -
Query:                 687         22.90           -
WebDAV Put:             0          0.00          0.00
WebDAV Get:             0          0.00          0.00

Hive Performance Statistics:

Load 1m: 4.12 Load 5m: 4.21 Load 15m: 4.43
Disk Used: 241.28 GB  Disk Total: 13.38 TB  Usage: 1.8%
```

- Display performance statistics for repeated intervals of time, for a specified period of time, using the commands `perfstats --howlong minutes` and `perfstats --interval seconds`.

Note – To specify that the `perfstats` command should run indefinitely, use `--howlong 0`.

- Display performance statistics for a single node in the system using the command `perfstats --node node_id`.

For example:

```
ST5800 $ perfstats --node NODE-101

NODE-101 Performance Statistics:

                Avg           Avg
                # Ops       Op/sec       KB/sec
-----
Add MD:                0         0.00         0.00
Store:                  0         0.00         0.00
Retrieve:                1         0.20         0.15
Retrieve MD:            0         0.00         0.00
Delete:                  0         0.00         -
Query:                  687        22.90         -
WebDAV Put:              0         0.00         0.00
WebDAV Get:              0         0.00         0.00

Hive Performance Statistics:

Load 1m: 4.12 Load 5m: 4.21 Load 15m: 4.43
Disk Used: 241.28 GB Disk Total: 13.38 TB Usage: 1.8%
```

Viewing the System Software Version

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

For example:

```
ST5800 $ version
ST5800 1.1 release [1.1-11076]
```

- Display version information for each node, the service node, and the switches using the command `version --verbose`.

For example:

Note – In normal operation, all nodes should be running the same version of Service Management Daughter Card (SMDC) and the same version of Basic Input Output System (BIOS).

```
ST5800 $ version --verbose
ST5800 1.1 release [1.1-11076]
Service Node:
    BIOS Version: 1.1.3
    SMDC Version: 4.13
Switch:
    Overlay Version (sw#1): 11068
    Overlay Version (sw#2): 11068

NODE-101:
    BIOS version: 0.1.8
    SMDC version: 4.18
NODE-102:
    BIOS version: 0.1.8
    SMDC version: 4.18
NODE-103:
    BIOS version: 0.1.8
    SMDC version: 4.18
NODE-104:
    BIOS version: 0.1.8
    SMDC version: 4.18

.
.
.

ST5800 $
```

Obtaining FRU Listings

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

For example:

```
ST5800 $ hwstat --cellid 8
```

Component	Type	FRU ID	Status
NODE-101	NODE	91230b2c-6ac5-d311-0180-c96e5981e000	ONLINE
DISK-101:0	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHYS3D	ENABLED
DISK-101:1	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHLS7D	ENABLED
DISK-101:2	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHJY1D	ENABLED
DISK-101:3	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHW6ED	ENABLED
NODE-102	NODE	96230b2c-6ac5-d311-0180-c1645981e000	ONLINE
DISK-102:0	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHW8VD	ENABLED
DISK-102:1	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGE8G9D	ENABLED
DISK-102:2	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHVX2D	ENABLED
DISK-102:3	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHW0ND	ENABLED
NODE-103	NODE	92230b2c-6ac5-d311-0180-7e6d5981e000	ONLINE
DISK-103:0	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHVZLD	ENABLED
DISK-103:1	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHN3RD	ENABLED
DISK-103:2	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHVR4D	ENABLED
DISK-103:3	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHVXSD	ENABLED
.			
.			
.			

```
ST5800 $
```

- Obtain information about a specific FRU using the command `hwstat --FRUID fruid`.

For example:

```
ST5800 $ hwstat --FRUID NODE-107
```

Component	Type	FRU ID	Status
NODE-107	NODE	72cda8b6-aec3-d311-0080-2a835981e000	ONLINE
DISK-107:0	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGLX7GD	ENABLED
DISK-107:1	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGLY5PD	ENABLED
DISK-107:2	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGGY8VD	ENABLED
DISK-107:3	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGLXA7D	ENABLED

```
ST5800 $
```

Obtaining Disk Status

Use the `df` command to display a summary of disk usage. Note the following information about the utilization numbers displayed:

- The *used* value in the display is not equivalent to the total number of object bytes stored in the system. The *used* value includes space consumed by data parity, object headers and footers, and query indexes.
- Storage utilization statistics displayed by `df` are refreshed every three minutes.
- When using `df` to view storage utilization, the system reserves 15% of raw storage space to allow for data recovery on a full system.
- Obtain a summary of disk usage in an easily readable format with the command `df --human-readable`.

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).

For example:

```
ST5800 $ df --human-readable
Total: 13.39 TB; Avail: 13.17 TB; Used: 219.52 GB; Usage: 1.6%
```

- Obtain information on the physical space available on the disk with the command `df --physical`.

Note – The system can no longer accept objects for storage when any disk in the system reaches 80% capacity.

For example:.

```
ST5800 $ df --physical
All sizes expressed in 1K blocks
DISK-101:0: Total: 449128448; Avail: 434057216; Used: 15071232; Usage: 3.4%
DISK-101:1: Total: 449128448; Avail: 444561408; Used: 4567040; Usage: 1.0%
DISK-101:2: Total: 449128448; Avail: 444561408; Used: 4567040; Usage: 1.0%
DISK-101:3: Total: 449128448; Avail: 444561408; Used: 4567040; Usage: 1.0%
DISK-102:0: Total: 449128448; Avail: 444561408; Used: 4567040; Usage: 1.0%.
.
.
.
```

Obtaining Voltage, Temperature, and Fan Speed Data

Use the command `sensors` to display voltage, temperature, and fan speed data, as collected by system sensors.

For example:

```
ST5800 $ sensors

NODE-101:
  DDR Voltage           2.60 Volts
  CPU Voltage           1.42 Volts
  VCC 3.3V              3.32 Volts
  VCC 5V                5.12 Volts
  VCC 12V               12.03 Volts
  Battery Voltage       2.98 Volts
  CPU Temperature       49 degrees C
  System Temperature    32 degrees C
  System Fan 1 speed    11340 RPM
  System Fan 2 speed    11340 RPM
  System Fan 3 speed    11070 RPM
  System Fan 4 speed    10980 RPM
  System Fan 5 speed    11070 RPM

NODE-102:
  DDR Voltage           2.60 Volts
  CPU Voltage           1.43 Volts
  VCC 3.3V              3.32 Volts
  VCC 5V                5.10 Volts
  VCC 12V               12.10 Volts
  Battery Voltage       2.98 Volts
  CPU Temperature       49 degrees C
  System Temperature    33 degrees C
  System Fan 1 speed    11700 RPM
  System Fan 2 speed    11430 RPM

.
.
.

ST5800 $
```


Monitoring the 5800 System Using the GUI

This chapter describes the GUI functions you can use to monitor the 5800 system. For descriptions of fields on the GUI screens, click the Help button from the screen.

For information on using CLI commands to monitor the system, see [“Monitoring the System Using the CLI” on page 105](#).

This chapter contains the following sections:

- [“Monitoring the System” on page 115](#)
- [“Monitoring Cells” on page 117](#)
- [“Monitoring Nodes and Disks” on page 118](#)

Note – For instructions on accessing the GUI functions described in this chapter, see [“Using the Administrative Interfaces” on page 21](#).

Monitoring the System

This section includes procedures for using the GUI to monitor the overall 5800 system.

▼ To Display Failed Components

1. **Click the Home icon.**



The Status at a Glance panel is displayed, providing a graphical representation of every cell in the system.

2. **Mouse-over the image of a cell that has a red light or red rectangle displayed.**

The component ID, its status, and its field-replaceable unit identifier (FRU ID) are displayed in the mouse-over window.

▼ To Display the System Space Usage

- **Click the Home icon.**



The Status at a Glance panel is displayed, providing space usage values in the Total Capacity and Total % Capacity Used fields.

▼ To Display System Performance Statistics

- **From the navigational panel, choose Monitoring > View Performance Statistics.**

The View Performance Statistics panel is displayed.

▼ To Display the Environmental Status

You can view environmental status, including temperature and voltage information, about system components.

1. **\From the navigational panel, choose Monitoring > View Environmental Status.**

The Environmental Status panel is displayed.

2. **Choose the search criteria from the drop-down list boxes.**

The environmental status of the components that meet your search criteria is displayed in the table.

Monitoring Cells

This section includes procedures for using the GUI to monitor information about cells.

▼ To Display the System Software Version on a Cell

- **Click the Home icon.**



The Status at a Glance panel is displayed, with the system software version value in the Cell *<identifier>* Summary Cell Version field.

▼ To Display the Nodes in a Cell

- **From the navigation panel, choose Cells > Cell *<identifier>*.**

The Cell Summary panel is displayed, listing the nodes in the selected cell.

▼ To Display the Disks in a Cell

- **From the navigation panel, choose Cells > Cell *<identifier>* > Disks.**

The Disks Summary panel is displayed, listing the disks in the selected cell.

▼ To Display the Cell IP Addresses

- Click the Home icon.



The Status at a Glance panel is displayed, listing the administrative and data IP addresses used by each cell.

Note – To see the IP address of the service node, move your mouse over the service node in the illustration of the cell.

Monitoring Nodes and Disks

This section includes procedures for using the GUI to monitor nodes and disks.

▼ To Display the FRU ID of a Node

- From the navigational panel, choose Cells > Cell *<identifier>*.

The Cell Summary panel is displayed, listing the Field Replaceable Unit (FRU) IDs of the nodes in the Node FRU ID column.

▼ To Display Node Space Usage

- From the navigational panel, choose Cells > Cell *<identifier>*.

The Cell Summary panel is displayed, listing the nodes in the selected cell and their percentage of space used.

▼ To Display the Node Status

1. Click the Home icon.



The Status at a Glance panel is displayed.

2. Mouse-over the image of the cell for which you want to view node status.
3. Determine whether the node is online or offline.

FIGURE 9-1 Online Node



FIGURE 9-2 Offline Node



▼ To Display the Disks in a Node

- From the navigational panel, choose Cells > Cell <identifier> > Nodes > Node <identifier>.

The Node Status panel is displayed, listing the disks in the node.

▼ To Display Disk Statistics

You can display the disk node, disk identifier, status, space usage, and field-replaceable unit identifier (FRU ID) of a disk.

1. Click the Home icon.


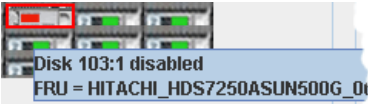


The Status at a Glance panel is displayed.

2. Mouse-over the image of the cell for which you want to view disk statistics.

3. Determine the disk statistics:

TABLE 9-1 Mouse-Over Values on the Status at a Glance Panel

Value Displayed	Description
 <p>Disk 112:1 OK 25% Used</p>	<p>This mouse-over value determines the following:</p> <ul style="list-style-type: none">• The node in which the disk resides is: 112.• The disk identifier is: 112 : 1.• The status of the disk is online: OK.• Twenty five percent of the disk storage space is used.
 <p>Disk 103:1 disabled FRU = HITACHI_HDS7250ASUN500G_0t</p>	<p>This mouse-over value determines the following:</p> <ul style="list-style-type: none">• The node in which the disk resides is: 103• The disk identifier is: 103:1• The disk is offline: disabled.• The field-replaceable unit identifier is listed.

Implementing Disaster Recovery

The 5800 system implements a distributed data model that includes extensive self-healing functionality to protect against localized hardware failure.

This chapter describes how to protect your 5800 system against catastrophic system loss. It contains the following sections:

- [“About the 5800 System Implementation of NDMP” on page 121](#)
- [“Using NetVault to Implement Disaster Recovery” on page 122](#)

About the 5800 System Implementation of NDMP

The 5800 system does not require back-ups in the conventional sense, since the system heals automatically from any failures.

To enable you to recover from a catastrophic system loss, however, the 5800 system implements a subset of the Network Data Management Protocol (NDMP). NDMP allows you to back-up the data stored on the system to tape and restore that data in the event of catastrophic system loss.

The 5800 system NDMP implementation allows only for complete restoration of data to an empty cell, not partial restorations. Before restoring data to a cell, you must use the CLI or GUI to delete all data from the cell. See [“wipe” on page 162](#) for information about the CLI wipe command. See [“To Delete All Data Using the GUI” on page 59](#) for information about deleting data using the GUI.

During a restoration of data to the 5800 system, you must restore the most recent back-up first, and then restore back-ups covering the entire span of the system’s operation. After the most recent back-up is restored, you can restore the other back-ups in any order.

See the remaining sections in this chapter for information about using NetVault as the back-up product and for more guidelines about backing up and restoring data to the 5800 system.

Note – The 5800 system acts as an NDMP data server (filer). It does not implement the optional Direct Access Recovery (DAR) portion of the NDMP protocol, which assumes a directory structure that the 5800 system does not have. The 5800 system does not require the DAR feature to recover individual files since the 5800 system automatically recovers data lost from any hardware failures.

Using NetVault to Implement Disaster Recovery

Sun has tested 5800 system disaster recovery using NetVault, Version 7.4.5, with the NDMP plug-in from BakBone Software on a SPARC[®]-based system running Solaris 10. NetVault supports a wide range of tape devices. After you configure NetVault with the BakBone patch for 5800 system support, you can use the NetVault GUI or CLI to control all back-up and restore operations.

For detailed information about using NetVault with the 5800 system, contact your service representative to obtain a copy of *Protecting the Sun StorageTek 5800 System with Bakbone NetVault using NDMP* and also a pointer to BakBone Software's documentation about using NetVault with the 5800 system.

Note – If you are using the authorized subnetworks feature on the 5800 system, the system that NetVault is running from must be on an authorized subnetwork. If you have left the authorized subnetwork setting at its default of `a11` (thereby allowing any client on the network to access the data stored on the 5800 system) this is not an issue. See [“Authorized Subnetworks” on page 44](#) for more information.

Checking NDMP Status

Use the `sysstat` command to determine the status of back-up and restore on the 5800 system.

```
ST5800 $ sysstat
Cell 0: Online. Estimated Free Space: 7.49 TB
8 nodes online, 32 disks online.
Data VIP 10.8.60.104, Admin VIP 10.8.60.103
Data services Online, Query Engine Status: HAFaultTolerant
Data Integrity check not completed since boot
Data Reliability check last completed at Wed Sep 05 07:12:43 UTC 2007
Query Integrity established as of Wed Sep 05 01:31:20 UTC 2007
NDMP status: Backup ready.
ST5800 $
```

- Backup unavailable - A very rare state indicating either that an error occurred during restore or that the system database that drives back-up is not ready. In the case of an error during restore, restart the restore operation from the beginning. In the case of a problem with the database, the system generally recovers on its own, but if the problem persists, contact Sun support.
- Backup ready - System is ready to be backed up.
- Backup writing to tape: *number_of_objects*, *number_of_bytes* processed. - Backup is in process.
- Restore reading tape: *number_of_objects*, *number_of_bytes* processed. - Restore is in process.
- Restore in progress. Ready for next tape. - Full disaster recovery has been initiated, but data from the entire range of dates to be restored has not been completed. Proceed with the next back-up job.
- Safe to backup to *date*. The back-up process might miss some data stored after the date indicated because the system database has fallen behind the data ingest rate. This state occurs only during periods when the 5800 system is ingesting large amounts of data and is corrected automatically when the database catches up. You may perform a back-up during this time anyway, and then later, when the database is caught up, perform another consolidating back-up to duplicate this time period.

General Guidelines for Backing Up Data

To back-up data to tape, follow these general guidelines:

- Back-ups occur over a single connection. The longer the back-up session, the more chance that something might cause the back-up process to end. Therefore, try to limit the amount of data backed up in a single job. As a general best practice, estimate the rate at which data is being stored on the 5800 system and specify a back-up time range during which 1 TB or less was stored.
- As described in the previous bullet, a best practice is to back-up data in multiple incremental sessions that back-up 1 TB or less of data during each session. You can also issue one or more consolidating back-ups, which span multiple sessions. The longer the range of time you specify for these consolidating back-ups, the more data will be backed up per session and the fewer number of sessions remain to manage. These consolidating sessions might take a long time to complete and should only be performed after the data being consolidated is already accounted for on tape.
- If something causes a back-up session to abort, you must run the entire back-up again from the beginning.

General Guidelines for Restoring Data

To restore data from tape, follow these general guidelines:

- The NDMP implementation on the 5800 system allows only for complete restoration of data to an empty cell, not partial restorations.
- You can only restore data to the same size cell as the cell from which the data was backed up. In other words, you cannot restore data from a half-cell system to a full-cell system.
- The system software version running on the 5800 system to which you are restoring data must be identical to the system software version that was running on the 5800 system when the last (most recent) backup was made.
- Before restoring data to a cell, you must use the CLI or GUI to delete all data from the cell. See [“wipe” on page 162](#) for information about the CLI wipe command. See [“To Delete All Data Using the GUI” on page 59](#) for information about deleting data using the GUI.
- All nodes and disks in the system must be online before starting the restore operation. If any nodes or disks are missing, contact Sun service to replace the disks and bring them online before attempting the restoration.

- If any nodes or disks fail during the restore operation, you must contact Sun service to replace the failed disks or nodes, and then you must use the CLI or GUI to delete all data from the cell and begin the restoration again. See [“wipe” on page 162](#) for information about the CLI wipe command. See [“To Delete All Data Using the GUI” on page 59](#) for information about deleting data using the GUI.
- You must restore the most recent back-up first, and then restore back-ups covering the entire span of the system’s operation. After the most recent back-up is restored, you can restore the other back-ups in any order.
- A complete restoration of the data can require a significant amount of time. For best results, after you have restored the most recent back-up, restore the highest priority data first.
- During restoration of the most recent backup, all client services, such as WebDAV and API access, are unavailable. To minimize this potential downtime, and to protect the most current data, a best practice is to do a relatively small back-up every day to have on hand as the first (most recent) back-up to restore.
- During restoration of the most recent backup, any administrative changes made to the system’s configuration will be lost. Therefore, while the first restore is in progress, do not make any changes to NTP settings, DNS settings, or other configuration settings.
- You may not have access to the data on the 5800 system via WebDAV for up to 12 hours after the first restore has completed.
- After the first restore session, you must reboot the 5800 system to ensure that the query engine and WebDAV function correctly after restoration is complete.

Reference: CLI Commands

The CLI allows you to see system status, configure system properties, and administer the hardware. It is also scriptable, as explained in [“To Create and Execute a Script for CLI Commands” on page 24](#).

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 can select between multiple parameters, options are separated by a bar (|).

This appendix contains the following commands:

- [“alertcfg” on page 129](#)
- [“cellcfg” on page 130](#)
- [“copyright” on page 132](#)
- [“date” on page 133](#)
- [“df” on page 134](#)
- [“help” on page 136](#)
- [“hiveadm -s|--status” on page 138](#)
- [“hivecfg” on page 139](#)
- [“hwstat” on page 142](#)
- [“logout” on page 144](#)
- [“mdconfig” on page 145](#)
- [“passwd” on page 148](#)
- [“perfstats” on page 150](#)
- [“reboot” on page 153](#)
- [“sensors” on page 154](#)
- [“shutdown” on page 156](#)
- [“sysstat” on page 157](#)

- “version” on page 160
- “wipe” on page 162

alertcfg

Description

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

Syntax

```
alertcfg [options]
```

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

Options

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

Examples

```
ST5800 $ alertcfg add to fred@samplecompany.com  
ST5800 $ alertcfg del cc admin@samplecompany.com
```

cellcfg

Description

Sets and displays the following network configuration parameters per cell or for all cells in the hive:

- Data IP address
- Administrative IP address
- Service node IP address
- Gateway
- Subnet

Syntax

```
cellcfg [options]
```

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

Options

- `-c, --cellid cellid`
Specifies ID of the cell you want to configure. In a multicell configuration, you must enter a cell ID.

Note – If you change the cell ID, you must use the `wipe` command to delete all data from the system. See [“wipe” on page 162](#) for information about deleting data from the system.

- `-z, --set`
Allows you to change values in interactive mode.
- `-a, --admin_ip address`
Sets the administrative IP address.
- `-d, --data_ip address`

Sets the data IP address.

- `-g, --gateway address`
Sets the IP address of the gateway.
- `-n, --service_node_ip address`
Sets the IP address of the service node.
- `-u, --subnet mask`
Sets the subnet mask of the gateway.

Examples

```
ST5800 $ cellcfg --set
Enter new value, or hit <enter> to leave the value unchanged:

Admin IP Address [10.7.225.161]:
Data IP Address [10.7.225.162]:
Service Node IP Address [10.7.225.160]:
Subnet [255.255.252.0]:
Gateway [10.7.227.254]:

No values changed.
ST5800 $
```

copyright

Description

Displays copyright information for the 5800 system

Syntax

```
copyright
```

Options

None.

Examples

```
ST5800 $ copyright  
Copyright (C) 2007 Sun Microsystems, Inc. All rights reserved.
```

```
Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product 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.
```

```
U.S. Government Rights - Commercial software.  
Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements. Use is subject to license terms. This distribution may include materials developed by third parties. Portions may be derived from Berkeley BSD systems, licensed from U. of CA. Sun, Sun Microsystems, the Sun logo, Java and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.
```

```
ST5800 $
```

date

Description

Returns master node date and time.

Syntax

```
date [options]
```

Option

- `-c, --cellid cellid`

Specifies ID of the cell for which you want to display system time. In a multicell configuration, you must enter a cell ID.

Example

```
ST5800 $ date  
Thu Jun 28 12:43:17 UTC 2007
```

df

Description

Reports estimated disk space usage on the system. Note the following information about the utilization numbers displayed:

- The *used* value in the display is not equivalent to the total number of object bytes stored in the system. The *used* value includes space consumed by data parity, object headers and footers, and query indexes.
- Storage utilization statistics displayed by `df` are refreshed every three minutes.
- When using `df` to view storage utilization, be aware that the system reserves 15% of raw storage space to allow for data recovery on a full system.

Syntax

```
df [options]
```

Options

- `-c, --cellid cellid`
Specifies ID of the cell for which you want to display statistics. In a multicell configuration, you must enter a cell ID.
- `-h, --human-readable`
Print the results in easily readable summary format. Rounding of numbers means totals are approximate.
- `-p, --physical`
Prints the physical space available on the disk.

Examples

```
ST5800 $ df
All sizes expressed in 1K blocks
Total: 14372110336; Avail: 14141931520; Used: 230178816; Usage: 1.6%
ST5800 $ df -p
All sizes expressed in 1K blocks
DISK-101:0: Total: 449128448; Avail: 434057216; Used: 15071232; Usage: 3.4%
DISK-101:1: Total: 449128448; Avail: 444561408; Used: 4567040; Usage: 1.0%
DISK-101:2: Total: 449128448; Avail: 444561408; Used: 4567040; Usage: 1.0%
DISK-101:3: Total: 449128448; Avail: 444561408; Used: 4567040; Usage: 1.0%
DISK-102:0: Total: 449128448; Avail: 444561408; Used: 4567040; Usage: 1.0%
DISK-102:1: Total: 449128448; Avail: 434057216; Used: 15071232; Usage: 3.4%.
.
.
.

ST5800 $
```

help

Description

Displays a list of available CLI commands.

Syntax

```
help [options]
```

Entering `help` with no parameters displays a list of available commands.

Option

- *command-name*

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

Examples

```
ST5800 $ help
Type "help <command>" or "<command> --help" for one of the
commands below
alertcfg      cellcfg      copyright     date
df            help          hiveadm      hivecfg
hwstat        logout        mdconfig     passwd
perfstats     reboot        sensors      shutdown
sysstat       version       wipe
ST5800 $
```

```
ST5800 $ help hwstat
Usage: hwstat [options]
Displays information about the various FRUs in the system. The
component name or FRU ID may be used when specifying the component
to view.

Options:
  -c, --cellid <cellid> Specifies the ID of the cell where the
command should run.
  -f, --FRUID <component> Shows information about component

ST5800 $
```

hiveadm -s|--status

Description

Displays number of cells in the configuration, as well as the cell ID and administrative and data IP address of each cell.

Syntax

```
hiveadm -s|--status
```

Options

None.

Example

```
ST5800 $ hiveadm -s
There is/are 2 cell(s) in the hive:
- Cell 1: adminVIP = 10.7.224.21, dataVIP = 10.7.224.22
- Cell 5: adminVIP = 10.7.224.101, dataVIP = 10.7.224.102
ST5800 $
```

hivecfg

Description

Sets and displays the following network configuration parameters for the hive:

- NTP server
- External logging host
- Authorized clients
- SMTP server and port
- DNS settings

Syntax

```
hivecfg [options]
```

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

Options

- `-z, --set`
Allows you to change values in interactive mode.
- `-h, --authorized_clients address(es)`
Sets the list of clients authorized to access the system. The default is **all**, which specifies that any client can access the system.
- `-D, --dns [y|n]`
Specifies whether DNS is enabled or disabled.
- `-e, --dns_search domain_name`
Specifies other domains that the system should search if the specified domain name does not result in a valid IP address.
- `-m, --domain_name domain_name`
Specifies the domain name the system should use.
- `-n, --ntp_server address`

Sets the NTP server address.

- `-p, --smtp_port port_number`

Sets the port of the SMTP server.

- `-s, --smtp_server address`

Sets the address of the SMTP server.

- `-x, --external_logger address`

Sets the address of the external logging system to which you want alerts sent.

- `-1, --primary_dns_server address`

Specifies the IP address of the first server the system should use to translate domain names.

- `-2, --secondary_dns_server address`

Specifies the IP address of the server the system should use to translate domain names if the primary server is unavailable.

Note – You can enter multiple values for NTP servers, authorized clients, and DNS search domains. Each value must be separated by a comma.

Examples

```
ST5800 $ hivecfg --primary_dns_server 10.8.11.110
You must reboot the hive with 'reboot -all' for all changes to take
effect.
ST5800 $ hivecfg --authorized_clients all
You must reboot the hive with 'reboot -all' for all changes to take
effect.
ST5800 $ hivecfg --dns n
Converting authorized client hc-dev.sfbay to 10.7.228.10.
Converting authorized client c192.sfbay to 10.7.227.92.
Converting external logger hclog301.sfbay to 10.7.224.10.
Converting NTP server hclog301.sfbay.sun.com to 10.7.224.10.
Converting SMTP server centralmail4brm.Central.Sun.COM to
129.147.62.198.
Updating hive configuration settings...
You must reboot the hive with 'reboot --all' for all changes to
take effect.
ST5800 $ hivecfg --set
Enter new value, or hit <enter> to leave the value unchanged:
[multiple values need to be comma separated]

NTP Server [10.7.224.10]: hclog301.sfbay.sun.com
SMTP Server [129.147.62.198]: centralmail4brm.Central.Sun.COM
SMTP Port [25]:
Authorized Clients [10.7.228.10,10.7.227.92]: all
External Logger [10.7.224.10]: hclog301.sfbay
DNS [y or n] [n]: y
Domain Name [sfbay.sun.com]:
DNS Search [sun.com]:
Primary DNS Server [10.7.224.10]:
Secondary DNS Server [129.146.11.21]:

Old NTP Server [10.7.224.10] new: hclog301.sfbay.sun.com
Old SMTP Server [129.147.62.198] new:
centralmail4brm.Central.Sun.COM
Old Authorized Clients [10.7.228.10,10.7.227.92] new: all
Old External Logger [10.7.224.10] new: hclog301.sfbay
Old DNS [n] new: y
Validating NTP server hclog301.sfbay.sun.com...
Updating hive configuration settings...
You must reboot the hive with 'reboot --all' for all changes to
take effect.
ST5800 $
```

hwstat

Description

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

Syntax

```
hwstat [options]
```

Options

- `-c, --cellid cellid`
Specifies the cell for which you want to see information. This option is required in a multicell configuration.
- `-f, --FRUID component`
Specifies the identification of the FRU for which you want to see information.

Examples

ST5800 \$ **hwstat -c 8**

Component	Type	FRU ID	Status
NODE-101	NODE	91230b2c-6ac5-d311-0180-c96e5981e000	ONLINE
DISK-101:0	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHYS3D	ENABLED
DISK-101:1	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHLS7D	ENABLED
DISK-101:2	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHJY1D	ENABLED
DISK-101:3	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHW6ED	ENABLED
NODE-102	NODE	96230b2c-6ac5-d311-0180-c1645981e000	ONLINE
DISK-102:0	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHW8VD	ENABLED
DISK-102:1	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGE8G9D	ENABLED
DISK-102:2	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHVX2D	ENABLED
DISK-102:3	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHW0ND	ENABLED
NODE-103	NODE	92230b2c-6ac5-d311-0180-7e6d5981e000	ONLINE
DISK-103:0	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHVZLD	ENABLED
DISK-103:1	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHN3RD	ENABLED
DISK-103:2	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHVR4D	ENABLED
DISK-103:3	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHVXSD	ENABLED
NODE-104	NODE	93230b2c-6ac5-d311-0180-e65a5981e000	ONLINE
DISK-104:0	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHYTYD	ENABLED
DISK-104:1	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGGM4WD	ENABLED
DISK-104:2	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGEJ9HD	ENABLED
DISK-104:3	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGHEXED	ENABLED

.
.
.

ST5800 \$ **hwstat -f NODE-107**

Component	Type	FRU ID	Status
NODE-107	NODE	72cda8b6-aec3-d311-0080-2a835981e000	ONLINE
DISK-107:0	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGLX7GD	ENABLED
DISK-107:1	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGLY5PD	ENABLED
DISK-107:2	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGGY8VD	ENABLED
DISK-107:3	DISK	ATA_____HITACHI_HDS7250S_____KRVN63ZAGLXA7D	ENABLED

ST5800 \$

logout

Description

Ends the administrative session.

Syntax

```
logout
```

Options

None

Example

```
ST5800 $ logout  
Connection to hcl-admin closed.  
client $
```

mdconfig

Description

Enables updates to the existing metadata schema.

Syntax

```
mdconfig [options]
```

Options

- `-a, --apply`
Applies changes into the schema file. The system reads in changes as standard input.
- `-l, --list`
Displays the current schema configuration as standard output.
- `-r, --retry`
Finishes creating tables that might not have been fully created if you issued the `mdconfig -a` or `mdconfig --apply` option while the system was under heavy load.
- `-p, --parse`
Parses and validates a schema before applying the changes. The system reads input from this option as standard input.
- `-t, --template`
Returns a template that you can use as a starting point to modify the schema configuration.

Example

```
ST5800 $ mdconfig -t
<?xml version="1.0" encoding="UTF-8"?>

<!--
  $Id: metadata_config_template.xml 9373 2006-09-30 01:07:35Z pc198268 $

  Copyright 2006 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.
  Refer to the administration guide (chapter 4) for more details
-->

<metadataConfig>

  <schema>

    <namespace name="NAMESPACE_NAME">
      <field name="FIELD1" type="string" length="512"/>
      <field name="FIELD2" type="long"/>
      <field name="FIELD3" type="double"/>
      <field name="FIELD4" type="binary" length="128"/>
      <field name="FIELD5" type="char" length="128"/>
      <field name="FIELD6" type="date"/>
      <field name="FIELD7" type="time"/>
      <field name="FIELD8" type="timestamp"/>
    </namespace>

  </schema>

  <fsViews>

    <fsView name="FSVIEW_NAME" filename="${FIELD3}.num" namespace=
"NAMESPACE_NAME">
      <attribute name="FIELD1"/>
      <attribute name="FIELD2"/>
    </fsView>

  </fsViews>

(continued)
```

```
<tables>
  <table name="TABLE1">
    <column name="NAMESPACE_NAME.FIELD1" />
    <column name="NAMESPACE_NAME.FIELD2" />
    <column name="NAMESPACE_NAME.FIELD3" />
    <column name="NAMESPACE_NAME.FIELD4" />
    <column name="NAMESPACE_NAME.FIELD5" />
    <column name="NAMESPACE_NAME.FIELD6" />
    <column name="NAMESPACE_NAME.FIELD7" />
    <column name="NAMESPACE_NAME.FIELD8" />
  </table>
</tables>

</metadataConfig>
```

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 [-K | --pubkey < pubkeyfile]
```

Option

- `-K, --pubkey < pubkeyfile`

Configures the system so that you can log in from a client without entering a password. *pubkeyfile* is the file containing the public key.

Consult your ssh application documentation for instructions on creating key files. Create the public key without a passphrase. You might be prompted to enter the CLI password (default is `admin`) during key generation.

Once you have configured the public key file on the 5800 system, you can log in from any client with the private version of that key without being prompted for a password. If you wish to return to interactive logins, remove the private key from the client, or configure a new public key on the 5800 system.

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

Note – The `--pubkey` option is available only in non-interactive mode (that is, you must enter it at the same time that you enter the `ssh` command, as in `ssh admin@10.7.227.101 passwd --pubkey < key.pub`).

Examples

```
ST5800 $ passwd
Enter current password:
Enter new password:
Re-enter new password:
CLI admin: The admin password has been changed successfully.
ST5800:

client $ ssh admin@10.8.163.10 passwd --pubkey < id_dsa.pub
Password:XXXXXX
CLI admin: The public key has been changed successfully
client $

client $ cat id_dsa.pub | ssh admin@10.8.163.10 passwd --pubkey
Password:XXXXXX
CLI admin: The public key has been changed successfully
client $
```

perfstats

Description

Displays real-time performance statistics about throughput and operations.

Syntax

```
perfstats [options]
```

Options

- `-c, --cellid cellid`
Specifies the cell for which you want to view statistics. In a multicell configuration, you must enter a cell ID.
- `-t, --howlong minutes`
Specifies the time, in minutes during which you want to monitor performance statistics. By default, the statistics display indefinitely.
- `-i, --interval seconds`
Specifies the interval, in seconds, at which you want to display statistics. By default, the interval is 15 seconds.
- `-n, --node node_id`
Specifies a node for which you want to display statistics. By default, statistics for all nodes are displayed.

Examples

```
ST5800 $ perfstats
```

```
Cell Performance Statistics:
```

	# Ops	Avg Op/sec	Avg KB/sec
	-----	-----	-----
Add MD:	0	0.00	0.00
Store:	0	0.00	0.00
Retrieve:	1	0.20	0.15
Retrieve MD:	0	0.00	0.00
Delete:	0	0.00	-
Query:	687	22.90	-
WebDAV Put:	0	0.00	0.00
WebDAV Get:	0	0.00	0.00

```
Hive Performance Statistics:
```

```
Load 1m: 4.12 Load 5m: 4.21 Load 15m: 4.43
```

```
Disk Used: 241.28 GB Disk Total: 13.38 TB Usage: 1.8%
```

```
ST5800 $ perfstats -n NODE-101
```

```
NODE-101 Performance Statistics:
```

	# Ops	Avg Op/sec	Avg KB/sec
	-----	-----	-----
Add MD:	0	0.00	0.00
Store:	0	0.00	0.00
Retrieve:	1	0.20	0.15
Retrieve MD:	0	0.00	0.00
Delete:	0	0.00	-
Query:	687	22.90	-
WebDAV Put:	0	0.00	0.00
WebDAV Get:	0	0.00	0.00

```
Hive Performance Statistics:
```

```
Load 1m: 4.12 Load 5m: 4.21 Load 15m: 4.43
```

```
Disk Used: 241.28 GB Disk Total: 13.38 TB Usage: 1.8%
```

reboot

Description

Reboots a cell. Rebooting causes an alert to be sent.

Syntax

```
reboot [options]
```

Options

- `-c, --cellid cellid`
Specifies ID of the cell you want to reboot. In a multicell configuration, you must enter a cell ID.
- `-A, --all`
Reboots the switches and service node along with the storage nodes.

Examples

```
ST5800 $ reboot  
Reboot? [y/N]: y  
Connection to hc1-admin closed.
```

sensors

Description

Displays voltage, temperature, and fan speed data, as collected by system sensors.

Syntax

```
sensors [options]
```

Option

- `-c, --cellid cellid`
Specifies ID of a cell for which you want to see sensor data. In a multicell configuration, you must enter a cell ID.

Example

```
ST5800 $ sensors

NODE-101:
    DDR Voltage           2.60 Volts
    CPU Voltage           1.42 Volts
    VCC 3.3V              3.32 Volts
    VCC 5V                5.12 Volts
    VCC 12V               12.03 Volts
    Battery Voltage       2.98 Volts
    CPU Temperature       49 degrees C
    System Temperature    32 degrees C
    System Fan 1 speed    11340 RPM
    System Fan 2 speed    11340 RPM
    System Fan 3 speed    11070 RPM
    System Fan 4 speed    10980 RPM
    System Fan 5 speed    11070 RPM

NODE-102:
    DDR Voltage           2.60 Volts
    CPU Voltage           1.43 Volts
    VCC 3.3V              3.32 Volts
    VCC 5V                5.10 Volts
    VCC 12V               12.10 Volts
    Battery Voltage       2.98 Volts
    CPU Temperature       49 degrees C
    System Temperature    33 degrees C
    System Fan 1 speed    11700 RPM
    System Fan 2 speed    11430 RPM
    System Fan 3 speed    11250 RPM
    System Fan 4 speed    10980 RPM
    System Fan 5 speed    10980 RPM

.
.
.

ST5800 $
```

shutdown

Description

Shuts down (powers off) all the nodes in a cell. System shutdown causes the system to send an alert.



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

Syntax

```
shutdown [options]
```

Options

- `-c, --cellid`
Specifies the cell you want to shut down. In a multicell configuration, you must enter a cell ID.
- `-A, --all`
Shuts down the service node in addition to the storage nodes.

Examples

```
ST5800 $ shutdown  
It is not safe to shutdown the system  
Do you want to shutdown anyway? [y/N]: y  
Connection to hcl-admin closed.
```

sysstat

Description

Displays information about the system status.

Syntax

```
sysstat [options]
```

Options

- `-c, --cellid`
Specifies the cell for which you wish to view statistics. In a multicell configuration, you must enter a cell ID.
- `-i, --interval`
Specifies a number of seconds at which to repeat the statistics listing.
- `-v, --verbose`
Displays information about the online/offline status of each node and disk in the system.

Note – The online/offline status reported by the `-v` option is the logical system status. To see the state of hardware components, refer to [“hwstat” on page 142](#).

Examples

```
ST5800 $ sysstat
Cell 0: Online. Estimated Free Space: 7.50 TB
8 nodes online, 32 disks online.
Data VIP 10.7.224.182, Admin VIP 10.7.224.181
Data services Online, Query Engine Status: HAFaultTolerant
Data Integrity check not completed since boot
Data Reliability check not completed since boot
Query Integrity established as of Wed Sep 05 01:31:20 UTC 2007
NDMP status: Backup ready.
ST5800 $
```

The output that the `sysstat` command produces is as follows. Data reported is for all online disks in the entire system.

- `Data services Online` indicates that the system is available to read and write to via the API, while `Data services Offline` means that the system is not available to read and write to via the API.
- `Query Engine Status` reports the states of the query engine, as follows:
 - `HAFaultTolerant` - Query services are available and highly fault tolerant.
 - `FaultTolerant` - Query services are available, but not as fault tolerant as in the `HAFaultTolerant` state.
 - `Operational` - Query services are available, but not fault tolerant.
 - `Starting` - The query engine is starting up. This process may include creating the query database or recreating the connection to the database. Query services are not available during this process.
 - `Unknown` - The query engine is in an undetermined state. This may be because it is too early in the starting up process to establish a connection to the query engine, or the query engine is in the process of restarting.
 - `Stopped` - The query engine is stopped; query services are not available.
 - `Unavailable` - The query engine is not returning any status at this time, probably because it is in a transitional state; query services may not be available.
 - `Nonoperational` - The query engine is corrupted; no query services will be available until the system has completed recreating the engine.
- `Data Integrity check` indicates when the system last completed checking each fragment on the system for integrity against bit rot. Each cycle of this testing might take up to one week to complete, so the check will be listed as `not complete` for the first week after a system reboot.

- **Data Reliability check** indicates when the system last completed a full cycle of testing to detect and recover any missing fragments, indicating that the system has full reliability. Each cycle of this testing takes approximately 12 hours to complete, so the check will be listed as `not complete` for the first 12 hours after a system reboot.
- **Query Integrity established** provides assurance that a query of data stored on the 5800 system will accurately reflect the contents of the object archive. Exceptions would be data that was stored or deleted from the 5800 system while the query was in progress, as well as objects that were stored after the query integrity time and for which the store operation returned the special error status `isIndexed=false` to the storing application.
- **NDMP Status check** indicates the status of the Network Data Management Protocol (NDMP), which allows you to back-up the data stored on the system to tape and restore that data in the event of catastrophic system loss. This check indicates whether the data has been backed up and is available for restoration and also whether the backup or restore is in progress.

```
ST5800 $ 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]
.
.
.

ST5800 $
```

version

Description

Displays the version of the system software.

Syntax

```
version [options]
```

Options

- `-c, --cellid`
Specifies the cell for which you wish to view the version. In a multicell configuration, you must enter a cell ID.
- `-v, --verbose`
Displays the version information for each node, the service node, and the switches.

Example

```
ST5800 $ version -v
ST5800 1.1 release [1.1-11076]
Service Node:
    BIOS Version: 1.1.3
    SMDC Version: 4.13
Switch:
    Overlay Version (sw#1): 11068
    Overlay Version (sw#2): 11068
NODE-101:
    BIOS version: 0.1.8
    SMDC version: 4.18
NODE-102:
    BIOS version: 0.1.8
    SMDC version: 4.18
NODE-103:
    BIOS version: 0.1.8
    SMDC version: 4.18
NODE-104:
    BIOS version: 0.1.8
    SMDC version: 4.18
.
.
.

ST5800 $
```

wipe

Description

Deletes all data and metadata on the entire hive. It does not delete the system software.



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

Syntax

`wipe`

Options

None.

Example

```
ST5800 $ wipe  
Destroy all data and clear the metadata schema? [y/N]:
```

Glossary

- administrative IP address** The virtual IP (VIP) address exported by the 5800 system for administrative access to a cell.
- API** Application programming interface. A set of routines, protocols, and tools that developers use to build software applications.
- attribute** An entry in the schema that associates a name with a type. For example, the name Doctor might be of type string. Metadata is stored by assigning a value of the appropriate type to an attribute name, and attributes can also be used to create virtual file system views.
- authorized client** Clients that are authorized to access data on the 5800 system. By default, the system allows any client on the network to access the data stored on the 5800 system, but you can specify a list of authorized clients, which are the only clients that have access to the data.
- cell** The basic building block of the 5800 system. A full-cell configuration consists of 16 storage nodes, two gigabit Ethernet switches, and one service node.
- CLI** Command-line interface. Text-based form of communication with the 5800 system. You access the CLI by issuing the command **ssh admin@adminIPaddress** from a host on the same network as the 5800 system.
- client** An application that runs on a personal computer or workstation and relies on a server to perform some operations.
- cluster** A term sometimes used to refer to the 5800 system cell or cells in a configuration.
- CPU** Central processing unit. The brains of the computer, sometimes referred to simply as the processor or central processor. The CPU is where most calculations take place.

ctime	Creation time. The system metadata includes information on the creation time, data length, and data hash.
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.
data IP address	The virtual IP (VIP) exported by the 5800 system for access to the data stored on a cell.
data object	A stored file with an associated Object ID (OID).
disk mask	A current record of disk availability across the system.
DNS	Domain Name Service. A service that defines naming conventions that translate domain names into IP (Internet Protocol) addresses.
DTD	Document Type Definition. 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.
emulator	Software that imitates the behavior of a 5800 system, allowing you to test applications.
extended metadata	Metadata that is added by the user of the 5800 system. User metadata consists of <i>name=value</i> 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.
file system view	See <i>virtual file system view</i> .
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 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 5800 system, it is divided into five data fragments and two corresponding parity fragments.
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).
fsView	Section of the metadata schema file where you specify virtual file system views. fsViews are also used to specify which indexes the system creates for responding to metadata queries.

- full-cell** A 5800 system configuration that includes 16 storage nodes, two gigabit Ethernet switches, and one service node.
- gateway** A router that connects the local subnet on which the 5800 system resides to the larger network. You must configure a default gateway for each 5800 system cell, to enable information about the system to be available on the network.
- GB** Gigabyte. Represents 2 to the 30th power (1,073,741,824) bytes. One gigabyte is equal to 1,024 megabytes.
- GUI** graphical user interface. A graphical form of communication with the 5800 system. You access the GUI by typing the administrative IP address and GUI port number in the URL line in a Java-enabled web browser connected to the same network as the 5800 system.
- HADB** High-availability database. A highly available and scalable, always-on relational database management system used to store metadata on the 5800 system.
- half-cell** A 5800 system configuration that includes eight storage nodes, two gigabit Ethernet switches, and one service node
- hive** A multicell configuration including at least two full-cell (16-node) 5800 system storage nodes.
- HTML** HyperText Markup Language. Designed to display data and focus on how data looks. The tags you use to mark up HTML documents and the document's structure are predefined, so 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.
- index** A sequence of columns in the metadata database against which queries are made.
- 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 5800 system: system and extended.
- 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.
- multicell** A configuration including more than one full-cell of sixteen 5800 system storage nodes. Also called a *hive*.

- namespace** A collection of names, identified by a uniform resource identifier (URI), that XML uses to keep names from separate sources from colliding unintentionally. You can have as many namespaces as desired in the 5800 system metadata schema. There is also no limit on the number of namespaces that can be encapsulated within a given namespace level (*subnamespaces*).
- NDMP** Network Data Management Protocol. An open standard backup protocol implemented on the 5800 system to allow you to back up the data stored on the system to tape and restore that data in the event of catastrophic system loss.
- 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.
- placement algorithm** Calculation that determines where to store the data and parity chunks of an object stored on the 5800 system. When a data object comes into the system, the gigabit Ethernet switch directs the store request to a storage node, and that node fragments the object and distributes the fragments to different disks in the system according to the placement algorithm.
- query** A request for information from a database.
- Reed-Solomon Encoding Algorithm** An encoding algorithm that protects data stored in the 5800 system. The Reed-Solomon (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 Advanced Technology Attachment (ATA). An evolution of the Parallel ATA physical storage interface. Serial ATA 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 Serial ATA begin at 150 MBps.
- schema** Defines how the 5800 system metadata is structured. The schema consists of attributes, each of which has a defined type.
- SDK** Software developer's kit. Includes sample applications and command-line routines that demonstrate the 5800 system's capabilities as well as provide good programming examples.

- service node** A Sun Microsystems Sun Fire™ X2100 M2 server with one 250-gigabyte serial ATA (SATA) disk drive. Used by the 5800 system for initial configuration and troubleshooting, and to upgrade the system software.
- SMTP** Simple Mail Transfer Protocol. A protocol for sending email messages between servers. Most email systems that send mail over the Internet use SMTP to send messages from one server to another.
- storage node** A node on which the 5800 system stores data. The storage node includes a single-core AMD Opteron processor, three GB of memory, four 500-GB disk drives, and two Ethernet ports.
- 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). One of the attribute types allowed for metadata on the 5800 system.
- 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.
 - table** Partition of the metadata schema. You partition the metadata schema into tables and specify each metadata `field` as a `column` within a particular table. You can greatly improve the performance of query and store operations by grouping metadata fields that commonly occur together in the same table and by separating metadata fields that do not commonly occur together into separate tables. Objects stored in the 5800 system become rows in one or more tables, depending on which fields are associated with that data.
- virtual IP (VIP)** Virtual IP address. The 5800 system exports two public IP addresses, one to access the data and one to access administrative functions.
- virtual file system view** Arrangements of the data stored in the 5800 system that allow you to use WebDAV to browse the files as though they were stored in a hierarchical path structure. A virtual file system view is defined using the metadata attributes in the metadata schema file.
 - WebDAV** Web-based Distributed Authoring and Versioning. A set of extensions to the HTTP/1.1 protocol that allows you to read, add, and delete files on remote web servers. Using the metadata schema file, you can set up virtual file system views in the 5800 system that allow you to use WebDAV to browse through data files on the system as though they were stored in a hierarchical path structure.
 - XML** Extensible markup language. 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

A

access
 network. See network access
 system. See system access
administrative IP address
 about, 18
 configuring, 38
administrative password, 23, 27, 28, 41
administrative task overview, 34
alert messages, 61
alertcfg command, 62, 129
algorithm for storing objects, 17
API, 18

B

backing up data, 124

C

cell IDs, 22
cellcfg command, 38, 39, 40, 48, 130
cell-level functions, 22
cells
 displaying disks on, 117
 displaying environmental status for, 116
 displaying IP addresses for, 118
 displaying nodes on, 117
 displaying software version of, 117

CLI

 command reference, 127 to ??
 help on commands for, 25
 logging in to, 23

 logging out of, 23

columns

 about, 77
 example, 78
 planning, 79, 81

command-line interface. See CLI

commands

 alertcfg, 62, 129
 cellcfg, 38, 39, 40, 48, 130
 date, 50, 132, 133
 df, 111, 134
 help, 25, 136
 hiveadm, 22, 138
 hivecfg, 45, 49, 50, 62, 63, 139
 hwstat, 110, 142
 logout, 23, 144
 mdconfig, 95, 145
 passwd, 23, 24, 42, 43, 148
 perfstats, 108, 150
 reboot, 55, 153
 sensors, 113, 154
 shutdown, 54, 156
 sysstat, 95, 105, 157
 version, 109, 160
 wipe, 58, 162

configuration

 full-cell, 3, 5
 half-cell, 3, 7

D

data

 backup, 124
 deleting all from system, 58

- data IP address
 - about, 18, 39
 - configuring, 39
- date and time, checking, 50
- date command, 50, 132, 133
- deleting data from system, 58
- df command, 111, 134
- disaster recovery, 121
- disks
 - displaying in a cell, 117
 - displaying in a node, 119
 - displaying statistics for, 119
 - displaying status of, 111
 - statistics, 119
- Drive Activity LED, 11
- Drive Fault LED, 11
- DTD (Document Type Definition) for schema file, 69

E

- email alert messages, 61
- environmental status, 116
- extended metadata, 17, 73
- extensible namespaces, 75
- external log host, 63

F

- failed components, 116
- fan speed, 113
- Fault LED, 10
- features of system, 2
- fields
 - adding to a namespace, 100
 - displaying in a namespace, 97
 - displaying in a view, 101
 - length attribute for string, 78
 - types allowed for metadata, 74
- FRU listings, 110, 118
- fsView attribute, 93
- full-cell configuration, 3, 5
- fully qualified names in namespaces, 76

G

- gateway, 48
- graphical user interface. See GUI
- GUI

- help for, 29
- logging in to, 26, 28
- logging out, 33
- MIME associations in browser for, 27

H

- half-cell configuration, 3, 7
- hardware overview, 3
- help
 - command, 25, 136
 - for GUI, 29
 - on CLI commands, 25
- hiveadm command, 22, 138
- hivecfg command, 45, 49, 50, 62, 63, 139
- hive-level functions, 22
- hwstat command, 110, 142

I

- indexes
 - about, 82, 94
 - examples, 83
 - planning, 82, 87
- IP addresses
 - administrative, 18, 38
 - data, 18, 39
 - service node, 40

L

- LEDs
 - Drive Activity, 11
 - Drive Fault, 11
 - Fault, 10
 - Locator, 11
 - Power, 10
- Locator LED, 11
- locator switch, 11
- log host for alert messages, 63
- logging in
 - to CLI, 23
 - to GUI, 26, 28
- logging out
 - of CLI, 23
 - of GUI, 33
- logout command, 23, 144

M

mdconfig command, 95, 145

metadata

- about, 17, 73
- schema, 67, 68
- system, 17
- types, 74

MIME associations for GUI browser, 27

monitoring

- cell environmental status, 116
- cell IP addresses, 118
- cell software versions, 117
- disk statistics, 119
- disk status, 111
- disks in a cell, 117
- disks in a node, 119
- environmental status, 116
- failed components, 116
- fields in a namespace, 97
- fields in a view, 101
- FRU listings, 110, 118
- metadata configuration, 96
- node space usage, 118
- node status, 119
- nodes in a cell, 117
- performance statistics, 108, 116
- schema, 97
- system space usage, 116
- system status, 105
- virtual file systems, 101
- voltage, temperature, and fan speed data, 113

multicell configuration

- using master cell for administrative access in, 23
- using master cell for administrative access in, 26
- wiping data from, 59

multiple users, 33

N

namespaces

- about, 74
- adding fields to, 100
- displaying fields in, 97
- fully qualified names for, 76
- planning, 77
- system, 76
- writable and extensible, 75

NDMP (Network Data Management Protocol), 121

NetVault, using for disaster recovery, 122

network access

- about, 47
- configuring gateway for, 48
- configuring NTP servers for, 49

Network Data Management Protocol (NDMP), 121

network patch panel, 15

nodes

- displaying disks in, 119
- displaying FRU listings for, 118
- displaying in a cell, 117
- displaying space usage on, 118
- monitoring, 118
- service, 12
- status, 119
- storage, 9

nodes, service, 12

NTP (Network Time Protocol) server, 49

O

object ID (OID), 1

P

passwd command, 23, 24, 42, 43, 148

passwords

- administrative, 23, 41
- default, 23, 27, 28, 41
- public key for, 24, 42

performance statistics, 108, 116

perfstats command, 108, 150

placement algorithm, 17

power failure, recovering from, 55

Power LED, 10

power switch, 11

powering up a cell, 55

privileges, write, 33

public key, 24, 42

Q

query

- improving performance of, 82, 86

R

rack

- filler panel position, 5
- populating, 8

- reboot command, 55, 153
- rebooting a cell, 55
- recovery from disaster, 121
- registering system, 64
- reset switch, 10

S

- schema
 - about, 67, 68
 - configuring using CLI, 95
 - configuring using GUI, 96
 - displaying, 97
 - DTD for, 69
 - example, 71
 - file structure, 68
 - using GUI to display, 96
 - virtual file system views in, 93
- sensors command, 113, 154
- service node IP address, 40
- shutdown command, 54, 156
- shutting down a cell, 54
- software versions, 109
- space usage, 116
- ssh connection, 23, 24, 43
- statistics
 - disk, 119
 - performance, 108, 116
- status of disks, 111
- storage access
 - administrative IP address for, 18
 - data IP address for, 18
- storage nodes, 9
- switches
 - components, 14
 - connections, 14
 - features, 14
 - installation, 5
 - locator, 11
 - node connections, 5
 - power, 11
 - rack location, 5
 - reset, 10
 - service node connections to, 14
- sysstat command, 95, 105, 157
- system
 - about, 1

- displaying performance statistics, 116
- displaying software versions on, 117
- displaying space usage, 116
- features, 2
- metadata, 17
- system access
 - about, 37
 - configuring, 37
 - configuring administrative IP address for, 38
 - configuring data IP address for, 39
 - configuring service node IP address for, 40
- system hardware, 3
- system namespace, 76
- system nodes, 9
- system notifications, 61
- system registration, 64
- system space usage, 116, 118
- system status, 105
- system time, checking, 50

T

- tables
 - about, 77
 - checklist for planning, 81
 - example, 78
 - planning, 79, 87
 - rows example, 80
- tape backup, 107, 121, 159
- task overview, 34
- temperature, 113
- time, checking, 50
- types of metadata, 74

V

- version command, 109, 160
- version of software, 109
- virtual file system views
 - about, 87
 - in schema file, 93
 - including additional file attributes in, 90
 - including archives in, 91
 - viewing files at higher levels in the directory structure in, 93
- voltage, 113

W

WebDAV

- about, 19, 88
 - access to data through, 19
 - browsing files with, 19, 88
 - example, 88
 - features, 88
 - metadata attributes and properties for, 90
- wipe command, 58, 162
- wiping data from system, 58
- writable namespaces, 75
- write privileges, 33

