Sun Enterprise™ 10000 SSP 3.5 User Guide

Please
Recycle

Adobe PostScript

Contents

Figures

Tables

Preface

The *Sun Enterprise 10000 SSP 3.5 User Guide* describes the System Service Processor (SSP), which enables you to monitor and control the Sun Enterprise™ 10000 system.

How This Book Is Organized

This document contains the following chapters:

Chapter 1 introduces the System Service Processor (SSP).

Chapter 2 introduces the Hostview Graphical User Interface (GUI)

Chapter 3 describes how to create, remove, rename, and bring up domains, and also how to get status information on a domain.

Chapter 4 describes how to use `netcon`(1M) and `netcontool`(1M).

Chapter 5 describes how to control the system power resources from within Hostview or from the command line, to control the peripherals power resources from the command line, and to monitor the power levels in Hostview.

Chapter 6 describes how to administer the thermal conditions from within Hostview and how to monitor the fans from within Hostview.

Chapter 7 describes how to configure components out of the system using the blacklist file.

Chapter 8 describes how automatic failover from the main to spare SSP works.

Chapter 9 provides information on the use of dual control boards and the control board failover process.

Chapter 10 provides more detailed information for system administrators interested in how the SSP works. Included are descriptions of the SSP booting process, SSP daemons, and failover conditions.

Appendix A describes miscellaneous SSP procedures, such as how to deconfigure the SSP.

Before You Read This Book

This manual is intended for Sun Enterprise 10000 system administrators, who have a working knowledge of UNIX® systems, particularly those based on the Solaris™ operating environment. If you do not have such knowledge, you should first read the Solaris system administrator documentation collection provided with this system, and consider UNIX system administration training.

Using UNIX Commands

This document does not contain information on basic UNIX commands and procedures such as shutting down the system, booting the system, and configuring devices.

See one or more of the following for this information:

■ Online documentation for the Solaris software environment, particularly those dealing with Solaris system administration
■ Other software documentation that you received with your system

Typographic Conventions

TABLE P-1   Typographic Conventions

Typeface or Symbol	Meaning	Examples
AaBbCc123	The names of commands, files, and directories; on-screen computer output.	Edit your `.login` file. Use `ls -a` to list all files. `% You have mail.`
AaBbCc123	What you type, when contrasted with on-screen computer output.	`% `**`su`** `Password:`
AaBbCc123	Book titles, new words or terms, words to be emphasized. Command-line variable; replace with a real name or value.	Read Chapter 6 in the *User Guide*. These are called *class* options. You *must* be `root` to do this. To delete a file, type `rm` *filename*.

Shell Prompts

TABLE P-2   Shell Prompts

Shell	Prompt
C shell	*machine_name*%
C shell superuser	*machine_name*#
Bourne shell and Korn shell	$
Bourne shell and Korn shell superuser	#

SSP Command Syntax

SSP commands ignore unrecognized parameters used on the command line.

Related Documentation

TABLE P-3   Related Documentation

Application	Title	Part Number
Installation	*Sun Enterprise 10000 SSP 3.5 Installation Guide and Release Notes*	816-3626
Reference (man pages)	*Sun Enterprise 10000 SSP 3.5 Reference Manual*	806-7614
Release Notes	*Sun Enterprise 10000 SSP 3.5 Installation Guide and Release Notes*	816-3626
Other	*Sun Enterprise 10000 Capacity on Demand 1.0 Administrator Guide*	806-2190
	Sun Enterprise 10000 Capacity on Demand 1.0 Reference Manual	806-2191
	Sun Enterprise 10000 Dynamic Reconfiguration User Guide	816-3627
	Sun Enterprise 10000 Dynamic Reconfiguration Reference Manual	806-7617
	Sun Enterprise 10000 InterDomain Networks User Guide	806-4121
	Sun Enterprise 10000 DR Configuration Guide	816-3630
	Sun Enterprise 10000 IDN Configuration Guide	806-5230
	Sun Enterprise 10000 IDN Error Messages	806-5231
	Sun Enterprise Server Alternate Pathing 2.3.1 User Guide	806-4150
	Sun Enterprise Server Alternate Pathing 2.3.1 Reference Manual	806-4151

If you need information on security considerations, contact your Sun sales professional.

Accessing Sun Documentation Online

A broad selection of Sun system documentation is located at:

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

A complete set of Solaris documentation and many other titles are located at:

`http://docs.sun.com`

Sun Welcomes Your Comments

We are interested in improving our documentation and welcome your comments and suggestions. You can email your comments to us at:

`docfeedback@sun.com`

Please include the part number (816-3624-10) of your document in the subject line of your email.

Introduction to the SSP

The System Service Processor (SSP) is a SPARC™ workstation or SPARC server that enables you to control and monitor the Sun Enterprise 10000 system. You can use a Sun Ultra™ 5 or Sun Enterprise 250 workstation, or a Sun Netra™ T1 server as an SSP. In this book, the SSP workstation or server is simply called the *SSP*. The SSP software packages must be installed on the SSP. In addition, the SSP must be able to communicate with the Sun Enterprise 10000 system over an Ethernet connection.

The Sun Enterprise 10000 system is often referred to as the *platform*. System boards within the platform may be logically grouped together into separately bootable systems called *Dynamic System Domains*, or simply *domains*. Up to 16 domains may exist simultaneously on a single platform. (Domains are introduced in this chapter, and are described in more detail in Chapter 3.) The SSP lets you control and monitor domains, as well as the platform itself.

Note – SSP 3.5 supports an OpenSSP environment in which certain types of lightweight, third-party applications can be run on the SSP. However, your SSP must meet the OpenSSP requirements described in the *SSP 3.5 Installation Guide and Release Notes.*

If you have a Sun Enterprise 10000 Capacity on Demand system, refer to the *Sun Enterprise 10000 Capacity on Demand 1.0 Administrator Guide* and *Sun Enterprise 10000 Capacity on Demand 1.0 Reference Manual*. If you are not sure if your system is a Capacity on Demand system, you can type the following command to determine whether the Capacity on Demand packages are installed:

```
ssp# pkginfo | grep SUNWcod
application SUNWcod        Capacity On Demand (COD)
application SUNWcodmn      Capacity On Demand (COD) Manual Pages
```

SSP Features

SSP 3.5 software can be loaded only on Sun workstations or Sun servers running the Solaris 7, 8, or 9 operating environment with the Common Desktop Environment (CDE). SSP 3.5 software is compatible with Sun Enterprise 10000 domains that are running the Solaris 2.5.1, 2.6, 7, 8, or 9 operating environments. The commands and GUI programs that are provided with the SSP software can be used remotely.

The SSP enables the system administrator to perform the following tasks:

- Perform an emergency shutdown in an orderly fashion. For example, the SSP software automatically shuts down a domain if the temperature of a processor within that domain rises above a preset level.
- Create domains by logically grouping system boards together. Domains are able to run their own operating system and handle their own workload. See Chapter 3.
- Boot domains.
- Dynamically reconfigure a domain so that currently installed system boards can be *logically* attached to or detached from the operating system while the domain continues running in multiuser mode.

  This feature is known as *Sun Enterprise 10000 dynamic reconfiguration (DR)*. You can also perform an automated dynamic reconfiguration (ADR) of domains. (A system board can easily be *physically* swapped in and out when it is not attached to a domain, even while the system continues running in multiuser mode.)

  SSP 3.5 supports two different models for dynamic reconfiguration on domains:

  - DR model 2.0 — Uses the dr_daemon(1M) to control DR operations on domains. You can use Hostview, dr(1M) shell, or ADR commands on the SSP to perform DR operations. SSP 3.5 supports DR model 2.0 on domains that run the Solaris 2.5.1, 2.6, 7, or 8 operating environments.
  - DR model 3.0 — Uses the domain configuration server, dcs(1M), to control DR operations on domains. You use the ADR commands on the SSP to perform DR operations. SSP 3.5 supports DR model 3.0 on domains that run the Solaris 9 operating environment or the Solaris 8 10/01 or 2/02 operating environments.

    DR model 3.0 domains also interface with the Reconfiguration Coordination Manager (RCM), which enables you to coordinate DR operations with other applications, such as database, clustering, and volume management software running on a Sun Enterprise 10000 domain. For details on RCM, refer to the *System Administration Guide: Basic Administration* in the Solaris 9 System Administrator Collection.

  For details on the DR models and operations, refer to the *Sun Enterprise 10000 Dynamic Reconfiguration User Guide*.

- Assign paths to different controllers for I/O devices, which enables the system to continue running in the event of certain types of failures. This functionality is referred to as alternate pathing or multipathing.

  For alternate pathing functionality compatible with DR model 2.0, use the Alternate Pathing software as described in the *Sun Enterprise Server Alternate Pathing User Guide.*

- For multipathing functionality compatible with DR model 3.0, use the IP multipathing (IPMP) software included with the Solaris operating environment and the Sun StorEdge™ Traffic Manager (also referred to as MPxIO) software for I/O multipathing.

  For further information on IPMP, refer to the *System Administration Guide: IP Services* documentation available with the Solaris 9 operating environment release or the *IP Network Multipathing Administration Guide* available with the Solaris 8 operating environment release.

  For details on the Sun StorEdge Traffic Manager, refer to the *Sun StorEdge Traffic Manager Software Installation and Configuration Guide* available on the Sun Network Storage Solutions web site (`http://www.sun.com/storage/san`).

Note – The Sun StorEdge Traffic Manager software may not support automatic path switching for all devices. For details, refer to the *Sun StorEdge Traffic Manager Software Installation and Configuration Guide.* If you require automatic path switching for devices that are not supported, use the Alternate Pathing software with DR model 2.0.

- Monitor and display the temperatures, currents, and voltage levels of one or more system boards or domains.
- Monitor and control power to the components within a platform.
- Execute diagnostic programs such as power-on self-test (POST).

In addition, the SSP environment:

- Warns you of impending problems, such as high temperatures or malfunctioning power supplies.
- Monitors a dual SSP configuration for single points of failure and performs an automatic failover from the main SSP to the spare or from the primary control board to the spare control board, depending on the failure condition detected.
- Notifies you when a software error or failure has occurred.
- Automatically reboots a domain after a system software failure (such as a panic).
- Keeps logs of interactions between the SSP environment and the domains.
- Provides support for InterDomain Networks (IDN).
- Provides support for the Sun Enterprise 10000 dual grid power option.

System Architecture

The Sun Enterprise 10000 platform, the SSP, and other workstations communicate over Ethernet (FIGURE 1-1). SSP operations can be performed by entering commands on the SSP console or by remotely logging in to the SSP from another workstation on the local area network. Whether you log in to the SSP remotely or locally, you must log in as user ssp and provide the appropriate password if you want to perform SSP operations (such as monitoring and controlling the platform).
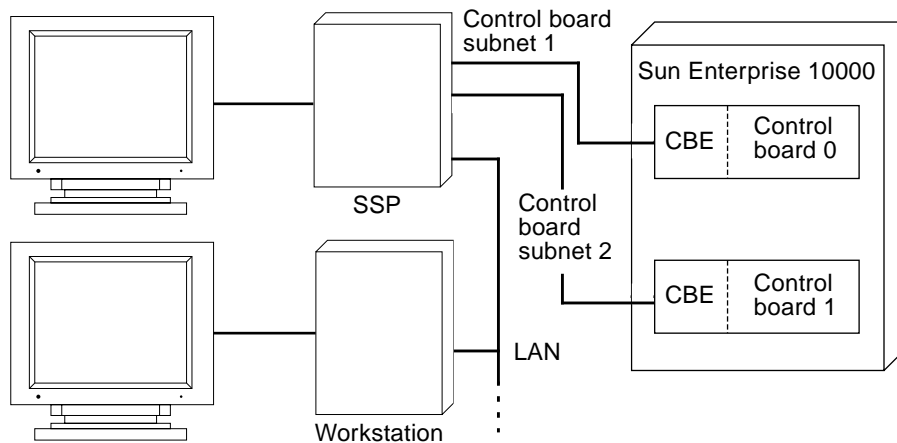
FIGURE 1-1   Sun Enterprise 10000 System and Control Boards

Dual control boards are supported within the Sun Enterprise 10000 platform. Each control board runs a control board executive (CBE) that communicates with the SSP over a private network. One control board is designated as the primary control board, and the other is designated as the spare control board. If the primary control board fails, the failover capability automatically switches to the spare control board as described in Chapter 9.

The SSP software handles most control boards as active components, and you need to check the system state before powering off any control board. For details, see Chapter 9.

SSP User Environment

You can interact with the SSP and the domains on the Sun Enterprise 10000 by using SSP commands and the SSP GUI programs.

▼ To Begin Using the SSP

1. **Boot the SSP.**

2. **Log in to the SSP as user** `ssp` **and type:**

```
ssp% tail -f $SSPLOGGER/messages
```

3. **Wait until you see the following message:**

```
Startup of SSP as MAIN complete
```

At this point you can begin using SSP programs such as Hostview and `netcontool`(1M).

SSP 3.5 Window

An SSP window provides a command line interface to the Solaris operating environment and SSP 3.5 environment (FIGURE 1-2).

SSP or other workstation display
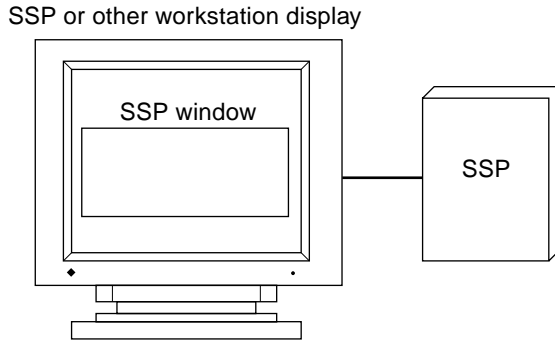
FIGURE 1-2   SSP Window

▼ To Display an SSP Window Locally in the Common Desktop Environment (CDE)

1. **Log in to the SSP as user** `ssp`**.**

2. **Open an SSP window using one of the following methods:**
   - From the CDE front panel under the Solaris 7, 8, or 9 operating environment, select the Hosts subpanel, and then select This Host.
   - From the Workspace Menu (right mouse click) choose Programs and then select Terminal.

▼ To Display an SSP Window Remotely

1. **Use the** `rlogin`**(1) command to remotely log in to the SSP 3.5 machine as user** `ssp`**, and enter the** `ssp` **password.**

2. **When prompted, type in the name of the platform or domain you wish to work with, and then press Return.**

   The `SUNW_HOSTNAME` environment variable is set to the value you enter.

SSP Console Window

The SSP console window is the console for the SSP workstation or server. The system uses it to log operating system messages.

▼ To Display an SSP Console Window Locally with CDE

1. Log in to the SSP as user `ssp`.

2. Open an SSP window using one of the following methods:

- From the Workspace Menu (right mouse click) choose Programs and then select Console.
- From the CDE front panel under the Solaris 7, 8, or 9 operating environment, select the Hosts subpanel, and then select Console.

Network Console Window

The network console window, or `netcon`(1M) window, receives system console messages (operating system messages) from a domain (FIGURE 1-3).
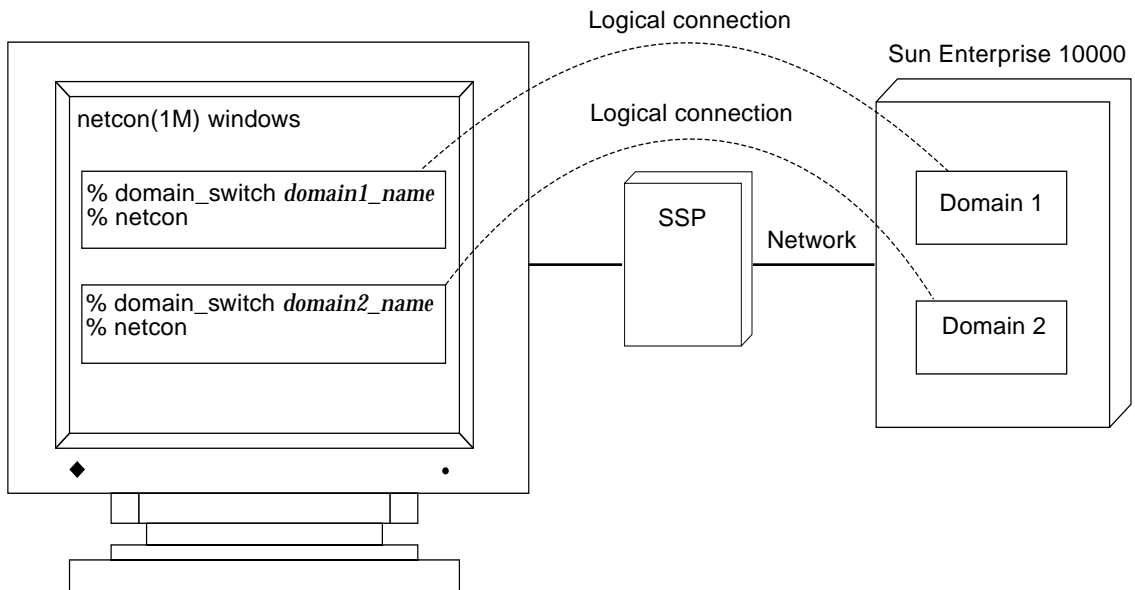
FIGURE 1-3 `netcon`(1M) Window

A `netcon`(1M) window behaves as if a console is physically connected to a domain. Domain console messages, such as those generated by dynamic reconfiguration operations, are displayed in the `netcon`(1M) window. For more information, see "Using netcon(1M)" on page 39 and the `netcon`(1M) man page.

Hostview

The Hostview program provides a graphical user interface (GUI) with the same functionality as many of the SSP commands (FIGURE 1-4).
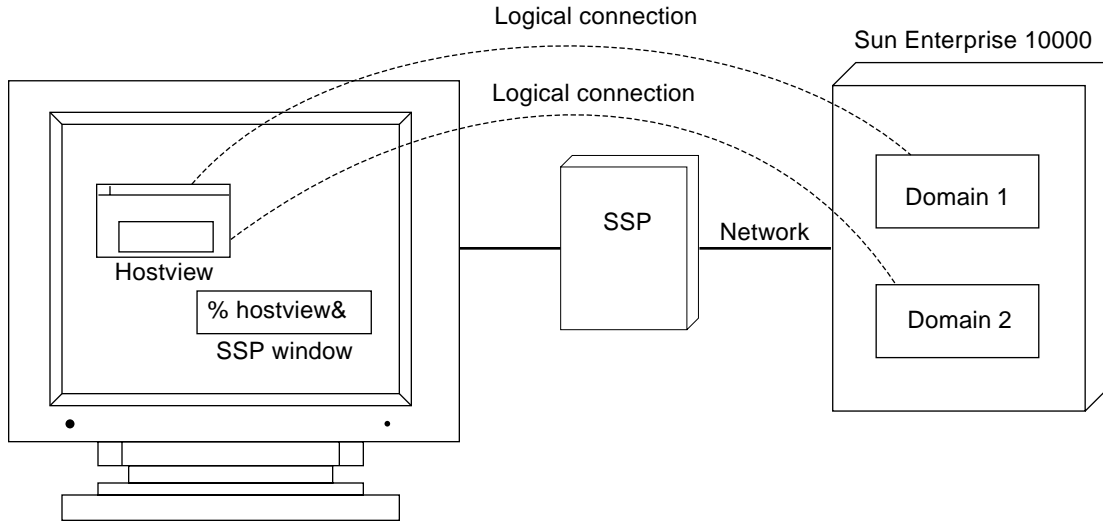
FIGURE 1-4 Hostview GUI Program

Hostview is described in detail in Chapter 2. It is also described in the hostview(1M) man page in the *Sun Enterprise 10000 SSP 3.5 Reference Manual* (man pages).

Hostview

This chapter describes Hostview, a GUI front-end to SSP commands.

Hostview enables you to perform administration operations such as:

- Dynamically grouping the system boards into domains. Each domain runs its own instance of the Solaris operating environment and has its own log messages file.
- Booting the Solaris operating environment for a domain.
- Accessing the SSP log messages file for each platform or domain.
- Remotely logging in to each domain.
- Displaying a `netcon`(1M) window for each domain.
- Editing the `blacklist`(4) file to enable or disable hardware components in a domain.
- Dynamically reconfiguring the boards within a platform, logically attaching or detaching them from the operating system. This feature is described in the *Sun Enterprise 10000 Dynamic Reconfiguration User Guide*.
- Powering the system boards on and off.
- Monitoring the temperature and voltage levels of hardware components.

If you want to run Hostview, you only need to run one instance for a given platform. However, it is possible to run more than one instance simultaneously (perhaps on different workstations) to work with the same platform.

If you have logged into the SSP environment from a remote login session, make sure your `DISPLAY` environment variable is set to your current display and that your `xhost` settings enable the SSP to display on your workstation (see `xhost`(1) in the *Solaris X Window System Reference Manual*).

▼ To Start Up Hostview From a Remote Login Session

1. Enable external hosts to display on your local workstation:

```
% xhost +
```

2. Log in to the SSP as user `ssp` **and type:**

```
ssp% hostview -display machine_name:0.0 &
```

▼ To Start Up Hostview From the Workspace Menu Locally on the SSP

● **From the Workspace Menu (right mouse button click), select SSP, and then select Hostview.**

This is available only when you use the SSP console, not when you use a remote login session to the SSP.

▼ To Start Up Hostview Under CDE From the Front Panel

● **Use one of the following methods:**

  ■ Click the `SSP` icon on the front panel. The icon shows a hand holding tools.
  ■ Click the arrow above the `SSP` icon on the front panel, and select Hostview.
  ■ Open an SSP window and type:

```
ssp% hostview &
```

Hostview Main Window

When you start Hostview, the main window is displayed (FIGURE 2-1).

FIGURE 2-1 Hostview Main Window

The menu bar on the main window provides the commands that you use to control the platform. The commands are described in "Main Window Menu Bar" on page 13.

The buttons on the main window (Power, Temperature, and Fans) bring up status details. The buttons are described in "Main Window Buttons" on page 17.

The rest of the main window provides a graphical view of the platform boards and buses. The system boards are named SB0 through SB15, and their processor numbers are shown. The control boards are named CB0 and CB1. The support boards are named CSB0 and CSB1. The buses are named ABUS0 through ABUS3, DBUS0, and DBUS1.

The system boards along the top of the display are arranged in the order they appear on the front side of the physical platform. The system boards along the bottom of the display are arranged in the order they appear on the back side of the physical platform.

If a system board is shown with no outline (FIGURE 2-2), the board is not part of a domain and is not currently selected.
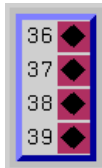
FIGURE 2-2 Unselected System Board (Domain independent)

If a system board is part of a domain (FIGURE 2-3), a colored outline surrounds it. The boards within a given domain all have outlines of the same color.
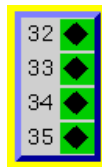
FIGURE 2-3 Unselected System Board (Domain dependent)

A black outline around the domain color outline indicates that a board is selected (FIGURE 2-4). (There are several reasons why you select a board in Hostview. For example, you could select one or more boards, and then create a domain that is based on those boards.)

FIGURE 2-4 Selected System Board (Domain dependent)

The processors within the boards are numbered 0 through 63. The processor symbols (diamond, circle, and so forth) indicate the state of the processors and are described in "Main Window Processor Symbols" on page 18.

Selecting Items in the Main Window

You can select one or more boards in the Hostview main window. You can also select one domain in the main window. You must select a set of boards prior to performing certain operations, such as creating a domain.

■ To select a single board, click on it with the left mouse button. The selected board is indicated by a black outline, and all other boards are deselected.

■ To select additional boards, click on them with the middle mouse button. You can also deselect a currently selected board by clicking on it with the middle mouse button. (The middle mouse button toggles the selection status of the board without affecting the selection status of any other board.)

■ To select a domain, click on a board within that domain with the left mouse button. Note that you can select boards from different domains (using the middle mouse button), but the selected domain will correspond to the board that you selected with the left mouse button.

Main Window Menu Bar

The items on the main Hostview menu are described in TABLE 2-1.

TABLE 2-1   Hostview Menu Items

Menu	Submenu Items	Description
File	SSP Logs	Displays a window that shows the SSP messages for a domain or for the platform. For more information, see "SSP Log Files" on page 21.
	Quit	Terminates Hostview.
Edit	Blacklist File	Enables you to specify boards and CPUs to be blacklisted.
Control	Power	Displays a window that enables you to use the power(1M) command. See "To Power Components On or Off From Within Hostview" on page 49.
	Bringup	Displays a window that enables you to run bringup(1M) on a domain. See "To Bring up a Domain From Within Hostview" on page 30.

TABLE 2-1 Hostview Menu Items *(Continued)*

Menu	Submenu Items	Description
	Fan	Displays a window that enables you to run the fan(1M) command to control the fans within the platform.
Configuration	Board	Enables you to attach and detach system boards. This feature is described in the *Sun Enterprise 10000 Dynamic Reconfiguration User Guide.*
	Domain	Provides a menu with several choices. The menu choices enable you to create domains, remove domains, rename domains, obtain the status of domains, and view the history of domains. A domain consists of one or more system boards running the same operating system kernel. Domains function independently of each other. Each domain can carry its own workload and has its own log messages file. For more information, see "To Create Domains From Within Hostview" on page 24 and "To Remove Domains From Within Hostview" on page 28.
Terminal	netcontool	Displays a window that provides a graphical interface to the netcon(1M) command, enabling you to open a network console window for a domain. This menu item is equivalent to executing the netcontool(1M) command. See "Using netcon(1M)" on page 39.
	Connect to SSP	Provides menu choices that enable you to display an SSP Window in xterm, dtterm, cmdtool, or shelltool format with a platform or domain as its host. Select a domain (by selecting any system board within that domain) before choosing this option.
	Connect to Domain	Provides menu choices that enable you to remotely log in to the selected platform or domain in an xterm, dtterm, cmdtool, or shelltool window. Select a domain (by selecting any system board within that domain) before choosing this option.

TABLE 2-1   Hostview Menu Items *(Continued)*

Menu	Submenu Items	Description
View	All Domains	Displays the boards within all domains, as well as any boards that are not part of a domain. (A board can be present without being part of a domain, although a board cannot be used when it is not part of a domain.)
	Individual Domains	When you select an individual domain, only the boards within that domain are displayed. Note that the color of the outline used to designate a given domain is also used as the background color for that domain in the menu. The system board numbers for the boards that belong to each domain are shown in square brackets.
Help	*topic*	Provides online help information on several topics.

Help Window

When you choose a topic from the Help menu, the Hostview - Help window is displayed (FIGURE 2-5).
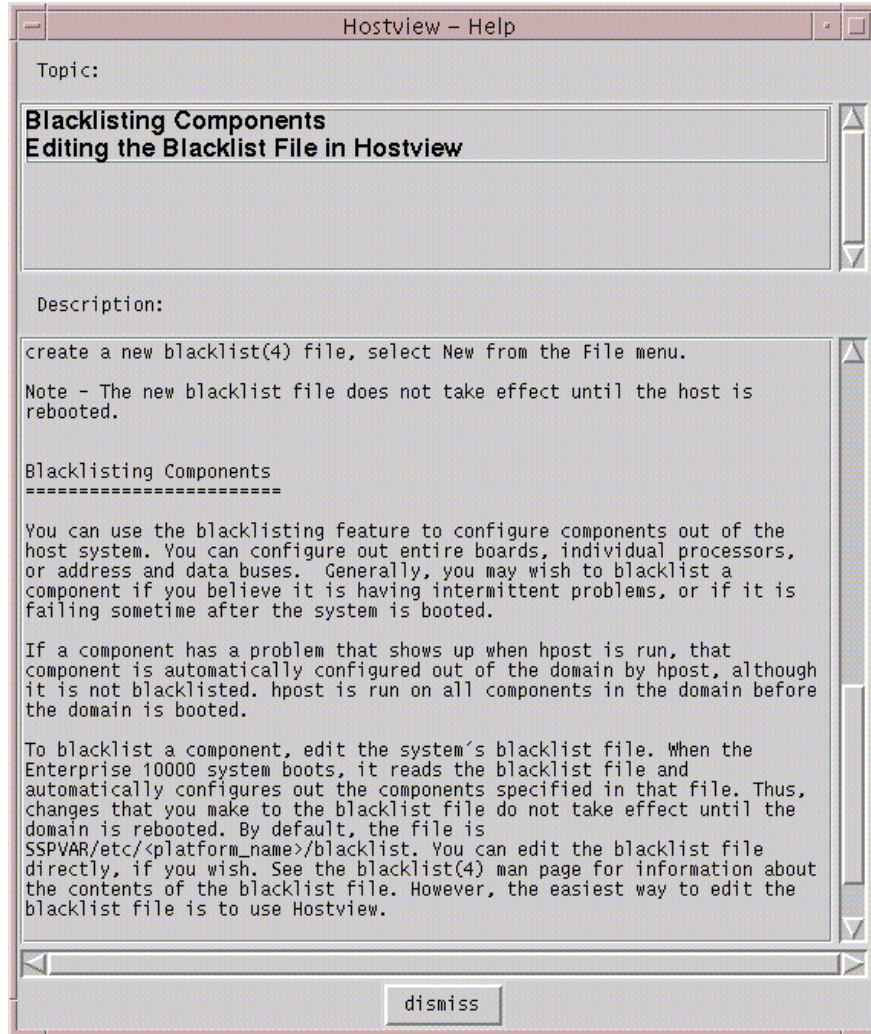
FIGURE 2-5   Hostview Help Window

You can select the desired topic in the upper pane. The corresponding help information is displayed in the lower pane.

Main Window Buttons

The main Hostview window contains the buttons described below. If an out-of-boundary condition exists or an error has occurred, one or more of these buttons turn red.

The Power button (FIGURE 2-6) displays the Power Control and Status window, which enables you to view the power status for the platform. See "To Power Components On or Off From Within Hostview" on page 49.

FIGURE 2-6 Power Button

The Temperature button (FIGURE 2-7) displays the Thermal Status window, which enables you to view the temperature status for the boards and components within the platform. See "To Monitor Thermal Conditions From Within Hostview" on page 55.
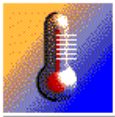
FIGURE 2-7 Temperature Button

The Fan button (FIGURE 2-8) displays the Fan Status window, which enables you to view the status of the fans within the platform. See "To Monitor Fans From Within Hostview" on page 57.

FIGURE 2-8 Fan Button

When certain error conditions occur, the Failure button (FIGURE 2-9) turns red. If you click a red Failure button, a window is displayed showing the error condition(s) that have occurred.

FIGURE 2-9   Failure Button

TABLE 2-2 describes the types of error conditions that are trapped by this mechanism.

TABLE 2-2   Error Conditions

Error	Description
Host panic recovery in progress	The operating system on a domain has failed and is recovering.
Heartbeat failure recovery in progress	The SSP was not receiving updated platform or domain information as expected.
Arbitration stop recovery in progress	A parity error or other fatal error has occurred, and the domain is recovering. See *arbitration stop* in the glossary.

Main Window Processor Symbols

In the main window, the shape and background color of a processor symbol indicate the status of that processor. For example, a diamond on a green background indicates the processor is running the operating system.

TABLE 2-3 lists the shapes and what the processor is running for each shape.

TABLE 2-3   Processor Symbol Shapes

Shape	Processor running
Diamond ◆	Operating system
Circle ●	`hpost`(1M)
Square ■	`download_helper`
Triangle ▲	OBP
Question mark ?	Unknown program

TABLE 2-4 lists the possible colors for processor symbols and the processor state indicated by each color.

TABLE 2-4   Processor Color Scheme

Color	State
Green	Running
Maroon	Exiting
Yellow	Prerun (The OS is currently being loaded.)
Blue	Unknown
Black	Blacklisted (The processor is unavailable to run programs or diagnostics.)
Red	Redlisted (The processor is unavailable to run programs or diagnostics and its state may not be changed.)
White	Present but not configured (The processor is unavailable, but not blacklisted or redlisted. One example is a board that has been hot-swapped in, but it has not yet been attached to the operating system.)

Hostview Resources

In the Hostview main window, boards that are in the same domain have the same color outline. If you want to change the domain colors or if your workstation does not use the default colors supported by Hostview, you can configure the colors used for each domain. Put the following resources in your $HOME/.Xdefaults file and modify the specified colors using valid color names:

```
Hostview*domainColor0: white
Hostview*domainColor1: orange
Hostview*domainColor2: yellow
Hostview*domainColor3: pink
Hostview*domainColor4: brown
Hostview*domainColor5: red
Hostview*domainColor6: green
Hostview*domainColor7: violet
Hostview*domainColor8: purple
Hostview*domainColor9: mediumaquamarine
Hostview*domainColor10: yellowgreen
Hostview*domainColor11: maroon
Hostview*domainColor12: cyan
Hostview*domainColor13: darkgoldenrod
Hostview*domainColor14: navyblue
Hostview*domainColor15: tan
```

You can use the showrgb(1) command (for details on this command, see the *Solaris X Window System Reference Manual*) to list the valid domain color names on your display workstation. If you specify an invalid domain color in the .Xdefaults file, an error is generated and the following occurs:

■ Domain outlines for the invalid color and subsequent domain colors are not displayed in the main Hostview window

■ Domain names are not listed in the View menu

Hostview Performance Considerations

Each instance of Hostview requires up to 10 Mbytes of the available swap space in the SSP. Before running multiple copies of Hostview, make sure the SSP has sufficient swap space available. For example, if you plan to run three instances of Hostview, make sure you have at least 30 Mbytes of swap space.

SSP Log Files

The SSP processes log, informational, warning, and error messages to a variety of log files.

Messages for the platform that are not specific to a domain are logged in the file:

```
$SSPLOGGER/messages
```

If you have a spare SSP, note that messages logged in the platform message file on the main SSP and any domain messages are also logged on the spare SSP.

Messages for a particular domain are logged in the file:

```
$SSPLOGGER/domain_name/messages
```

where *domain_name* is the host name of the domain for which the error occurred.

The SSP environment variables, such as SSPLOGGER, are described in "Environment Variables" on page 108.

The message format and logging level are specified in the `$SSPLOGGER/.logger` and the `/etc/syslog.conf` files on the SSP. Do not change the default values in these files unless your service provider instructs you to do so.

Note – During installation of the Solaris operating environment on a domain, the domain `/etc/syslog.conf` file is modified so that system messages are routed to the SSP `/var/adm/messages` file and the domain `/var/adm/messages` file.

▼ To View a `messages` File From Within Hostview

1. **Select the appropriate board.**
   - If you want to view the `messages` file for a particular domain, select that domain in the main Hostview window (by clicking on a board from that domain with the left mouse button).
   - If you want to view the `messages` file for the platform, make sure that no domain is selected.

2. Choose SSP Logs from the File menu.

The SSP Logs window is displayed (FIGURE 2-10).


```
                              SSP Logs

        Domain Name: xf4
           Command: tail -f /var/opt/SUNWssp/adm/xf4/messages

Sep  1 08:53:22 xf4-ssp bringup-(xf4): INFO : BRINGUP COMMAND STARTED
Sep  1 08:53:35 xf4-ssp bringup-(xf4): INFO : domain xf4-b2 is inactive
Sep  1 08:53:40 xf4-ssp bringup-(xf4): INFO : configuring the centerplane, POST
args  -17 -C
Sep  1 08:53:41 xf4-ssp bringup-(xf4): INFO : Starting: hpost  -17 -C
Sep  1 08:55:44 xf4-ssp bringup-(xf4): INFO : Starting: obp_helper -m 1  -A off
Sep  1 08:55:44 xf4-ssp bringup-(xf4): INFO : Starting: netcon_server -p 1
Sep  1 08:55:51 xf4-ssp bringup-(xf4): INFO : BRINGUP COMMAND FINISHED



        execute          dismiss          abort            help
```

FIGURE 2-10 SSP Logs Window

The Domain Name field shows the name of the domain that you selected. The messages file is displayed in the main panel of the window.

Domain Administration

The SSP supports commands that let you logically group system boards into Dynamic System Domains, or simply *domains*, which are able to run their own operating system and handle their own workloads. Domains can be created and deleted without interrupting the operation of other domains. You can use domains for many purposes. For example, you can test a new operating system version or set up a development and testing environment in a domain. In this way, if problems occur, the rest of your system is not affected.

You can also configure several domains to support different departments, with one domain per department. You can temporarily reconfigure the system into one domain to run a large job over the weekend.

Domain Configuration Requirements

You can create a domain out of any group of system boards, provided the following conditions are met:

■ The boards are present and not in use in another domain.

■ At least one board has a network interface.

■ The boards have sufficient memory to support an autonomous domain.

■ The name you give the new domain is unique (as specified in the domain_create(1M) command) and this name matches the host name of the domain to be booted (as specified by the SUNW_HOSTNAME environment variable).

■ You have an eeprom.image file for the domain that was shipped to you by the factory. If your eeprom.image file has been accidentally deleted or corrupted and you do not have a backup, see "To Recreate the eeprom.image File" on page 27.

There must be at least one boot disk connected to one of the boards that will be grouped together into a domain. Alternatively, if a domain does not have its own disk, there must be at least one network interface so that you can boot the domain from the network.

▼ To Create Domains From Within Hostview

Note – Before proceeding, read the requirements in the previous section, "Domain Configuration Requirements" on page 23. If the system configuration must be changed to meet any of these requirements, call your service provider. Also, after you create a domain, you must update /etc/hosts to reflect the new name of the domain.

---

1. **Click the left mouse button on the first board.**

2. **Click the middle mouse button on any additional boards.**

   Ensure that the boards you select do not currently belong to any domain.

3. **Choose Domain then Create from the Configuration menu.**

   The Create Domain window is displayed (FIGURE 3-1).
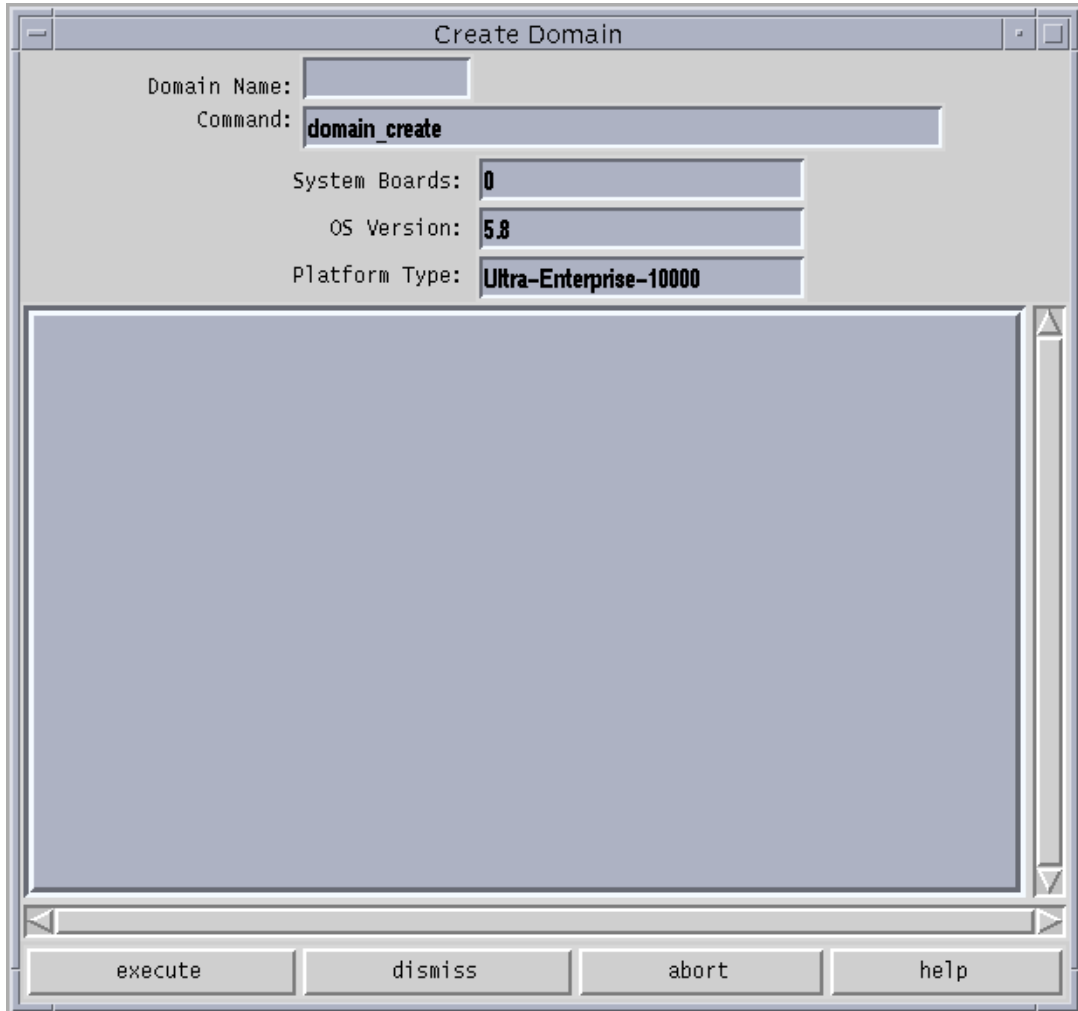
FIGURE 3-1 Create Domain Window

4. **You must be sure to specify the proper OS version number for the domain you are creating. The default is 5.8. Edit this version number, if necessary, to reflect the version of the operating system for the domain you are creating.**

5. **Type the domain name.**

   The name of the domain must be the one given to you by the factory and contained in the eeprom.image file. It cannot be an arbitrary name.

6. **If all other fields are acceptable, click execute.**

   Note that the System Boards field indicates the boards that you selected in the main Hostview window. The default OS version and the default platform type are shown.

   If Hostview successfully executes the command, it displays the message `Command completed` in the informational panel of the window.

   Note – Hostview can run only one `create` or `remove` command at a time. If you attempt to execute a second `create` or `remove` command before the first has completed, your second attempt fails.

▼ To Create Domains From the Command Line

Note – Before proceeding, see "Domain Configuration Requirements" on page 23. If the system configuration must be changed to meet any of these requirements, contact your service provider.

---

1. **In an SSP window, type:**

   ```
   ssp% domain_create -d domain_name -b system_board_list -o os_version -p
   platform_name
   ```

   where:

   domain_name is the name you want to give to the new domain. It should be unique among all Sun Enterprise 10000 systems controlled by the SSP.

   system_board_list specifies the boards that are to be part of this domain. The specified system boards must be present and not in use. Each domain must have a network interface, disk interface, and sufficient memory to support an autonomous system. List the board numbers, separated by commas or spaces, for all boards you want to include.

   os_version is the version of the operating system to be loaded into the domain.

   platform_name is the name of the platform managed by the SSP.

2. **Optionally, create a new SSP window for the domain, as described in "SSP 3.5 Window" on page 5.**

   Use the `domain_switch`(1M) command to set the `SUNW_HOSTNAME` environment variable to the new domain name.

▼ To Recreate the `eeprom.image` File

Note – You cannot create a domain if you do not have the corresponding `eeprom.image` file. The `eeprom.image` files for the domains you ordered are shipped to you by the factory. If you accidentally delete an `eeprom.image` file or your boot disk is corrupted, and you do not have a backup copy of your `eeprom.image` file, you can contact your Sun service representative to recreate it. Alternatively, you may be able to recreate the `eeprom.image` file if you have the original serial number and the EEPROM key. In this case, follow the steps in this procedure.

---

1. **Log in to the SSP as user** `ssp`.

2. **Recreate the** `eeprom.image` **file.**

Note – All `key` and `host_id` numbers are case sensitive and must be entered exactly as they are received.

---

a. **For the first domain, type:**

```
ssp% domain_switch domain_name
ssp% sys_id -k key -s serial_number
```

   where:

   domain_name is the hostname of the domain

   key is the `eeprom` key number

   serial_number is the number provided with the *key* in the form of 0XA65*xxx*

b. **For all subsequent domains, type:**

```
ssp% domain_switch domain_name
ssp% sys_id -k key -h hostid
```

   where:

   domain_name is the hostname of the domain.

   key is the `eeprom` key number.

   hostid is the number provided with the *key* in the form of 0X80A66*xxx*.

3. **Check the result by typing:**

```
ssp% sys_id -d
```

In the following example, `49933C54C64C858CD4CF` is the `key` and `0x80a66e05` is the *host_id*:

```
ssp% domain_switch domain_name
ssp% sys_id -k 49933C54C64C858CD4CF -h 0x80a66e05
ssp% sys_id -d

 IDPROM in eeprom.image.domain_name

                   Format = 0x01
             Machine Type = 0x80
         Ethernet Address = 0:0:be:a6:6e:5
       Manufacturing Date = Wed Dec 31 16:00:00 1998
Serial number (machine ID) = 0xa66e05
                 Checksum = 0x3f
```

4. **Back up the SSP** `eeprom.image` **files to tape or disk where they can be accessed in case of SSP boot disk failure.**

▼ To Remove Domains From Within Hostview

1. **In the main Hostview window, click any board in the domain to be removed.**

2. **Choose Domain then Remove from the Configuration menu.**

   The Remove Domain window is displayed (FIGURE 3-2).
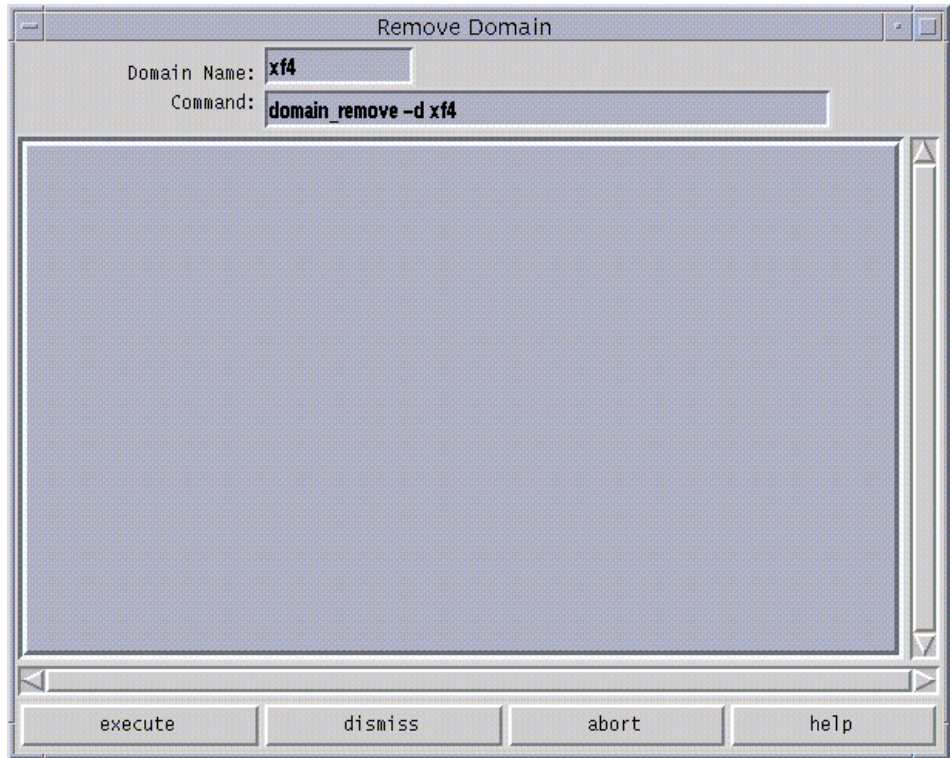
FIGURE 3-2 Remove Domain Window

3. **If the default** domain_remove**(1M) command is satisfactory, click the execute
button; otherwise, edit the command first.**

For help on the domain_remove(1M) command, click the help button. A help
window is displayed (see "Help Window").

4. **Specify whether or not domain subdirectories should be removed when you are
prompted to do so.**

The pathnames of the subdirectories are displayed. These subdirectories contain
domain-specific information such as message files, configuration files, and
hpost(1M) dump files. You can keep these directories if you still need the
information. It is easier to recreate a domain if you keep these directories.

Note – If the system cannot remove your domain, see domain_remove(1M) for a
list of potential errors.

▼ To Remove Domains From the Command Line

1. As user `ssp`**, type:**

```
ssp% domain_remove -d domain_name
```

The domain must not be running the operating system.

2. Specify whether or not domain subdirectories should be removed when you are prompted to do so.

The pathnames of the subdirectories are displayed. These subdirectories contain domain-specific information such as message files, configuration files, and `hpost`(1M) dump files. You can keep these directories if you still need the information. The domain can be recreated whether or not you keep this information.

3. Type `domain_status`**(1M) to verify that the domain was removed.**

Note – If the system cannot remove your domain, an error message is displayed. See `domain_remove`(1M) for a list of potential errors.

▼ To Bring up a Domain From Within Hostview

1. Use the mouse to select any system board belonging to the domain you want to bring up.

2. Choose Bringup from the Control menu.

A window is displayed that shows the name of the selected domain.

3. Click `Execute` **to perform the bringup.**

4. After the `bringup` **operation has completed, choose** `netcontool` **from the Terminal menu.**

5. Click the `Connect` **button to open a** `netcon`**(1M) window.**

6. If the OBP prompt appears (that is, the `OK` **prompt), boot the domain:**

```
OK boot boot_device
```

The domain should boot and then display the login prompt. Note that you can use the OBP command `devalias` to determine the alias for the disk you want to use as *boot_device*.

▼ To Bring up a Domain From the Command Line

Before you can bring up a domain from the command line in an SSP window, the system boards for the domain must be powered on. Ensure that the SUNW_HOSTNAME environment variable is set to the proper valid domain name.

1. **As user** ssp, **set the SUNW_HOSTNAME environment variable by typing:**

```
ssp% domain_switch domain_name
```

where *domain_name* is the name of the domain you want to bring up.

2. **Power on the power supplies for all of the boards in the domain:**

```
ssp%  power -on
```

3. **Bring up the domain by typing:**

```
ssp% bringup -A [off|on] [disk]
ssp% netcon
ok boot
```

The -A option is the autoboot option. If the autoboot option is on, the domain will automatically boot. If it is off, you need to explicitly boot the domain from the OBP prompt. For information on other command line options, see the bringup (1m) man page.

▼ To Obtain Domain Status From Within Hostview

● **Choose Domain then Status from the Configuration menu.**

The Domain Status window is displayed (FIGURE 3-3).
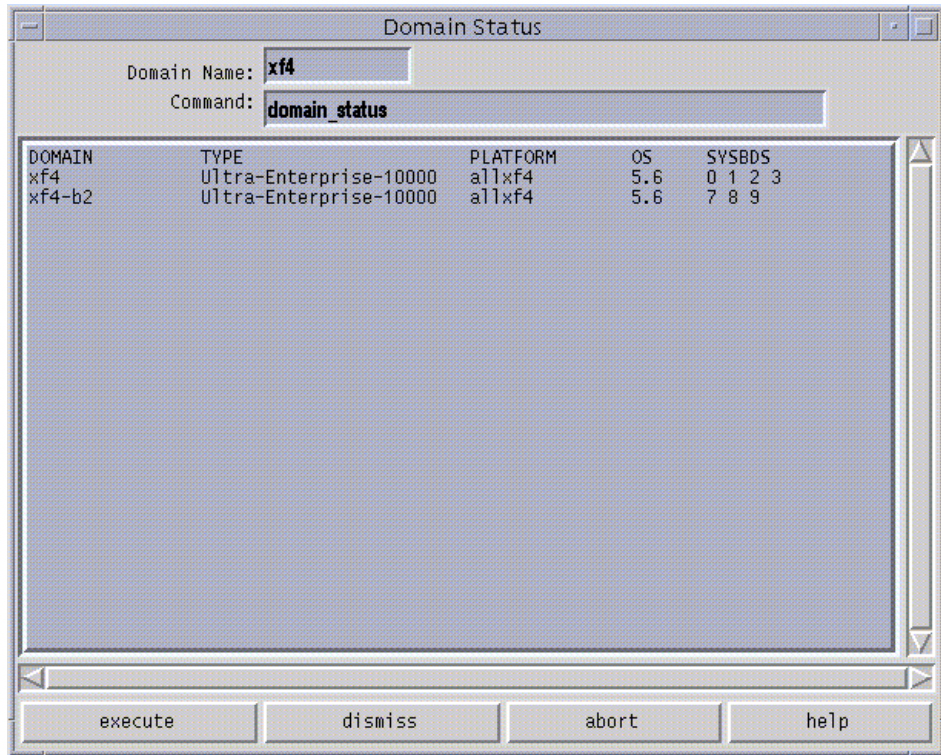
FIGURE 3-3   Domain Status Window

The status listing is displayed in the main panel of the window. The following table explains the columns in the Domain Status window.

Note – You can determine which DR model is running on your domains by entering the `-m` option after the `domain_status` command in the Command box, then clicking the execute button. The information displayed will include the DR model, which is either 2.0, 3.0, or unknown, as explained in TABLE 3-1.

TABLE 3-1   Domain Status Columns

Name	Description
DOMAIN	Lists the name of the domain.
TYPE	Lists the platform type. It can only have the value `Ultra-Enterprise-10000` in the current release.
PLATFORM	Lists the name of the platform. The platform name is set at the time the SSP packages are installed.
DR-MODEL	If you specify the `-m` option after the `domain_status` command in the Command box and then click the execute button, the DR model number is displayed for each domain. The DR model can be 2.0, 3.0, or unknown:
	DR model 2.0 uses the `dr_daemon`(1M) to control DR operations on the domain. DR model 2.0 is supported on domains running the Solaris 2.5.1, 2.6, 7, and 8 operating environments.
	DR model 3.0 uses the domain configuration server, `dcs`(1M) to control DR operations on the domain and interfaces with the Reconfiguration Coordination Manager (RCM) to coordinate DR events with other applications, such as database and system management tools. DR model 3.0 is supported on domains running the Solaris 9 or Solaris 8 10/01 and 2/02 operating environments.
	If a domain is down or DR was not configured correctly for the domain, the SSP cannot determine the DR model for the domain and lists the DR model as unknown.
	For details on the DR models, refer to the *Sun Enterprise 10000 Dynamic Reconfiguration User Guide*.
OS	Lists the operating system version for the domain.
SYSBDS	Lists the system boards that make up the domain.

▼ To Shut Down a Domain

1. **Log in to the domain as superuser and run the** `shutdown` **(1M) command.**

   A message indicates that the system has been halted.

▼ To Rename Domains From Within Hostview

Note – After you rename a domain, you must also update the standard host
configuration files to reflect the new name of the domain. For details, refer to the
Solaris documentation for the operating environment version that you are running
on the domain (for example, the Solaris 2.6, 7, or 8 User Collection and the Solaris
2.6 System Administrator Collection Vol 1, the Solaris 7 or 8 System Administrator
Collection, or the Solaris 9 System Administration Guides).

---

1. **Log in to the domain as superuser.**

2. **Run** `sys-unconfig` **(1M) to deconfigure the host.**

3. **Back up the** `eeprom.image` **files in the directory**
   `/var/opt/SUNWssp/.ssp_private/eeprom_save`**.**

4. **Change the host name in the NIS and the** `/etc/hosts` **files on the SSP to reflect
   the name of the new domain.**

5. **As user** `ssp`**, in the main Hostview window, select a board from the domain that
   you want to rename by clicking on it with the left mouse button.**

6. **Choose Domain then Rename from the Configuration menu.**

   The Rename Domain window is displayed (FIGURE 3-4).
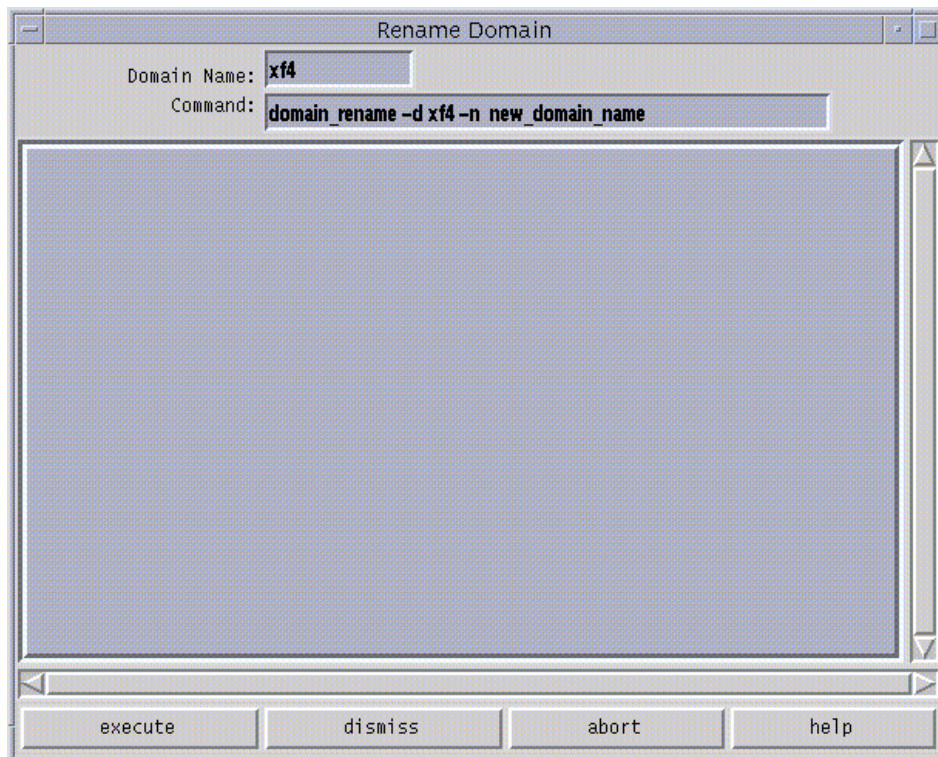
FIGURE 3-4   Rename Domain Window

7. **If the default** domain_rename**(1M) command is satisfactory, click the execute button. Otherwise, edit the command first.**

   For help on the domain_rename(1M) command, click the help button. A help window is displayed (see "Help Window").

8. **Bring up the domain using Hostview or the** bringup**(1M) command.**

   For details, see "To Bring up a Domain From Within Hostview" on page 30 and "To Bring up a Domain From the Command Line" on page 31.

9. **Start a** netcon**(1M) session and answer the prompts regarding the configuration of the domain.**

▼ To Rename Domains From the Command Line

> **Note –** After you rename a domain, you must also update the standard host configuration files to reflect the new name of the domain. For details, refer to the Solaris documentation for the operating environment version that you are running on the domain (for example, the Solaris 2.6, 7, or 8 User Collection and the Solaris 2.6 System Administrator Collection Vol 1, the Solaris 7 or 8 System Administrator Collection, or the Solaris 9 System Administration Guides).

1. **Log in to the domain as superuser.**

2. **Run** `sys-unconfig(1M)` **to deconfigure the host.**

3. **Back up the** `eeprom.image` **files in the directory** `/var/opt/SUNWssp/.ssp_private/eeprom_save`**.**

4. **Change the host name in the NIS and the** `/etc/hosts` **files on the SSP to reflect the name of the new domain.**

5. **As user** `ssp`**, rename the domain.**

   ```
   % domain_rename -d old_domain_name -n new_domain_name
   ```

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

6. **Bring up the domain using Hostview or the** `bringup`**(1M) command.**

   For details, see "To Bring up a Domain From Within Hostview" on page 30 and "To Bring up a Domain From the Command Line" on page 31.

7. **Start a** `netcon`**(1M) session and answer the prompts regarding the configuration of the domain.**

▼ To Change the Version of the Operating System for a Domain From the Command Line

1. **Log in to the domain as user** `ssp`**.**

2. Change the SunOS operating system version for the domain by using the `domain_rename` **command.**

```
% domain_rename -d domain_name -o new_OS_version
```

where

domain_name is the name of the domain to be changed

new_os_version is the version of the SunOS operating systems, for example, 5.5.1, 5.6, 5.7, 5.8, or 5.9, to be run on the domain.

`netcon` **and** `netcontool`

This chapter describes `netcon`(1M) and `netcontool`(1M), a GUI front-end to the `netcon`(1M) command. `netcontool`(1M) simplifies the process of configuring and bringing up `netcon`(1M) windows. You can also use the `netcon`(1M) command directly to display a `netcon`(1M) window. However, when using `netcon`(1M), you must know escape sequences to perform operations that can be performed by clicking on buttons under `netcontool`(1M).

Using `netcon`(1M)

The `netcon`(1M) command is similar to `netcontool`(1M) except that no GUI interface is provided, making it more functional for dial-in or other low-speed network access. Typically, you log in to the SSP machine as user `ssp`, and enter the `netcon`(1M) command in an SSP window. For example:

```
ssp% domain_switch domain_name
ssp% netcon
```

This action changes the window in which you run the `netcon`(1M) command into a `netcon`(1M) window for the domain specified by the `domain_switch`(1M) command. Multiple `netcon`(1M) windows can be opened simultaneously, but only one window at a time can have write privileges to a specific domain. When a `netcon`(1M) window is in read-only mode, you can view messages from the `netcon`(1M) window, but you cannot enter any commands.

You can specify the `netcon`(1M) `-g` option for Unlocked Write permission, `-l` for Locked Write permission, `-f` to force Exclusive Session mode, or `-r` for read-only mode. See TABLE 4-2 for a description of these configuration options. Also refer to the `netcon`(1M) man page for an explanation of how `netcon`(1M) behaves if you do not specify any of these arguments.

If you have write permission, you can enter commands. In addition, you can enter special commands with the tilde (~) prefix to perform the functions offered by the `netcontool`(1M) window.

If `netcon` displays the following message, "`netcon_server is not running for` *domain_name*", the domain may not be up. If it is up, you can run "`netcon_server -r &`" to restart `netcon_server`(1M).

▼ To Start `netcon`(1M) From the Command Line

● **Log in to the SSP as user `ssp` and type:**

```
ssp%  domain_switch domain_name
ssp%  netcon
```

▼ To Start `netcon`(1M) From the CDE Front Panel

1. **From the CDE front panel, select the `SSP` subpanel and then select the `netcon` option.**

2. **Specify the domain name when prompted to do so.**

▼ To Start `netcon`(1M) From the CDE Workspace Menu

1. **From the CDE Workspace menu (right click), select the `SSP` submenu and then select the `netcon` option.**

2. **Specify the domain name when prompted to do so.**

▼ To Exit From a `netcon`(1M) Window

● **Type a tilde (~) followed by a period in the `netcon`(1M) window:**

```
# ~.
```

The `netcon`(1M) session is terminated and the window returns to its previous state.

Note – If you are logging on remotely to the SSP to run netcon(1M), and depending on the terminal emulation package you are using, the escape sequence of the terminal emulator might be the same as that used to exit from a netcon(1M) window. For example, if you enter the tilde period sequence remotely through an rlogin(1) session, the netcon(1M) session is terminated *and* the rlogin(1) window is terminated as well. If you want to avoid this behavior, you can use the sequence ~~., that is, tilde tilde period, to exit from a netcon(1M) window running inside of an rlogin(1) session, without exiting the rlogin(1) session.

For more information about escape sequences, see the netcon(1M) man page.

Using `netcontool`(1M)

The netcontool(1M) GUI program provides the buttons shown in FIGURE 4-1.
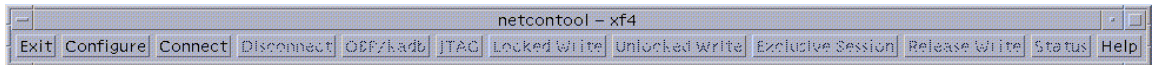
FIGURE 4-1   netcontool GUI Program

TABLE 4-1 explains the netcontool(1M) buttons.

TABLE 4-1   netcontool buttons

Button	Description
Configure	Displays the Console Configuration window. See "To Configure the netcontool(1M) Window" on page 43.
Connect	Displays the netcon(1M) window and initiates the connection process.
Disconnect	Disconnects the console window from the domain and removes the console window. The netcontool(1M) window is still available so that you can reconfigure for another connect session.
OBP/kadb	Breaks to the OpenBoot™ PROM (OBP) or kadb(1M) programs.
JTAG	Toggles the SSP-to-platform connection between a network connection and a JTAG connection.
Locked Write, Unlocked Write, and Exclusive Write	Requests the corresponding mode for the console window. For an explanation of the meaning of these modes, see "To Configure the netcontool(1M) Window" on page 43.

TABLE 4-1   `netcontool` buttons *(Continued)*

Button	Description
Release Write	Releases write access and places the console in read only mode.
Status	Displays information about all open consoles that are connected to the same domain as the current session, as well as the connection type currently used.
Help	Displays information about the `netcontool`(1M) window.
Exit	Exits the program and closes the `netcon`(1M) window if it is still open.

▼ To Display a `netcontool`(1M) Window From the Command Line

● **Log in to the SSP as user** `ssp` **and type:**

```
ssp% domain_switch domain_name
ssp% netcontool &
```

▼ To Display a `netcontool`(1M) Window From the CDE Front Panel

1. **From the CDE front panel, select the** `SSP` **subpanel and then select the** `netcontool` **option.**

2. **Specify the domain name when prompted to do so.**

▼ To Display a `netcontool`(1M) Window From the CDE Workspace Menu

1. **From the CDE Workspace menu (right click), select the** `SSP` **submenu and then select the** `netcontool` **option.**

2. **Specify the domain name when prompted to do so.**

▼ To Display the `netcontool`(1M) Window From Hostview

1. **Select a board from the domain for which you want to display a** `netcontool`**(1M) window by clicking on that board with the left mouse button.**

2. **Select** `Terminal netcontool`.

3. **In the** `netcontool`**(1M) window, click the Connect button.**

   The `netcontool`(1M) window (FIGURE 4-2) is displayed beneath the `netcontool`(1M) buttons.
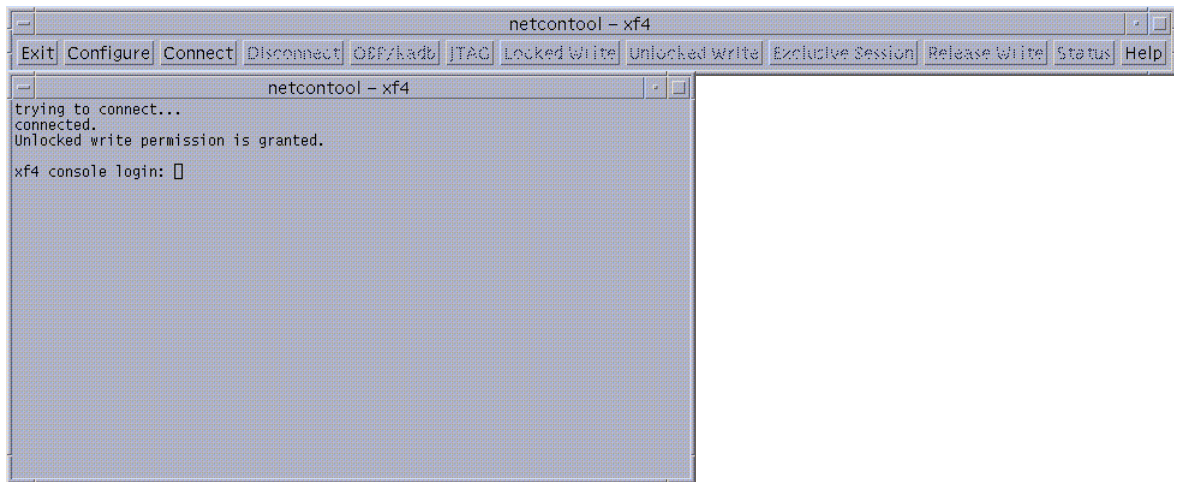
FIGURE 4-2 `netcontool` Window in Hostview

▼ To Configure the `netcontool`(1M) Window

1. **Click the Configure button if you want to configure the** `netcontool`**(1M) window before you display a** `netcon`**(1M) window.**

   The Console Configuration window is displayed (FIGURE 4-3).
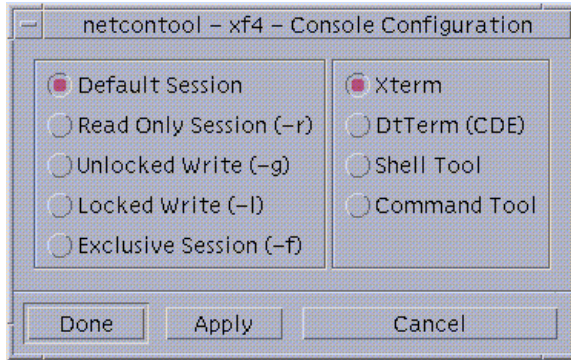
FIGURE 4-3 `netcontool` Console Configuration Window

2. **Select the session type in the left panel, and the terminal emulation type in the right panel.**

3. When you are satisfied with the contents of the window, click Done to accept the settings and dismiss the window, or click Apply to accept the settings without dismissing the window.

The following table contains the options in the Console Configuration window.

TABLE 4-2   Console Configuration Options

Console	Options
Default Session	Causes the default type of session to be started. If no other session is running, the default is unlocked-write mode. If any other session is running, the default is read-only mode.
Read Only Session	Displays a console window where you can view output from a domain, but you cannot enter commands.
Unlocked Write Session	Attempts to display a netcon(1M) window with unlocked write permission. If this attempt succeeds, you can enter commands into the console window, but your write permission is taken away whenever another user requests Unlocked Write, Locked Write, or Exclusive Session permission for the same domain. • If another user currently has Unlocked Write permission, it is changed to read-only permission, and you are granted Unlocked Write permission. • If another user currently has Locked Write permission, you are granted read-only permission. • If another user currently has Exclusive Session permission, you are not allowed to display a netcon(1M) window. • If you are granted Unlocked Write permission and another user requests Unlocked Write or Locked Write permission, you are notified and your permission is changed to read only. You can attempt to re-establish Unlocked Write permission at any time, subject to the same constraints as your initial attempt to gain Unlocked Write permission.

TABLE 4-2   Console Configuration Options *(Continued)*

Console	Options
Locked Write	Attempts to display a console window with Locked Write permission.
	• If you are granted Locked Write permission, no other user can remove your write permission unless that user requests Exclusive Session permission.
	• If another user currently has Locked Write permission, you are granted read-only permission.
	• If another user currently has Exclusive Session permission, you are not allowed to display a netcon(1M) window.
Exclusive Session	Displays a console window with Locked Write permission, terminates all other open console sessions for this domain, and prevents new console sessions for this domain from being started. You can change back to multiple session mode by clicking the Release Write button to release write access, or by clicking the Disconnect button to terminate your console session for the domain. You can also simply quit from the console window (using the Control menu of the window). You are not granted Exclusive Session permission if any other user currently has Exclusive Session permission.
Terminal Emulation Type	The netcon(1M) window is brought up in the specified type of window, otherwise it is grayed out. The xterm(1), dtterm(1), shelltool(1), or cmdtool(1) terminal emulator are available.

netcon(1M) Communications

netcon(1M) uses two distinct paths for communicating console input/output between the SSP and a domain: the standard network interface and the CBE interface. Usually, when the domain is up and running, console traffic flows over the network. If the local network becomes inoperable, the communication mode of the netcon(1M) session automatically switches to the Joint Test Action Group (JTAG) protocol through the CBS interface. You can switch to JTAG mode even when the network is inoperable. To perform this switch, use the ~= command in the netcon(1M) window.

`netcon`(1M) Message Logging

Certain messages sent from the kernel are not displayed in the domain `syslog` messages file, such as OpenBoot™ messages, panic messages, and some console messages. `syslogd` on the domain (1M) must run on the domain to log the messages; this is not possible when a panic occurs, nor is it possible at certain times during the boot sequence. Moreover, panic dumps often fail, so these types of messages may not even appear in a dump file to help you determine the cause of the failure.

However, you can capture all output displayed on an active `netcon`(1M) console through the LOCAL1 facility of `syslog`(1M). This functionality is enabled by default through the `/etc/syslog.conf` file. By default, `netcon`(1M) session output is recorded in the `$SSPLOGGER`/*domain_name*/`netcon` file.

Power Administration

This chapter describes how to control the system power resources from within Hostview or from the command line, control the peripherals power resources from the command line, monitor the power levels in Hostview, and recover from power failure.

▼ To Power Components On or Off From Within Hostview

Note – If you are powering off a board to replace it, use the `power`(1M) command. Do not use the breakers to power off the board; this can cause an arbstop.

1. **Click the left mouse button to select a board in the main Hostview window.**

2. **Choose Power from the Control menu.**

   The Power Control and Status window is displayed (FIGURE 5-1).
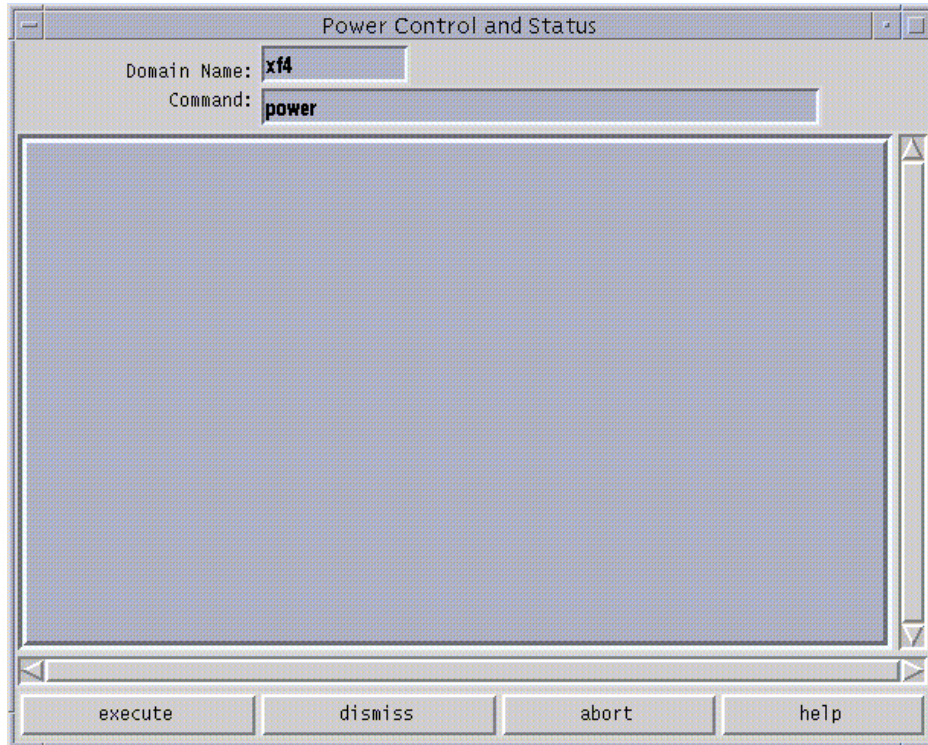
FIGURE 5-1   Power Control and Status Window

The default power(1M) command is displayed in the Command field.

3. Type any power **(1M) command options.**

4. Click the execute button (or press Return) to run the command.
The results are shown in the main panel of the window.

5. For information about the power**(1M) command, click the Help button.**
A help window is displayed. See "Help Window" on page 15.

▼ To Power System Boards On and Off From the Command Line

- **To power on system boards, type:**

```
ssp% power -on -sb board_list
```

where *board_list* is a list of system boards separated by spaces, such as
3 5 6.

Note – If you are powering off a board to replace it, use the power(1M) command.
Do not use the breakers to power off the board; this can cause an arbstop.

After powering on the necessary components, you can run the bringup(1M)
commands on the SSP for the domains you want to boot. See "To Bring up a Domain
From Within Hostview" on page 30.

- **To power off system boards, type:**

```
ssp% power -off -sb board_list
```

where *board_list* is a list of system boards separated by spaces, such as
3 5 6.

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

If you try to power off the system while any domain is actively running the
operating system, the command fails and a message is displayed in the message
panel of the window. In this case, you have two choices. You can force a power off
by executing the power(1M) command again with the -f (force) option. Or, you can
issue a shutdown(1M) or a similar command on the domain for the active domain(s)
to gracefully shut down the processors, and then reissue the command to power off.
Using shutdown(1M) on the domain ensures that all resources are de-allocated and
users have time to log out before the power is turned off. To use shutdown(1M), you
must be logged in to the domain as superuser.

If the platform loses power due to a power outage, Hostview displays the last state
of each domain before power was lost.

▼ To Monitor Power Levels in Hostview

1. **Click the Power button.**

FIGURE 5-2   Power Button

The Power Status Display window is displayed (FIGURE 5-3).
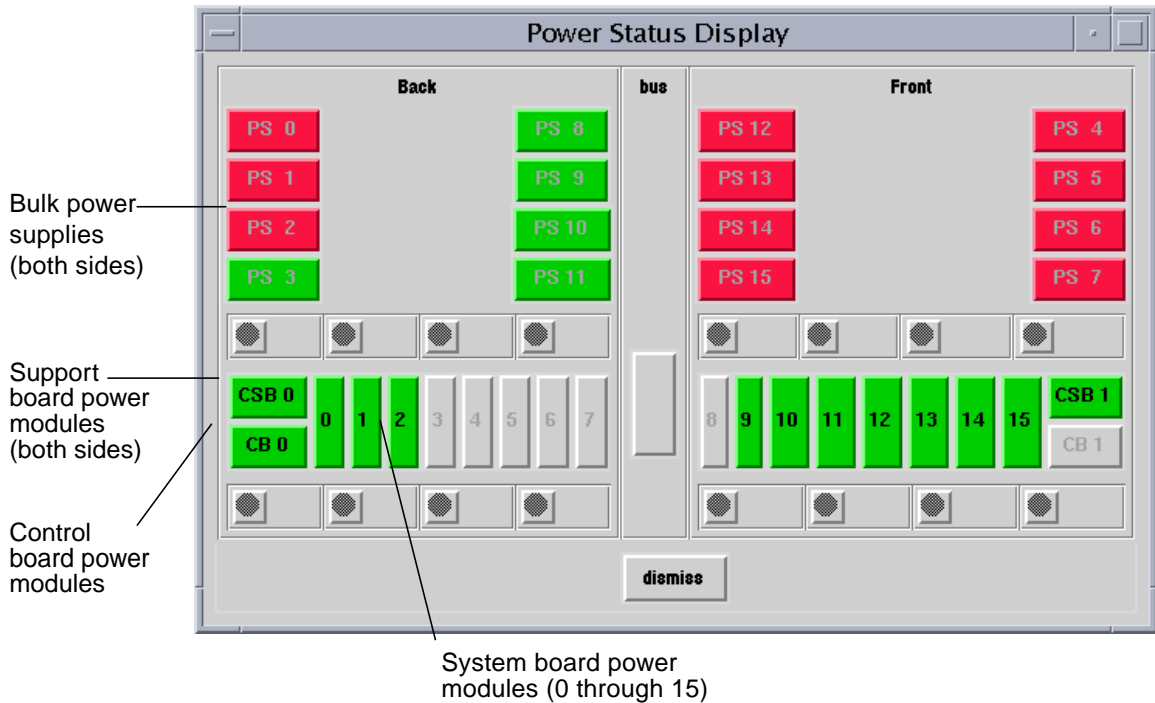
FIGURE 5-3   Power Status Display

In FIGURE 5-3, the bulk power supplies are named PS0 through PS15. If you do not have the dual grid power option for the Sun Enterprise 10000 system, you will see 8 power supplies instead of 16 (PS0 through PS7). The system board power modules are numbered 0 through 15. The support board power modules are named CSB0 and CSB1. The control board power modules are named CB0 and CB1.

Power modules can be colored green, red, or gray. A green power module is functioning properly. A red power module has failed. A gray power module is not present.

2. **Click on a system board.**

   The Power Detail window for that board is displayed (FIGURE 5-4).
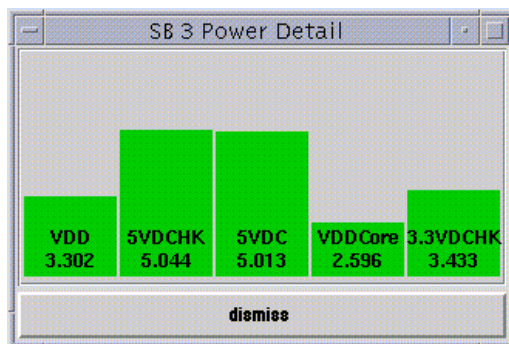
FIGURE 5-4   System Board Power Detail Window

The Power Detail window shows the voltage for each of the five power modules on the board. The power levels are indicated in volts. The bars give a visual representation of the relative voltage levels so that you can monitor them more easily. If a bar is green, the voltage level is within the acceptable range. If a bar is red, the voltage level is either too low or too high. (Thus, a red bar can be short or tall.) The bars never grow taller than the height of the window, so voltage levels that exceed the maximum threshold are displayed as red maximum-height bars. Similarly, bars never shrink below a minimum height, so voltage levels below the minimum threshold are displayed as red minimum-height bars.

The only difference between the detail for a system board and the detail for a control board or support board is the number of power modules.

▼ To Recover From Power Failure

Note – If you lose power only on the SSP, switch on the power to the SSP. The Sun Enterprise 10000 domains are not affected by the loss of power. If you lose power to both the SSP and the Sun Enterprise 10000 system, use the following procedure to recover from the power failure.

1. **Manually switch off the bulk power supplies on the Sun Enterprise 10000 system and the power switch on the SSP.**

   This prevents power surge problems that can occur when power is restored.

2. **When the power is restored, manually switch on the bulk power supplies on the Sun Enterprise 10000 system.**

3. **Manually switch on the SSP power.**

   This boots the SSP and starts the SSP daemons. Check your SSP platform message file for completion of the SSP daemons.

4. **Wait for the recovery process to complete.**

   Any domain that was powered on and running the Solaris operating environment returns to the operating environment run state. Domains at OBP eventually return to an OBP run state.

   The recovery process must finish before any SSP operation is performed.You can monitor the domain message files to determine when the recovery process has completed.

Thermal Conditions Administration

This chapter describes how to administer the thermal conditions and fans from within Hostview and how to monitor and control the fans from within Hostview.

▼ To Monitor Thermal Conditions From Within Hostview

You can use Hostview to monitor thermal conditions for power supplies, processors, ASICs, and other sensors located on system boards, support boards, controller boards, and the centerplane.

1. **Click the Temperature button.**
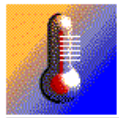
FIGURE 6-1   Temperature Button

The Thermal Status Display window is displayed (FIGURE 6-2). If you do not have the dual grid power option for the Sun Enterprise 10000 system, you will see 8 power supplies instead of 16.
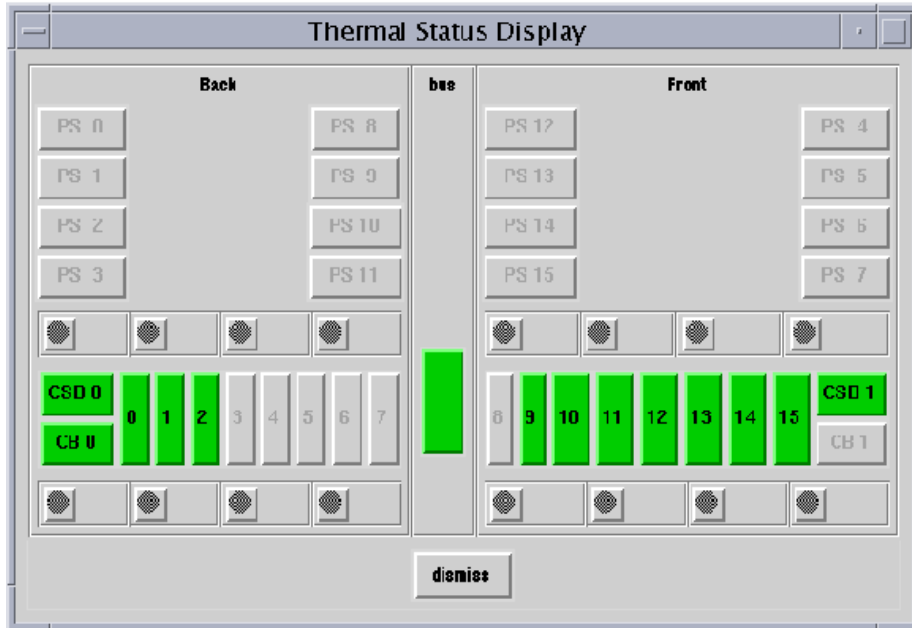
FIGURE 6-2   Thermal Status Display

The centerplane, support boards, control boards, and system boards are shown in green if their temperatures are in the normal range, and in red otherwise.

2. Click a component with the left mouse button to see the thermal details about that component.

The Thermal Detail window for that component is displayed (FIGURE 6-3).
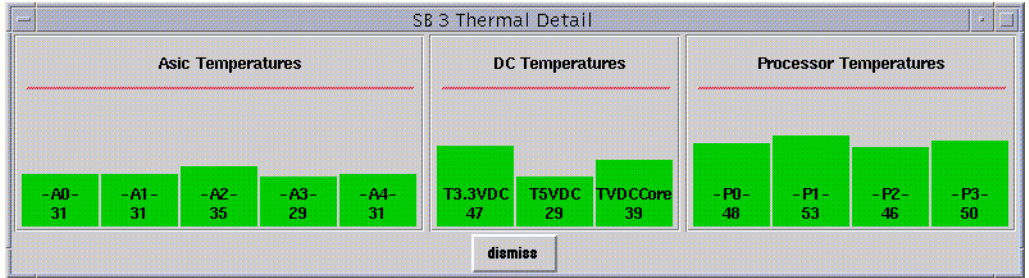
FIGURE 6-3 System Board Thermal Detail

The left panel of the system board detail shows the temperatures for the five ASICs, named A0 through A4. The middle panel shows the temperatures for the three power supplies. The right panel shows the temperatures for the four processors, named P0 through P3.

The temperatures are displayed in degrees centigrade, and the values are shown numerically and as vertical bars. The vertical bars are colored green if the temperature is within the normal range, and red otherwise. The bars never grow taller than the height of the window, so temperature levels above the maximum threshold are displayed as red maximum-height bars. Similarly, bars never shrink below a minimum height, so temperature levels below the minimum threshold are displayed as red minimum-height bars.

The detail windows for control boards, support boards, and the center plane are similar.

▼ To Monitor Fans From Within Hostview

You can use Hostview to monitor fan speeds and fan failures for the 32 fans located throughout the Sun Enterprise 10000 platform.

1. Click the Fan button.
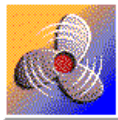
FIGURE 6-4 Fan Button

The Fan Status Display window is displayed (FIGURE 6-5). If you do not have the Sun Enterprise 10000 dual grid power option, you will see 8 power supplies instead of 16.
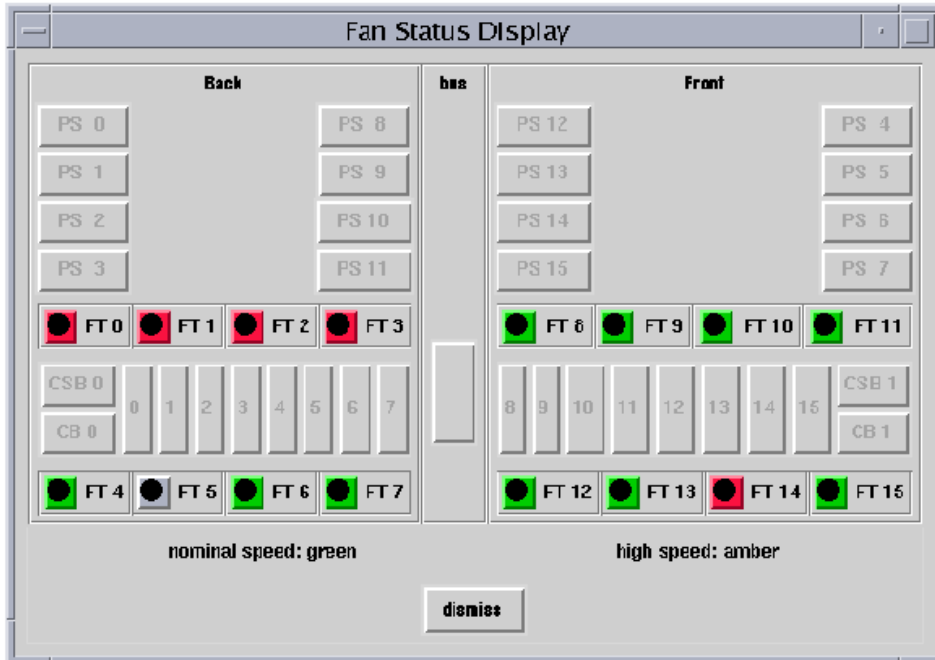
FIGURE 6-5   Fan Status Display

The fan trays are named FT0 through FT7 on the back, and FT8 through FT15 on the front. Each fan tray contains two fans. The color of the fan tray symbol is green if both fans in the tray are functioning at normal speed, amber if both fans are functioning at high speed, and red if either fan within the fan tray has failed.

2. Click a fan tray symbol with the left mouse button to see a detail window about that fan.

The Fan Tray window is displayed (FIGURE 6-6).
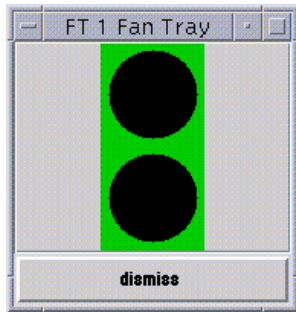
FIGURE 6-6   Fan Tray Window

The top circle indicates the inner fan when you open the fan tray, and the lower circle indicates the outer fan. The color surrounding each circle in the fan detail indicates the status of that fan. The colors are green for normal operation at normal speed, amber for normal operation at high speed, and red for failure.

Blacklist Administration

The blacklisting feature enables you to configure the following components out of the system:

■ System boards
■ Processors
■ Address buses
■ Data buses
■ Data routers
■ I/O controllers
■ I/O adapter card
■ System board memory
■ Memory DIMM groups
■ Sun Enterprise 10000 half-centerplane
■ Port controller ASICs
■ Data buffer ASICs
■ Coherent interface controller ASICs
■ 72-bit half of 144-bit, local, data router within system boards

Generally, you may want to blacklist a component if you believe that component is having intermittent problems, or if it is failing sometime after the system is booted.

If a component has a problem that shows up in the power-on self-test (POST) run by `hpost`(1M) (which is run by the `bringup`(1M) command), that component is automatically configured out of the system by `hpost`(1M). However, that component is not blacklisted. `hpost`(1M) is run on the components in the system before a domain is booted, and on the components on a given board before that board is attached with dynamic reconfiguration (DR). See the *Sun Enterprise 10000 Dynamic Reconfiguration User Guide*.

To blacklist a component, you can edit the `blacklist`(4) file with a text editor, or use Hostview. Hostview does not allow you to blacklist all possible components, so there may be times when you need to edit `blacklist`(4) directly. When a domain

runs POST, hpost(1M) reads the blacklist(4) file and automatically excludes the components specified in that file. Thus, changes that you make to the blacklist(4) file do not take effect until the domain is rebooted.

The file is $SSPVAR/etc/*platform_name*/blacklist, where *platform_name* is the name of the platform. See the blacklist(4) man page for information about the contents of the blacklist(4) file.

▼ To Blacklist Components From Within Hostview

1. **Choose Blacklist File from the Edit menu.**

   The Blacklist Edit window is displayed (FIGURE 7-1).
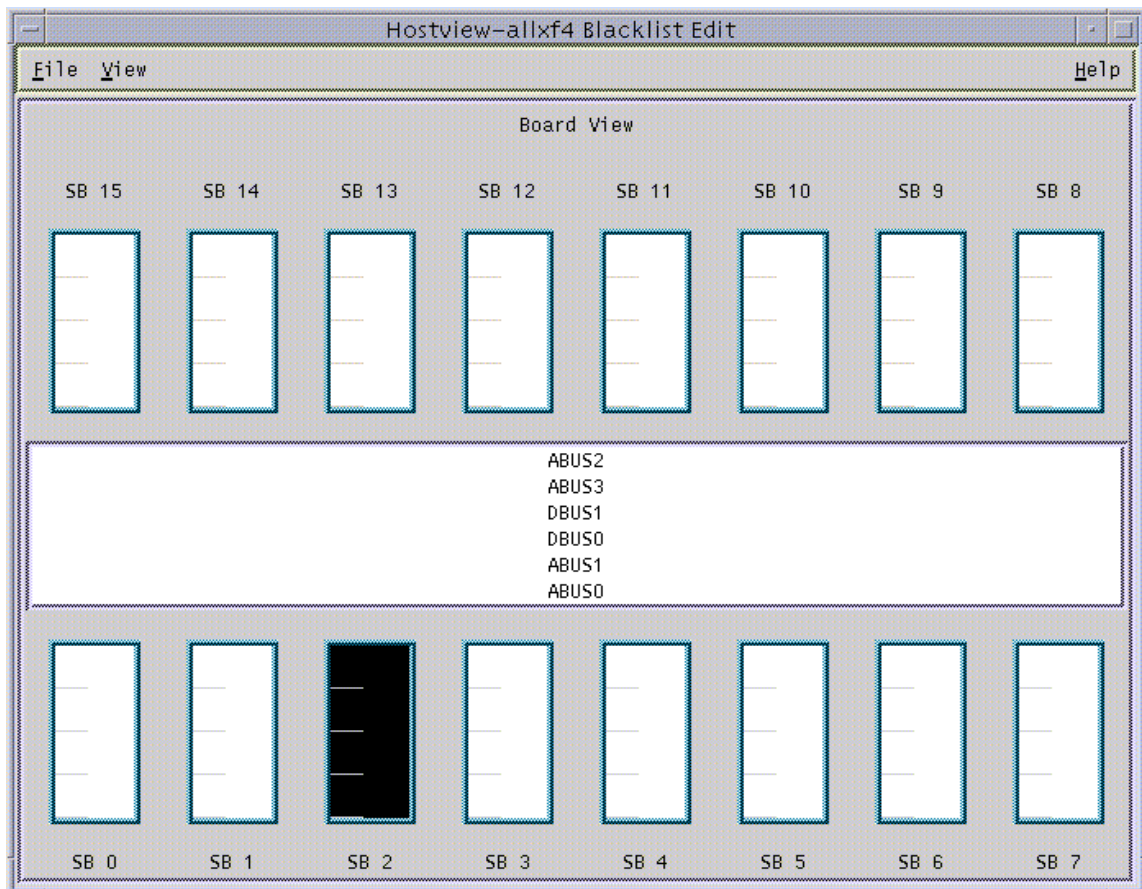
FIGURE 7-1   Blacklist Edit Window—Board View

2. **Select the boards and/or buses that you want to place onto the blacklist.**

   To select a single component and deselect all other components of that type (for example, to select a single board and deselect all other boards), click on that component with the left mouse button. To toggle the selection status of a single component without affecting the selection status of any other component, click that component with the middle mouse button. The selected components are displayed in black.

3. **To save the changes, choose Save from the File menu.**

4. **To exit the Blacklist Edit window, choose Close from the File menu.**

   If you have unsaved changes and you close the Blacklist Edit window by choosing Close from the File menu, you are prompted to save the changes.

▼ To Blacklist Processors From Within Hostview

1. **Choose Blacklist File from the Edit menu.**

   The Blacklist Edit window is displayed.

2. **From the Blacklist Edit window, choose Processors from the View menu.**

   The Blacklist Edit window displays the Processor View window (FIGURE 7-2).
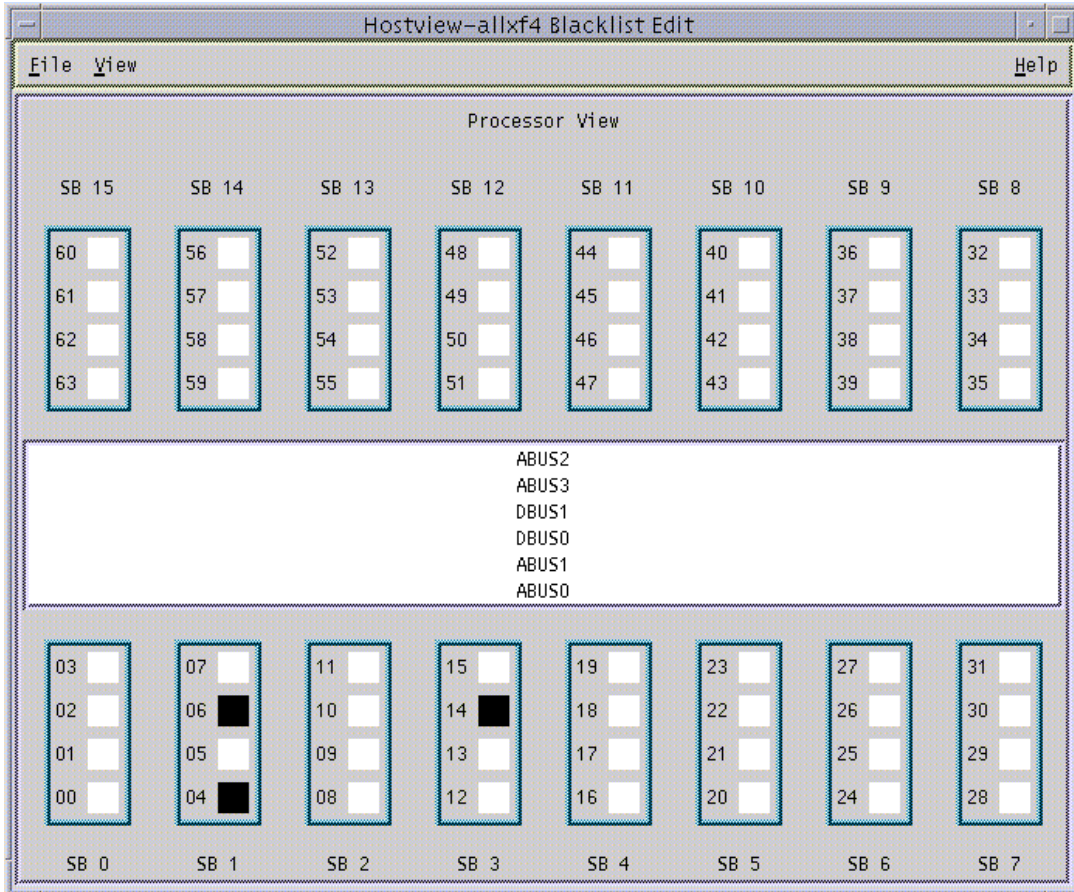
FIGURE 7-2 Blacklist Edit Window—Processor View

3. Select the processors that you want to add to the blacklist.

To select a single processor on a board and deselect all other processors on that board, click that processor with the left mouse button. To toggle the selection status of a processor on a board without affecting the selection status of any other processors on that board, click that processor with the middle mouse button. The selected processors are displayed in black.

4. To save the changes, choose Save from the File menu.

5. To exit the Blacklist Edit window, choose Close from the File menu.

If you have unsaved changes and you close the Blacklist Edit window by choosing Close from the File menu, you are prompted to save the changes.

▼ To Clear the Blacklist File From Within Hostview

1. **In Hostview, choose Blacklist File from the Edit menu.**

2. **From the Blacklist Edit window, choose New from the File menu.**

3. **From the Blacklist Edit window, choose Close from the File menu.**

SSP Failover

SSP provides an automatic failover capability that switches the main SSP to the spare within several minutes of detecting a failover condition, without operator intervention. A failover condition is a point of failure that occurs between the main and spare SSP, their control boards, or their network connections. The automatic failover mechanism continuously monitors both SSPs and their related components to detect a failover condition.

This chapter explains

- Required main and spare SSP architecture
- How to maintain a dual SSP configuration for failover purposes
- How to maintain a single SSP configuration
- How automatic failover works

Note – You can have SSP failover, control board failover, or both. For information on automatic failover for control boards, see Chapter 9. For details on how the SSP, control board, and hub components must be configured for the various types of failover (SSP failover, control board failover, or both), refer to the *Sun Enterprise 10000 SSP 3.5 Installation Guide and Release Notes*.

Required Main and Spare SSP Architecture

For automatic SSP and control board failover to function properly, you must set up your dual SSP configuration as illustrated in the following figure.
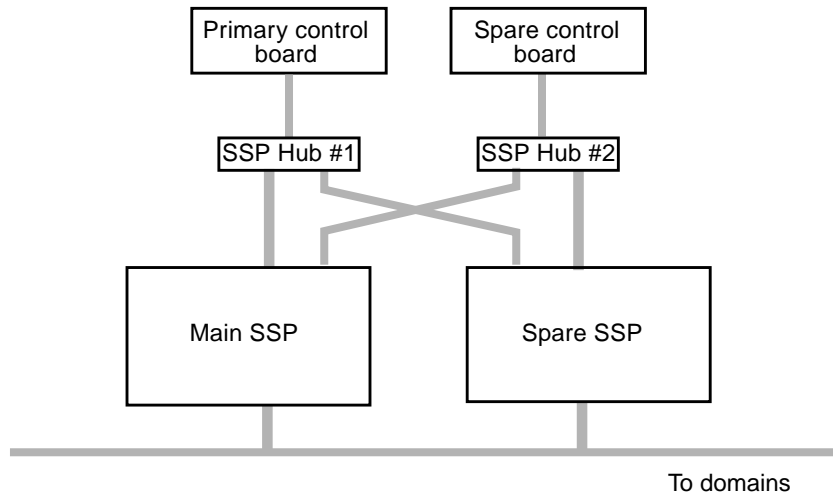
FIGURE 8-1 Dual SSP Configuration Required for Automatic Failover

FIGURE 8-1 shows the SSP, control board, and hub configuration required for dual SSP and control board failover (two SSPs, two hubs, and two control boards). Refer to the *Sun Enterprise 10000 SSP 3.5 Installation Guide and Release Notes* for details on the other configurations (for example, you can have a single SSP configuration with two control boards) supported by the failover feature and the prerequisites for implementing automatic failover.

Maintaining a Dual SSP Configuration

To maintain a dual SSP configuration for failover purposes, note the following:

- The spare SSP must be properly configured to function in the same way as the main SSP within the network.
- The main and the spare SSP must run the same version of the SSP software.
- You can run certain types of third-party applications on your SSPs, provided that your SSPs meet the OpenSSP requirements described in the *SSP 3.5 Installation Guide and Release Notes.*
- For automatic failover:
    - If you have user-created files on the main SSP that need to be maintained on the spare SSP for failover purposes, you must identify those files in the data propagation list.

This data propagation list determines which user-created files on the main SSP are to be automatically copied to and updated on the spare SSP, as part of the data synchronization process. For details on controlling this list, see "Managing Data Synchronization" on page 76.

■ If you have user-created commands that run on the main SSP, you must prepare those user commands for failover recovery, as explained in "Performing Command Synchronization" on page 81.

If you have user commands that require specific files for processing, be sure to add those files to the data propagation list.

■ Any changes that you make to the main SSP must be made to the spare SSP as well.

If failover is disabled or a failover occurs, and you change the SSP configuration, you must *immediately* run ssp_backup(1M) on the main SSP to create an SSP backup file. To successfully switchover to the spare SSP if the main SSP crashes, you must have a backup file that can be restored on the spare SSP.

Maintaining a Single SSP Configuration

In single and dual SSP configurations, the SSP configuration files are copied to the /tmp directory for data synchronization purposes. (For information on data synchronization, see "Managing Data Synchronization" on page 76.) However, for single SSP configurations it is suggested that you run the setdatasync clean command on a regular basis to reduce the number of SSP message and log files that accumulate in the /tmp directory. For additional details on using the setdatasync clean(1M) command, see "To Remove the Data Propagation List" on page 78 and the setdatasync(1M) man page.

How Automatic Failover Works

Automatic failover of the main to the spare SSP is accomplished through the following:

■ Failover monitoring

Failover monitoring is performed by the fod daemon, which continuously monitors the components in a dual SSP configuration for failure conditions. When a failover condition is detected, the fod daemon, in conjunction with the ssp_startup daemon, actually initiates the failover from the main SSP to the spare.

For details on the `fod` daemon and the various failure conditions that it detects, see Chapter 10.

■ Data synchronization

For failover purposes, data on the main SSP must be synchronized with data on the spare SSP. The data synchronization daemon, `datasyncd`(1M),ensures that all SSP configuration files and specified user-created files (identified in the data propagation list) are copied from the main SSP to the spare, so that both SSPs are synchronized when a failover occurs. For further information on the `datasyncd` daemon, see Chapter 10.

This data synchronization occurs whenever the SSP configuration or user-created files change on the main SSP, failover is enabled initially, or a data synchronization backup occurs. For details on data synchronization backup, see "To Synchronize SSP Configuration Files Between the Main and the Spare SSP" on page 78.

■ When a change to an SSP configuration file occurs, the change is propagated immediately to the spare SSP, except for the `ssp_resource`(4) file and the COD license file, which are checked once every minute and then propagated if they have changed.

■ Any change to a user-created file is propagated to the spare SSP at the time interval designated through the `setdatasync`(1M) command.

You control the data synchronization process using the `setdatasync`(1M) command, as described in "Managing Data Synchronization" on page 76.

■ Command synchronization

The recovery of user-defined commands interrupted by an automatic failover is called command synchronization. You use synchronization commands to indicate how these user commands are to be rerun on the new main SSP after a failover. For details on controlling command synchronization, see "Performing Command Synchronization" on page 81.

■ Floating IP address

The working main SSP is identified by a floating IP address that you assign during SSP installation or upgrade. This floating IP address is a logical interface that eliminates the need for a specific SSP host name to communicate between the Sun Enterprise 10000 domains and the main SSP. When a failover occurs, the floating IP address identifies the new main SSP. The floating IP address enables communication between the external monitoring software and the working main SSP.

The following sections provide an overview of the basic SSP failover situations and the various ways to control automatic failover.

SSP Failover Situations

An automatic failover is triggered when a failure in the dual SSP configuration affects the proper operation of the main SSP. Failure points can be caused by the following:

- Failed network connections
- SSP system failure due to a
  - System panic
  - Complete power failure
  - Drop in the OpenBoot PROM (OBP) that persists for five minutes or less
- Resource depletion

  Resource depletion refers to the insufficient amount of disk space and virtual memory needed to perform SSP operations. If these resources drop below a certain threshold, the `fod` daemon initiates a failover. These resources are stored in the `ssp_resource`(4) file and can be modified using the `setfailover` command. For details, see "To Modify the Memory or Disk Space Threshold in the ssp_resource File" on page 74.

However, note that failover will not occur when it has been disabled by operator request or when certain failure conditions prevent the failover. The various failure conditions and the resulting failover actions are summarized in Chapter 10, which identifies and explains the different points of failure detected by the failover process.

SSP Failover State Changes

After a failover occurs, you can obtain failover status information by running the `showfailover`(1M) command on the working SSP. For details, see "Obtaining Failover Status Information". Note that the failover status information displayed reflects the failover state at the time you run the `showfailover` command.

The following state changes occur after an SSP failover:

- The initial failover state is `Failed`, which indicates that a failover occurred.
- The failover state changes to `Disabled` when the working SSP recognizes that the other SSP or its connections are no longer functioning. As a result, the failover feature is disabled.

  If you run `showfailover` at this point and review the output, you will probably find that the states for the various connection links are listed as `FAILED`, indicating that the connections are not working properly.

- When the disabled SSP and its connections are restored, the failover state returns to `Failed`.

The failover feature is not working, even though both SSPs and their connections are working properly. If you run showfailover again and review the output, you will probably find that the states for all connection links are described as GOOD, which indicates that the SSPs and their connection links are functioning. At this point, you must re-enable automatic failover, as described in "To Enable SSP Failover".

Controlling Automatic SSP Failover

The SSP failover capability is automatically enabled upon SSP installation or upgrade. You control the failover state through the setfailover(1M) command, which enables you to do the following:

- Disable, enable, or force an SSP failover.
- View or set the memory or disk space thresholds in the ssp_resource file.

For additional information, see the setfailover(1M) man page.

▼ To Disable SSP Failover

1. **As user ssp on the main SSP, type:**

```
ssp% setfailover off
```

SSP failover remains disabled it until you enable it, as explained in the next procedure.

Note – If you reboot both the main and spare SSP, failover is automatically re-enabled.

---

2. **Run the showfailover(1M) command to verify that failover was disabled.**

For details, see "Obtaining Failover Status Information". The failover state should be listed as Disabled.

▼ To Enable SSP Failover

When you use the setfailover(1M) command to enable failover after it has been disabled, the connection states are checked before failover is enabled. All connection links must be functioning properly before failover can be enabled. If any failed connections exist, failover is not enabled.

1. **As user** `ssp` **on the main SSP, type:**

```
ssp% setfailover on
```

SSP failover is enabled if both SSPs and all their connection links are working.

2. **Run the** `showfailover`**(1M) command to verify that failover was enabled.**

   For details on reviewing the failover state and connection status, see "Obtaining Failover Status Information" on page 75.

   Note – Wait several minutes before verifying the failover state. During this time, the `setfailover` command checks the control board connections before activating SSP failover.

▼ To Force a Failover to the Spare SSP

Note – Before forcing an SSP failover, be sure that both the main and spare SSP are synchronized. Use the `showdatasync`(1M) command to review the status of data synchronization between the main and spare SSP. For details, see "Obtaining Data Synchronization Information" on page 80.

---

1. **As user** `ssp` **on the main SSP, type:**

```
ssp% setfailover force
```

The `setfailover` command checks the data synchronization state before forcing a failover. The forced failover will not occur if any of the following conditions exist:

- A data synchronization backup, referred to as an active archive, is currently being performed.
- A file is being propagated from the main SSP to the spare SSP.
- One or more files exist in the data synchronization queue.

You can run the `showdatasync`(1M) command to obtain information on the synchronization state.

2. **Run the** `showfailover`**(1M) command to verify that the forced failover occurred and review the failover state and connection status.**

    For details, see "Obtaining Failover Status Information" on page 75.

3. **Re-enable SSP failover, as explained in "To Enable SSP Failover" on page 72.**

▼ To Modify the Memory or Disk Space Threshold in the `ssp_resource` File

When memory or disk space resources drop below a certain threshold, a failover occurs. However, you can change the threshold for these resources, which are stored in the `ssp_resource`(4) file, by using the `setfailover`(1M) command.

1. **As user** `ssp` **on the main SSP, do one of the following:**

   ■ To change the memory threshold, type:

   ```
   ssp% setfailover -m memory_threshold
   ```

   where *memory_threshold* is the updated virtual memory value in Kbytes.

   ■ To change the disk space threshold, type:

   ```
   ssp% setfailover -d disk_space_threshold
   ```

   where *disk_space_threshold* is the updated disk space value in Kbytes.

2. **Verify the updated threshold value by using the** `setfailover`**(1M) command with only the** `-m` **or** `-d` **option.**

Obtaining Failover Status Information

Use the showfailover(1M) command on the main SSP to display failover status information. The following example shows the failover information displayed.

```
ssp% showfailover
Failover State:
     SSP Failover: Disabled
     CB Failover:  Active
Failover Connection Map:
     Main SSP to Spare SSP thru Main Hub:    FAILED
     Main SSP to Spare SSP thru Spare Hub:   FAILED
     Main SSP to Primary Control Board:      GOOD
     Main SSP to Spare Control Board:        GOOD
     Spare SSP to Main SSP thru Main Hub:    FAILED
     Spare SSP to Main SSP thru Spare Hub:   FAILED
     Spare SSP to Primary Control Board:     FAILED
     Spare SSP to Spare Control Board:       FAILED
SSP/CB Host Information
     Main SSP:                               xf12-ssp
     Spare SSP:                              xf12-ssp2
     Primary Control Board (JTAG source):    xf12-cb1
     Spare Control Board:                    xf12-cb0
     System Clock source:                    xf12-cb1
```

The failover status includes the

■ Failover state

  The failover state is one of the following:

  ■ Active — automatic failover is enabled and functioning normally
  ■ Disabled — automatic failover has been disabled by operator request or by a failure condition that prevents a failover from occurring
  ■ Failed — a failover occurred

    After a failover, the status is listed as Failed until you re-enable failover using the setfailover(1M) command. You must manually re-enable failover, even after you have fixed all connections and they are identified as GOOD in the failover connection map (explained below).

  Be aware that the failover state undergoes several changes after a failover occurs. For details, see "SSP Failover State Changes" on page 71.

■ Failover connection map

  The connection map provides the status of the control board connection links monitored by the failover processes. A connection link is either GOOD, which means the connection is functioning properly, or FAILED, which indicates the connection is not working.

If you have failed connections, use this connection map to help determine the failure condition. For additional details on the failure conditions associated with the various failure points, see "Description of Failover Detection Points" on page 103 in Chapter 10.

■ SSP/CB host information

The host information section identifies the SSPs, control boards, and the control board that manages the JTAG interface and system clock.

You can also obtain information about the role of the current SSP by specifying the showfailover(1M) command with the -r option. The SSP role is either UNKNOWN (SSP role has not been determined), MAIN, or SPARE.

For additional details on the showfailover(1M) command, see the showfailover(1M) man page.

Managing Data Synchronization

The data synchronization process copies any changes to the SSP configuration or specified user files on the main SSP to the spare SSP. As part of this process, the files to be copied are listed in a data synchronization queue so that you can see which files will be copied from the main to the spare SSP. You can use the showdatasync(1M) command to see which files are in the queue.

If you have user-created files (non-SSP files that are not contained in the SSP directories) that must be maintained on the spare SSP for failover purposes, you must identify these files in a data propagation list (/var/opt/SUNWssp/.ssp_private/user_file_list). The datasyncd daemon uses this list to determine which files to copy from the main SSP to the spare.

By default, the data synchronization process checks for any changes to the user-created files on the main SSP every 60 minutes. You can use the setdatasync command to set the interval at which the data propagation list is to be checked for modifications (see "To Add a File to the Data Propagation List" on page 77). The interval starts from the time at which a file is added to the data propagation list. The files in this list are propagated to the spare SSP only when they have changed from the last interval check.

Note – The data synchronization daemon uses the available disk space in the /tmp directory to copy files from the main SSP to the spare. If you have files to be copied that are larger than the /tmp directory, those files cannot be propagated. For example, if the data synchronization backup file (ds_backup.cpio) file gets larger than the available space in /tmp, you must reduce the size of this file before data propagation can occur. For details on reducing the size of the data synchronization backup file, see "To Reduce the Size of the Data Synchronization Backup File" on page 79.

Use the setdatasync(1M) command to do the following:

- Add a file to the data propagation list and indicate how often this file is to be checked for modifications.
- Remove a file from the data propagation list.
- Erase all entries and temporary files in the data propagation list and remove the data propagation list.
- Push a file to the spare SSP without adding the file to the data propagation list.
- Resynchronize the SSP configuration files between the main and the spare SSP.

Note – The files on the spare SSP are not monitored by the datasyncd daemon, which means that if you remove a user-created file on the spare SSP, the user file will not be automatically restored (copied) from the main to the spare SSP. In addition, do not remove SSP configuration files from the spare SSP.

For additional details, see the setdatasync(1M) man page.

▼ To Add a File to the Data Propagation List

- **As user ssp on the main SSP, type:**

```
ssp% setdatasync -i interval schedule filename
```

where *interval* indicates the frequency (number of minutes) that the specified *filename* is to be checked as part of the data synchronization process. The specified file name must contain the absolute path. The files on the data propagation list are copied to the spare SSP only when those files change on the main SSP, and not each time the files are checked.

▼ To Remove a File From the Data Propagation List

● **As user** `ssp` **on the main SSP, type:**

```
ssp% setdatasync cancel filename
```

where *filename* is the file to be removed from the data propagation list. The file name must contain the absolute path.

▼ To Remove the Data Propagation List

The `setdatasync clean` command is useful for managing disk space in single SSP configurations, where the data propagation list can grow quite large and consume unnecessary disk space. It is possible for the `/tmp` directory to become full, which can cause the system to hang. You can run the `setdatasync clean` command as needed, either daily or weekly to prevent the `/tmp` directory from growing too large. Or, you can automate the cleanup by using the `cron`(1M) command with a `crontab`(1M) entry that uses the `setdatasync clean` command.

Note – Do *not* use this option when you have a dual SSP configuration because it can desynchronize data between the main and spare SSP.

● **As user** `ssp` **on the main SSP, type:**

```
ssp% setdatasync clean
```

▼ To Push a File to the Spare SSP

● **As user** `ssp` **on the main SSP, type:**

```
ssp% setdatasync push filename
```

where *filename* is the file to be moved to the spare SSP without adding the file to the data propagation list. The file name must contain the absolute path.

▼ To Synchronize SSP Configuration Files Between the Main and the Spare SSP

Use this procedure to keep data between the main and spare SSP synchronized, for example, after SSP failover has been disabled then re-enabled. If you want to archive an SSP configuration, use the `ssp_backup`(1M) command.

● **As user** `ssp` **on the main SSP, type:**

```
ssp% setdatasync backup
```

A data synchronization backup file (`/tmp/ds_backup.cpio`) of all SSP
configuration data on the main SSP is created and then restored on the spare SSP.
Note that the data synchronization backup differs from a backup created by the
`ssp_backup`(1M) command:

■ The data synchronization backup, while similar to a backup created by the
`ssp_backup` command, does not back up the `/tftpboot` directory.

■ The data synchronization backup does not restore the following files:

　■ `/var/opt/SUNWssp/.ssp_private/machine_server_fifo`

　■ `/var/opt/SUNWssp/adm/messages`

　　This file is propagated to the `/var/opt/SUNWssp/adm/messages.dsbk` file
　　on the spare SSP.

　■ `/var/opt/SUNWssp/adm/messages.dsbk`

　■ `/var/opt/SUNWssp/.ssp_private/user_file_list`

　■ `/var/opt/SUNWssp/.ssp_private/.ds_queue`

The data synchronization backup can fail if the backup file exceeds the available
disk space in the `/tmp` directory. For details on reducing the size of the data
synchronization backup file, see the following procedure.

▼ To Reduce the Size of the Data Synchronization Backup File

1. **As superuser on the main SSP, run** `ssp_backup`(1M) **to create an archive of your
SSP environment.**

2. **Remove the following files to reduce the size of the data synchronization backup
created before you run** `setdatasync backup`**:**

■ `$SSPLOGGER/messages.`*x*

■ `$SSPLOGGER/`*domain*`/Edd-`*recovery_files*

■ `$SSPLOGGER/`*domain*`/messages.`*x*

■ `$SSPLOGGER/`*domain*`/netcon.`*x*

■ `$SSPLOGGER/`*domain*`/post/`*files*

where *x* is the archive number of the file. Because these files are propagated from the
new main SSP to the spare after a failover, you must remove these files on both the
main and spare SSP to prevent regeneration of these files.

Obtaining Data Synchronization Information

Use the showdatasync(1M) command on the main SSP to obtain basic status information about data synchronization. The examples in this section show the different types of information displayed by the showdatasync command. For additional details, see the showdatasync(1M) man page.

The next example shows the file propagation status of the data synchronization process, the file currently propagated (none), and the number of files queued for data propagation (none). In this case, the status ACTIVE ARCHIVE indicates that a data synchronization backup is being performed.

```
ssp% showdatasync
File Propagation Status:  ACTIVE ARCHIVE
Active File:              -
Queued files:            0
```

The following example shows the file propagation status of the data synchronization process, the name of the file currently being propagated, and the number of files queued for data propagation (none). In this case, the status ACTIVE indicates that the data synchronization process is enabled and functioning normally. The data synchronization backup file is the active file currently propagated.

```
ssp% showdatasync
File Propagation Status:  ACTIVE
Active File:              /tmp/ds_backup.cpio
Queued files:            0
```

The next example shows a data propagation list. Note that the INTERVAL indicates the frequency, in minutes, at which the file is to be checked for changes, as part of the data synchronization process.

```
ssp% showdatasync -l
TIME PROPAGATED          INTERVAL      FILE
Mar 23 16:00:00          60            /tmp/t1
Mar 23 17:00:00          120           /tmp/t2
```

The example below shows the files queued for data synchronization:

```
ssp% showdatasync -Q
FILE
/tmp/t1
/tmp/t2
```

Performing Command Synchronization

Command synchronization recovers user-defined commands that are interrupted by a failover and automatically reruns those commands on the new main SSP after a failover. Command synchronization does the following:

- Maintains a command synchronization list on the spare SSP that specifies the commands to be restarted after a failover. Each command is run as user `ssp`.
- After a failover, reruns specified user commands.
- After a failover, resumes processing of specified user scripts from certain marked points (that you identify within each script).

  These user scripts must be structured so that processing can be resumed from a labeled marker point in the script.

If you want user commands to be automatically recovered after a failover, you must prepare these user commands for synchronization as explained in the following sections.

Preparing User Commands for Automatic Restart

The `runcmdsync`(1M) command prepares a user command for automatic restart. `runcmdsync` adds the user command to the command synchronization list, which identifies the commands to be rerun after a failover.

▼ To Prepare a User Command for Restart

- **As user** `ssp` **on the main SSP, type:**

  ```
  ssp% runcmdsync script_name [parameters]
  ```

where:

script_name is the name of the user command to be restarted.

parameters are the options associated with the specified command.

The specified command will be rerun automatically on the new main SSP after a failover.

Preparing User Scripts for Automatic Recovery

If you want to resume processing of a user script from a certain marked point (location) within the script, you must include the following synchronization commands in the user script:

- initcmdsync(1M) creates a command synchronization descriptor that identifies a particular script and its associated data.

  These descriptors are placed in a command synchronization list that determines which user scripts are to be restarted after an automatic failover.
- savecmdsync(1M) specifies a marker point from which the script can be restarted.
- cancelcmdsync(1M) removes the command synchronization descriptor from the command synchronization list.

Each script must contain the initcmdsync and cancelcmdsync commands to initialize the script for synchronization and then remove the command from the command synchronization list respectively. For details on the synchronization commands, see the cmdsync(1M) man page.

Note – These synchronization commands are intended for use by experienced programmers. You can use the runcmdsync(1M) command instead of the synchronization commands described in this section to prepare a script for recovery. However, the runcmdsync(1M) command will prepare the script so that it is rerun from the beginning and not from specified marker points.

The following procedures describe how to use these synchronization commands.

Note – After an SSP failover or in a single SSP configuration, SSP failover is disabled. When failover is disabled, scripts that contain synchronization commands will generate error messages to the platform log file and return non-zero exit codes. These error messages can be ignored.

▼ To Create a Command Synchronization Descriptor

● **In your user script, type the following to create a command synchronization descriptor that identifies your script:**

```
initcmdsync script_name [ parameters]
```

where:

script_name is the name of the script.

parameters are the options associated with the specified script.

The output returned from the initcmdsync command serves as the command synchronization descriptor.

▼ To Specify a Command Synchronization Marker Point

● **In your user script, type the following to mark an execution point from which processing can be resumed:**

```
savecmdsync -M identifier cmdsync_descriptor
```

where:

identifier is a positive integer that marks an execution point from which the script can be restarted.

cmdsync_descriptor is the command synchronization descriptor output by the `initcmdsync` command.

▼ To Remove a Command Synchronization Descriptor

● **In your user script, type the following after the script termination sequence:**

```
cancelcmdsync cmdsync_descriptor
```

where *cmdsync_descriptor* is the command synchronization descriptor output by the `initcmdsync` command. The specified descriptor is removed from the command synchronization list so that the user script is not run on the new main SSP after a failover.

Obtaining Command Synchronization Information

Use the `showcmdsync`(1M) command on the main SSP to review the command synchronization list that identifies the user commands to be restarted on the new main SSP after an automatic failover.

The following is an example command synchronization list output by the `showcmdsync` (1M) command:

```
ssp% showcmdsync
DESCRIPTOR      IDENTIFIER   CMD
        0              -1   c1 c2 a2
```

For further details, see the `showcmdsync`(1M) man page.

Example Script with Synchronization Commands

SSP provides an example user script that shows how the synchronization commands can be used. This script is located in the `/opt/SUNWssp/examples/cmdsync` directory. This directory also contains a README file that explains how the script works.

Recovering After an SSP Failover

After an SSP failover occurs, you must perform certain recovery tasks:

■ Identify the failure point or condition that caused the failover and determine how to correct the failure.

Depending on the failover condition, note that a failover is either initiated or disabled. To identify the failure point, use the `showfailover`(1M) command to review the failover state and connection status. Review the connection map in the `showfailover` output and the summary of the failover detection points in Chapter 10.

You can also review the platform log file to review other error conditions and determine the corrective action needed to reactivate the failed components.

■ After resolving the problem, re-enable SSP failover using the `setfailover`(1M) command (see "To Enable SSP Failover" on page 72).

■ Rerun any SSP commands that were interrupted by a failover, with the exception of the DR commands `addboard`(1M), `deleteboard`(1M), and `moveboard`(1M), which are automatically resumed on the new main SSP.

Dual Control Board Handling

A platform can be configured with dual control boards for redundancy purposes. One of the control boards is identified as the primary control board and the other control board is considered the spare. The switchover from the primary control board to the spare when a failure occurs is called control board failover. This failover is done automatically. If necessary, you can also force a control board failover.

This chapter explains how control boards function in a dual configuration and how control board failover works.

Note – You can have dual control boards in a single SSP configuration, as well as in a dual SSP configuration (main and spare SSP). Control board failover works the same in either a single or dual SSP configuration.

Control Board Executive

The control board executive (CBE) runs on the control board and facilitates communication between the SSP and the platform.

When power is applied, both control boards boot from the main SSP. After the CBE is booted, it waits for the control board server and the `fod` (failover) daemon running on the SSP to establish a connection. The connections between the `fod` daemon and the control board facilitate SSP and control board failover.

A failover task within CBE enables the main and spare SSP to establish connections for monitoring failover conditions. This task listens for and accepts TCP/IP connections from the `fod` daemons running on the main and spare SSP. The failover task also reads and transmits heartbeat messages to the `fod` daemons on both the main and spare SSP.

Primary Control Board

When the control board server running on the SSP connects to the CBE running on a control board, the CBE asserts the control board as the primary control board. The primary control board is responsible for the JTAG interface, which enables control board components to communicate with other Sun Enterprise 10000 system components so that the Sun Enterprise 10000 system can be monitored and configured. The primary control board also provides the system clock, which synchronizes and controls the speed of the centerplane, CPU clock, and system boards.

Control Board Server

After the SSP is booted, the control board server (CBS) is started automatically, as are several other daemons, including the `fod` daemon. The CBS is responsible for all nonfailover communication between the SSP and the primary control board.

The CBS attempts to connect only to the primary control board identified in the control board configuration file.

Note – Do not manually modify the control board configuration file. Use the `ssp_config`(1M) command to change the control board configuration.

The format of the control board configuration file is as follows:

platform_name:*platform_type*:*cb0_hostname*:*status0*:*cb1_hostname*:*status1*

where:

platform_name is the name assigned by the system administrator.

platform_type is `Ultra-Enterprise-10000`.

cb0_hostname is the host name for control board 0, if available.

status0 indicates that control board 0 is the primary control board (`P` indicates primary, and anything else indicates non-primary).

cb1_hostname is the host name for control board 1, if available.

status1 indicates that control board 1 is the primary control board.

For example:

```
xf2:Ultra-Enterprise-10000:xf2-cb0:P:xf2-cb1:
```

This example indicates that there are two control boards in the `xf2` platform. They are `xf2-cb0` and `xf2-cb1`. `xf2-cb0` is specified as the primary. See the `cb_config`(4) man page for more information.

The communication port that is used for communication between the control board server and the control board executive is specified in `/tftpboot/`*xxxxxxxx*`.cb_port` where *xxxxxxxx* is the control board IP address represented in hexadecimal format.

Control Board Executive Image and Port Specification Files

The main SSP is the boot server for the control board. Two files are downloaded by the control board boot PROM during boot time: the image of `CBE` and the port number specification file. These files are located in `/tftpboot` on the SSP and the naming conventions are:

```
/tftpboot/xxxxxxxx for the cbe image
/tftpboot/xxxxxxxx.cb_port for the port number
```

where *xxxxxxxx* is the control board IP address in hex format.

For example, if the IP address of `xf2-cb0` is 129.153.3.19, the files for control board `xf2-cb0` are:

```
/tftpboot/81990313
/tftpboot/81990313.cb_port
```

Automatic Failover to the Spare Control Board

Control board failover is automatically enabled upon SSP installation or upgrade. The `fod` daemon performs failover monitoring of the control boards and other failover components. If the primary control board is not functioning properly, the `fod` daemon will trigger an automatic failover to the spare control board. A control board failure can be caused by

- A clock failure

  When a clock failure occurs, all active domains arbstop simultaneously and a control board failover is automatically triggered. Both the system clock and JTAG interface are automatically moved to the spare control board. When the new control board is started, normal EDD recovery actions reboot the Sun Enterprise 10000 domains.

- A JTAG interface failure

  If the SSP cannot communicate with the JTAG interface, the SSP determines that the control board failed and automatically triggers a control board failover.

- Failure of the Ethernet interface on the control board
- Failure of the control board processor
- Disconnected cable between the control board and the hub
- Failure of the hub connected to the control board
- Disconnected cable between the main SSP and the hub
- Failure of the SSP network interface card (NIC) for the control board network
- User error caused by disabling the NIC for the control board network

Note that under certain failure conditions the `fod` daemon can disable a control board failover. For a detailed description of the failure conditions and a summary of the failover actions performed, see Chapter 10.

A control board failover can be either partial or complete, depending on whether domains are running:

- If domains are active and a control board failure condition is detected, a partial failover occurs.

  In a partial failover, the JTAG interface is moved from the primary control board to the spare. However, the system clock source remains on the failed primary control board. You must complete the control board failover so that both the JTAG interface and system clock source are managed by the same control board. For details, see "To Force a Complete Control Board Failover" on page 90.

■ If no domains are running and a control board failure condition is detected, a complete failover occurs.

In a complete control board failover, both the JTAG interface and the system clock source are moved from the primary control board to the spare.

Managing Control Board Failover

You can enable, disable, or force a control board failover as explained in the following procedures. Use the setfailover(1M) command on the main SSP to manage the failover state. For example, after a control board failover occurs, you must use the setfailover(1M) command to re-enable the control board failover capability.

▼ To Disable Control Board Failover

● **As user** ssp **on the main SSP, type:**

```
ssp% setfailover -t cb off
```

Control board failover remains disabled until you enable it. To determine whether control board failover was disabled, use the showfailover(1M) command to verify the failover state, as explained in "Obtaining Control Board Failover Information" on page 91.

▼ To Enable Control Board Failover

● **As user** ssp **on the main SSP, type:**

```
ssp% setfailover -t cb on
```

Control board failover is activated when all the connection links are functioning properly. If any failed connections exist, control board failover is not enabled. You can use the showfailover(1M) command to verify that control board failover is enabled and review the connection status.

▼ To Force a Complete Control Board Failover

> **Note –** If you want to force a complete control board failover, where both the JTAG connection and the system clock source are moved from the primary control board to the spare, you must shut down any domains that are running and power off, then power on all system boards before you switch control boards. If you do not shut down all the domains, a partial control board failover occurs. The JTAG connection is moved to the spare control board but the system clock source remains on the former primary control board.

1. **If any domains are running, shut down those domains using the standard** shutdown**(1M) command.**

2. **Log in to the main SSP as user** ssp**.**

3. **To ensure that domains do not arbstop, do the following:**

   a. **Stop event detection monitoring.**

   ```
   ssp% edd_cmd -x stop
   ```

   b. **Power off all of the system boards.**

   ```
   ssp% power -off -all
   ```

   c. **Power on all of the system boards.**

   ```
   ssp% power -on -all
   ```

   d. **Start event detection monitoring.**

   ```
   ssp% edd_cmd -x start
   ```

4. **Type the following to force the control board failover:**

   ```
   ssp% setfailover -t cb force
   ```

5. **Issue the** bringup**(1M) command for all domains.**

6. **Re-enable control board failover as described in "To Enable Control Board Failover" on page 89.**

Obtaining Control Board Failover Information

Use the showfailover(1M) command on the main SSP to obtain the failover state of an SSP or control board failover and the status of the private connection links. The names of the SSPs and control boards are also provided, and the control boards responsible for the JTAG interface and system clock are identified. For details on the failover information displayed, see "Obtaining Failover Status Information" on page 75.

The following example shows the information displayed for a control board failover in which the primary control board failed.

```
ssp% showfailover
Failover State:
     SSP Failover: Active
     CB Failover:  Failed
Failover Connection Map:
     Main SSP to Spare SSP thru Main Hub:   GOOD
     Main SSP to Spare SSP thru Spare Hub:  GOOD
     Main SSP to Primary Control Board:     FAILED
     Main SSP to Spare Control Board:       GOOD
     Spare SSP to Main SSP thru Main Hub:   GOOD
     Spare SSP to Main SSP thru Spare Hub:  GOOD
     Spare SSP to Primary Control Board:    FAILED
     Spare SSP to Spare Control Board:      GOOD
SSP/CB Host Information
     Main SSP:                              xf12-ssp
     Spare SSP:                             xf12-ssp2
     Primary Control Board (JTAG source):  xf12-cb1
     Spare Control Board:                   xf12-cb0
     System Clock source:                   xf12-cb1
```

You can also use Hostview to verify the type of control board failover (complete or partial). When you use Hostview to verify a control board, the "J" (JTAG) and "C" (system clock source) characters indicate which control board manages the JTAG interface and system clock.

FIGURE 9-1 shows an example Hostview window after a partial control board failover. One control board handles the JTAG interface, while the other serves as the system clock source.
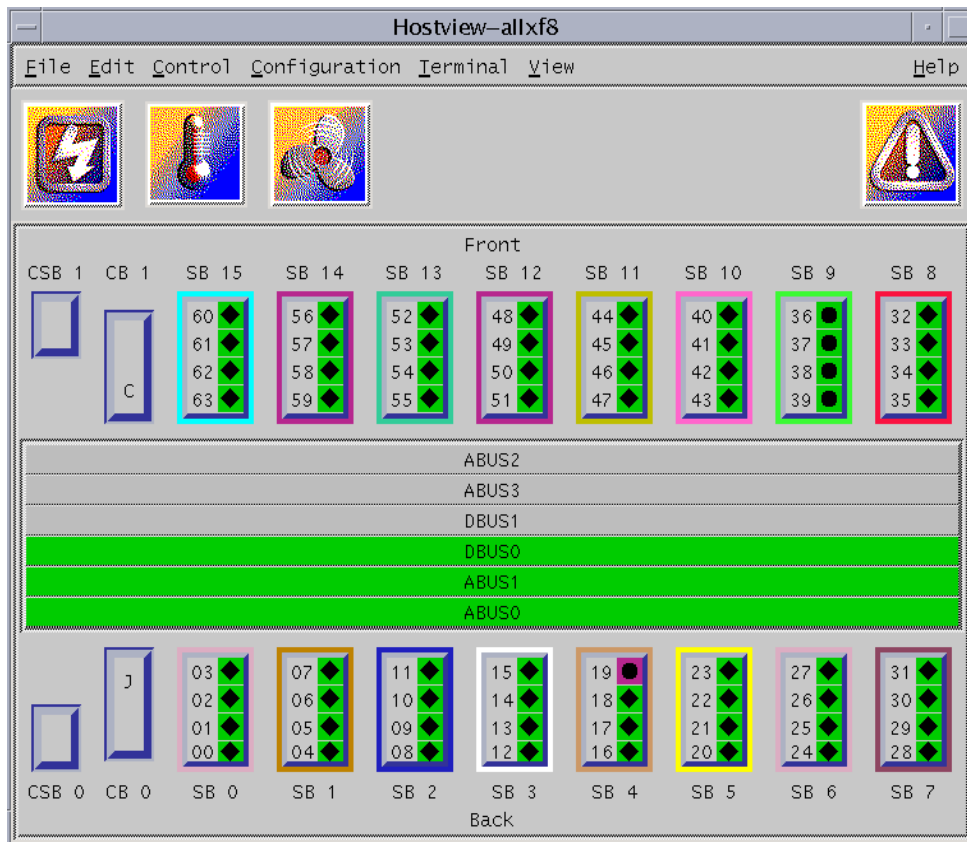
Hostview—allxf8

File   Edit   Control   Configuration   Terminal   View   Help

Front

CSB 1   CB 1   SB 15   SB 14   SB 13   SB 12   SB 11   SB 10   SB 9   SB 8

SB 15: 60 61 62 63
SB 14: 56 57 58 59
SB 13: 52 53 54 55
SB 12: 48 49 50 51
SB 11: 44 45 46 47
SB 10: 40 41 42 43
SB 9: 36 37 38 39
SB 8: 32 33 34 35

C

ABUS2
ABUS3
DBUS1
DBUS0
ABUS1
ABUS0

J

SB 0: 03 02 01 00
SB 1: 07 06 05 04
SB 2: 11 10 09 08
SB 3: 15 14 13 12
SB 4: 19 18 17 16
SB 5: 23 22 21 20
SB 6: 27 26 25 24
SB 7: 31 30 29 28

CSB 0   CB 0   SB 0   SB 1   SB 2   SB 3   SB 4   SB 5   SB 6   SB 7

Back

FIGURE 9-1   Example Hostview Window After a Partial Control Board Failover

After Control Board Failover

After a control board failover occurs, you must perform certain recovery tasks:

- Identify the failure point or condition that caused the failover and determine how to correct the failure.

  For example, if a control board failover occurred due to a faulty control board, you must determine whether you need to replace the failed control board.

  Use the showfailover(1M) command to review the failover state and verify which control board is responsible for the JTAG interface and system clock. Review the connection map in the showfailover output and the summary of the failover detection points in Chapter 10.

You can also review the platform log file to review other error conditions and determine the corrective action needed to reactivate the failed components.

■ If a partial failover occurred, resynchronize the JTAG and system clock interfaces so that they are managed by the same control board.

To resynchronize the JTAG and system clock interfaces, perform a complete control board failover as described in "To Force a Complete Control Board Failover" on page 90. The first domain that is brought up resynchronizes the system clock and the JTAG interface on the primary control board.

■ Once you have resolved the control board failure, re-enable control board failover (for details, see "To Force a Complete Control Board Failover" on page 90).

SSP Internals

SSP operations are generally performed by a set of daemons and commands. This chapter provides an overview of how the SSP works and describes the SSP 3.5 daemons, processes, commands, and system files. For more information about daemons, commands, and system files, refer to the *Sun Enterprise 10000 SSP 3.5 Reference Manual*.

Startup Flow

The events that take place when the SSP boots are as follows:

1. User powers on the SSP (monitor, CPU/disk, and CD-ROM). The SSP boots automatically.

2. During the SSP boot process, the `/etc/rc2.d/s99ssp` startup script is called when the system enters run level 2. This script starts `ssp_startup`, which is responsible for starting other SSP daemons. If any of these SSP daemons die, `ssp_startup` restarts them.

3. `ssp_startup` first initiates the following SSP daemons on both the main and spare SSP: `machine_server`, `fad`, and `fod`. The `fod` daemon determines the role of the SSP by first querying the `fod` daemon on the other SSP. If this query is not successful, `fod` will connect to the control board to determine the SSP role.

   If the SSP is the main, `ssp_startup` also initiates the following daemons: `datasyncd`, `cbs`, `straps`, `snmpd`, `edd`, and if domains are running, `obp_helper` and `netcon_server`. `ssp_startup` also calls `cb_reset` to start control board initialization. The control board server (CBS) connects to the primary control board, which is responsible for the JTAG interface.

   If the SSP is the spare, `ssp_startup` is complete.

   `ssp_startup` monitors the role of the SSP. If a role change is detected, `ssp_startup` initiates an SSP failover. After the failover, `ssp_startup` will configure the spare SSP as the new main SSP and initiate the daemons (listed above) needed for the new main SSP.

4. When you get a message in the platform message file indicating that the startup of the SSP as the main or spare is complete, you can use SSP 3.5 commands such as `domain_create`(1M) or `bringup`(1M).

Sun Enterprise 10000 Client/Server Architecture

The Sun Enterprise 10000 system control board interface is accessed over an Ethernet connection using the TCP/IP protocol. The control board executive, CBE, runs on the control board. The control board server, `cbs`(1M), runs on the SSP and makes service requests. The SSP control board server provides services to SSP clients.

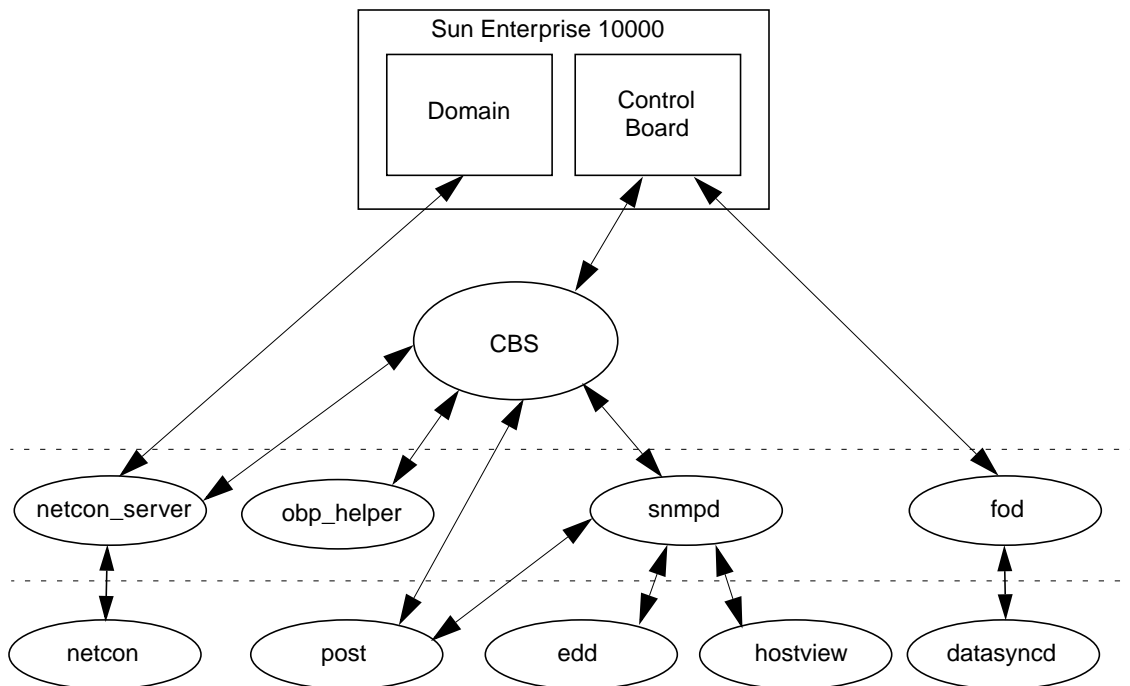FIGURE 10-1 illustrates the Sun Enterprise 10000 system client/server architecture:

FIGURE 10-1 Sun Enterprise 10000 Client/Server Architecture

Note – There is one instance of edd(1M) for the platform supported by the SSP. There is one instance of obp_helper(1M) and netcon_server(1M) for each domain on the platform.

SSP Daemons

The SSP daemons play a central role on the SSP. The following table briefly describes these daemons.

TABLE 10-1   SSP Daemons

Name	Description
cbs	The control board server provides central access to the Sun Enterprise 10000 control board for client programs running on the SSP.
edd	The event detector daemon initiates event monitoring on the control boards. When a monitoring task detects an event, edd(1M) runs a response action script.
fad	The file access daemon provides distributed file access services to SSP clients that need to monitor, read, and write to the SSP configuration files.
fod	The failover daemon monitors SSP components (connections to the SSPs, control boards, and domains) and SSP resources for failure conditions that prevent the proper operation of the main SSP.
datasyncd	The data synchronization daemon propagates SSP configuration data and specified files from the main SSP to the spare. This synchronization keeps SSP data on the spare SSP current with the main SSP for failover purposes.
machine_server	The machine server daemon routes platform and domain messages to the proper messages file. See machine_server(1M).
netcon_server	The netcon server daemon is the connection point for all netcon(1M) clients. netcon_server(1M) is responsible for communication to the domains.
obp_helper	The OpenBoot PROM (OBP) helper daemon runs OpenBoot. obp_helper(1M) is responsible for providing services to OBP, such as NVRAM simulation, IDPROM simulation, and time of day.

TABLE 10-1   SSP Daemons *(Continued)*

Name	Description
snmpd	The SNMP proxy agent listens to a UDP port for incoming requests and also services the group of objects specified in Ultra-Enterprise-10000.mib.
straps	The SNMP trap sink server listens to the SNMP trap port for incoming trap messages and forwards received messages to all connected clients.
xntpd / ntpd	The network time protocol (NTP) daemon provides time synchronization services. ntpd is used in the Solaris 2.6, Solaris 7, and Solaris 8, and Solaris 9 operating environments and replaces the xntpd daemon used in the Solaris 2.5.1 operating environment. For details on ntpd, see the *Sun Enterprise 10000 SSP 3.5 Installation Guide and Release Notes* and the xntpd(1M) man page.

Event Detector Daemon

The event detector daemon, edd(1M), is a key component in providing the reliability, availability, and serviceability (RAS) features of the Sun Enterprise 10000 system. edd(1M) initiates event monitoring on the Sun Enterprise 10000 control board, waits for an event to be generated by the event detection monitoring task running on the control board, and then responds to the event by executing a response action script on the SSP. The conditions that generate events and the response taken to events are fully configurable.

edd(1M) provides the mechanism for event management, but does not handle the event detection monitoring directly. Event detection is handled by an event monitoring task that runs on the control board. edd(1M) configures the event monitoring task by downloading a vector that specifies the event types to be monitored. Event handling is provided by response action scripts, which are invoked on the SSP by the edd(1M) when an event is received.

At SSP startup, edd(1M) obtains many of its initial control parameters from the following:

- $SSPVAR/etc/*platform_name*/edd.erc provides configuration information for the Sun Enterprise 10000 platform.
- $SSPVAR/etc/*platform_name*/*domain_name*/edd.erc provides configuration information for a particular domain. The event response configuration files (edd.erc) specify how the event detector will respond to events.
- $SSPVAR/etc/*platform_name*/edd.emc lists the events that edd(1M) will monitor.

The RAS features are provided by several collaborative programs. The control board within the platform runs a control board executive (CBE) program that communicates through the Ethernet with a control board server daemon, cbs(1M), on the SSP. These two components provide the data link between the platform and the SSP.

The SSP provides a set of interfaces for accessing the control board through the control board server and the simple network management protocol (SNMP) agent. edd(1M) uses the control board server interface to configure the event detection monitoring task on the control board executive (FIGURE 10-2).
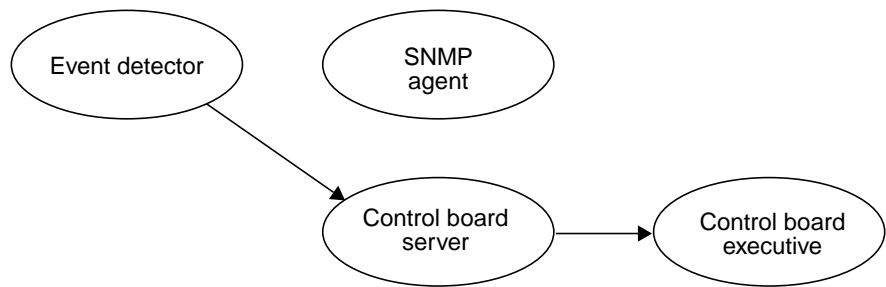
FIGURE 10-2 Uploading Event Detection Scripts

After it is configured, the event detection monitoring task polls various conditions within the platform, including environmental conditions, signature blocks, power supply voltages, performance data, and so forth. If an event detection script detects a change of state that warrants an event, an event message containing the pertinent information is generated and delivered to the control board server, cbs(1M). Upon receipt of the event message, the control board server delivers the event to the SNMP agent, which in turn generates an SNMP trap (FIGURE 10-3).
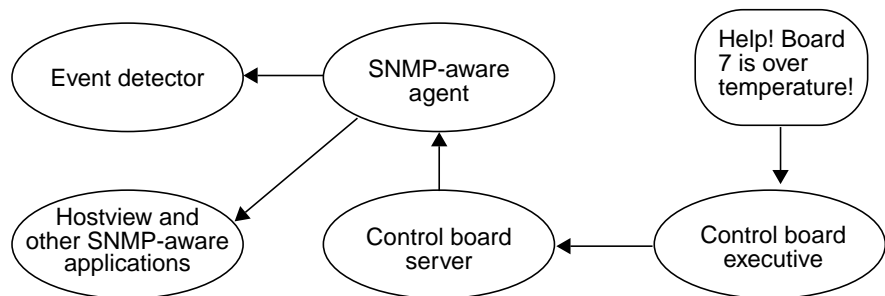
FIGURE 10-3 Event Recognition and Delivery

Upon receipt of an SNMP trap, edd(1M) determines whether to initiate a response action. If a response action is required, edd(1M) runs the appropriate response action script as a subprocess (FIGURE 10-4).
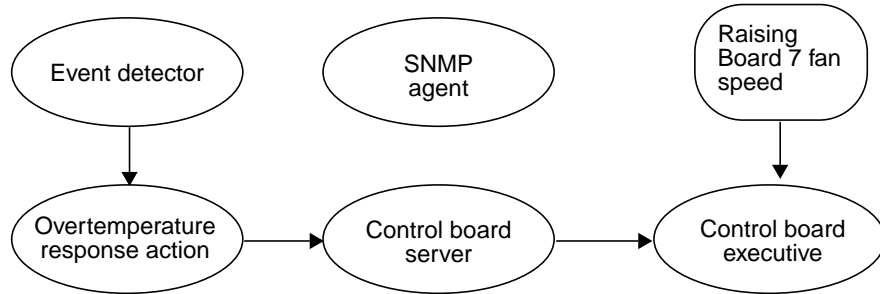

FIGURE 10-4 Response Action

Event messages of the same type or related types can be generated while the response action script is running. Some of these secondary event messages may be meaningless or unnecessary if a responsive action script is already running for a similar event. For example, when edd(1M) runs a response action script for an overtemperature event, additional overtemperature events can be generated by the event monitoring scripts. edd(1M) does not respond to those overtemperature events (generated in response to the same overtemperature condition) until the first response script has finished. It is the responsibility of applications, such as edd(1M), to filter the events they will respond to as necessary. The cycle of event processing is completed at this point.

The edd(1M) response to a domain crash is another example of how edd(1M) responds to an event. After a domain crash, edd(1M) invokes the bringup(1M) script. The bringup(1M) script runs the POST program, which tests Sun Enterprise 10000 components. It then uses the obp_helper(1M) daemon to download and begin execution of OBP in the domain specified by the SUNW_HOSTNAME environment variable. This happens only if a domain fails (for example, after a kernel panic), in which case it is rebooted automatically. After a halt or shutdown, you must manually run bringup(1M), which then causes OBP to be downloaded and run.

Control Board Server

The control board server (CBS) runs on the SSP. Whenever a client program running on the SSP needs to access the Sun Enterprise 10000 system, the communication is funneled through cbs(1M). cbs(1M), in turn, communicates directly with a control

board executive (CBE) running on the primary control board in the Sun Enterprise 10000 system. The primary control board provides the JTAG interface. cbs(1M) converts client requests to the control board management protocol (CBMP) that is understood by CBE. The following diagram shows how the CBS and CBEs are connected.
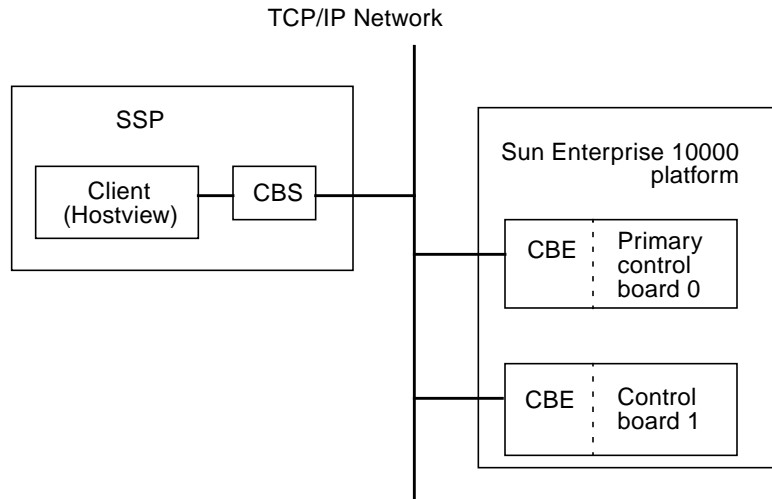
FIGURE 10-5 CBS Communication Between SSP and Sun Enterprise 10000 System

cbs(1M) relies on the cb_config(4) file to determine the platform it will manage, and the control board with which it will interact. Do not directly modify the cb_config(4) file; it is automatically maintained by domain management tools and commands.

File Access Daemon

The file access daemon, fad(1M), is responsible for providing distributed file access services, such as file locking, to all SSP clients that need to monitor, read, and write changes to SSP configuration files. Once a file is locked by a client, other clients are prevented from locking that file until the first client releases the lock.

Failover Daemon

The failover daemon, fod(1M), continuously monitors the following to detect a failure condition that prevents the proper operation of the main SSP:

- Connections between the
  - Main and spare SSP
  - Main and spare SSP with the Sun Enterprise 10000 domains
  - Main and spare SSP with the Sun Enterprise 10000 control boards
- SSP operating resources, such as disk space and memory usage

This `fod` daemon runs on both the main and spare SSP. Depending on the type of failure condition detected, the `fod` daemon either initiates a control board failover, or it works with `ssp_startup` to initiate an SSP failover. The following section identifies the failover detection points and the conditions that initiate or disable a failover.

Failover Detection Points

The following figure illustrates the standard layout of a dual SSP and control board configuration required for automatic failover. The numbers identify points of failure that are detected by the `fod` daemon, and are summarized in TABLE 10-2.
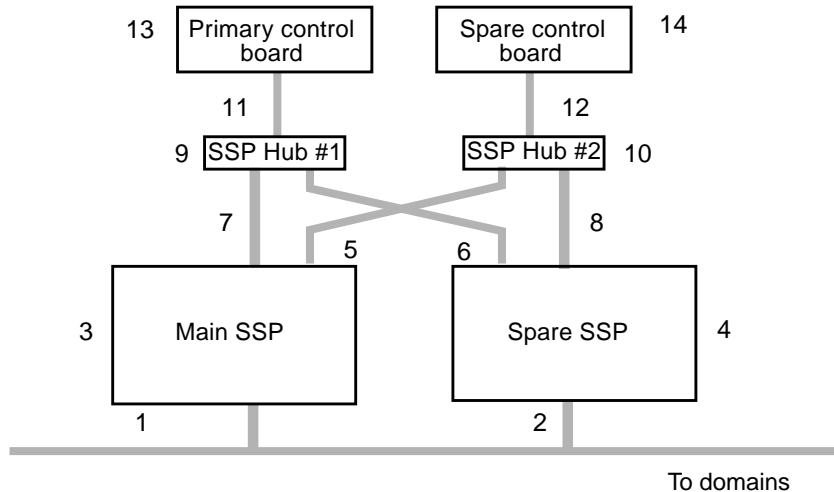
FIGURE 10-6 Automatic Failover Detection Points

The following table summarizes each failure condition and the resulting failover actions. For each failure point, refer to the detailed description of that failure point provided in the next section.

TABLE 10-2   Summary of Failover Detection Points and Actions

Failure Point	SSP Failover	SSP Failover Disabled	Control Board Failover	Control Board Failover Disabled
1 Main SSP to Domains	X			
2 Spare SSP to Domains		X		
3 Main SSP	X			
4 Spare SSP		X		
5 Main SSP to Spare Hub		X		X
6 Spare SSP to Main Hub		X		
7 Main SSP to Main Hub	X			
8 Spare SSP to Spare Hub		X		
9 Main Hub			X	
10 Spare Hub		X		X
11 Primary Control Board to Main Hub			X	
12 Spare Control Board to Spare Hub				X
13 Primary Control Board			X	
14 Spare Control Board				X

Description of Failover Detection Points

This section provides a detailed description of each failover detection point identified in TABLE 10-2:

1. Main SSP to Domains Failure

    The main SSP detects this failure of the public network interface on the main SSP to the domains and initiates an SSP failover.

    The public network interface failure is not fatal to the main SSP, but it affects dynamic reconfiguration (DR), Sun Enterprise Cluster, and Sun Management Center operations. This failure

- Prevents DR operations from communicating with the DR daemons in the active domains
- Restricts `netcon` sessions to the JTAG interface

- Prevents the net booting of the SSP
- Makes the CD-ROM inaccessible
- Prevents the main SSP in a Sun Enterprise Cluster configuration from shutting down cluster nodes in a split-brain situation, which could allow a potential corruption of the cluster database
- Prevents Sun Management Center from querying domains about their current state and configuration

---

Note – The `fod` daemon monitors connections between the SSPs and the Sun Enterprise 10000 domains less frequently than the connections between the SSPs and and the control boards. If the main SSP cannot communicate with the domains, but the spare SSP can communicate with some or all of the domains, this failure condition must persist for 25 minutes before a failover is triggered. After 25 minutes, the `fod` daemon will initiate a failover, provided that the spare SSP can communicate with the primary control board and the spare SSP has sufficient memory and disk space.

---

2. Spare SSP to Domain Failure

   The spare SSP detects this failure of the public network interface on the spare SSP to domains. This public interface failure does not cause a loss in critical SSP functionality, but it can affect dynamic reconfiguration, Sun Remote Services (SRS), Sun Management Center, and the Sun Cluster console.

   As a result, SSP failover is disabled.

3. Main SSP Failure

   A failure in the main SSP can be caused by the following:

   - The depletion of SSP resources, such as virtual memory or disk space. The main SSP detects this failure and initiates a failover.
   - A system crash, which is detected by the spare SSP and the control boards. The spare SSP initiates the failover.

4. Spare SSP Failure

   Both control boards and the main SSP detect this spare SSP failure. This failure disables SSP failover.

5. Main SSP to Spare Hub Failure

   Both SSPs detect this failure of the control board network connection from the main SSP to the spare hub and spare control board. Both SSP and control board failover are disabled.

6. Spare SSP to Main Hub Failure

Both SSPs and the primary control board detect this failure of the control board network connection from the spare SSP to the main hub and primary control board.

SSP failover is disabled because the spare SSP cannot monitor the SSP as required.

7. Main SSP to Main Hub Failure

Both SSPs and the primary control board detect this failure of the control board network connection from the main SSP to the main hub and primary control board. When connectivity from the spare SSP to the primary control board is verified, an SSP failover is attempted. If the SSP failover is unsuccessful, a control board failover occurs instead.

8. Spare SSP to Spare Hub Failure

Both SSPs and the spare control board detect this failure of the control board network connection from the spare SSP to the spare hub, and spare control board. SSP failover is disabled.

9. Main Hub Failure

Both SSPs and the primary control board detect this failure of the main hub and all connections to the primary control board. If connectivity to the domains exists and the domains are running, this failure causes a partial control board failover to the spare control board (JTAG failover only). If no domains are currently running, this failure causes a complete control board failover (JTAG and system clock failover).

If a partial control board failover occurs, note that full control board functionality is retained, even though the JTAG interface and system clock are split between the primary and spare control boards.

10. Spare Hub Failure

Both SSPs and the spare control board detect this failure of the spare hub and all connections to the spare control board.

11. Primary Control Board to Main Hub Failure

Both SSPs and the primary control board detect this failure of the control board network connection from the main hub to the primary control board. If domains are running, this failure causes a partial control board failover (JTAG only) to the spare control board. If no domains are running, this failure causes a full control board failover.

If a partial control board failover occurs, note that full control board functionality is retained, even though the JTAG interface and system clock are split between the primary and spare control boards.

12. Spare Control Board to Spare Hub Failure

Both SSPs and the spare control board detect this failure of the control board network connection from the spare hub to the spare control board. This failure disables the control board failover.

13. Primary Control Board Failure

Both SSPs detect this failure. If domains are running, this failure causes a partial control board failover (JTAG only) to the spare control board. If no domains are running, this failure causes a full control board failover.

If a partial control board failover occurs, note that full control board functionality is retained, even though the JTAG interface and system clock are split between the primary and spare control boards.

14. Spare Control Board Failure

Both SSPs detect this failure, which disables a control board failover.

Data Synchronization Daemon

The data synchronization daemon, `datasyncd`(1M), propagates all SSP configuration information from the main to spare SSP. The `datasyncd` daemon uses a data propagation list that identifies the SSP and non-SSP files to be monitored and propagated. You use the `setdatasync`(1M) command to add non-SSP files to the data propagation list.

The `datasyncd` daemon runs on the main SSP and works with the `fad` daemon to monitor updates to SSP files on the main SSP. The `datasyncd` daemon then copies these updated files to the spare SSP, so that data on both SSPs is synchronized.

OpenBoot PROM

On the domain, OpenBoot PROM (OBP) is not a hardware PROM; it is loaded from a file on the SSP. An SSP file also replaces the traditional OBP NVRAM and `idprom` (`hostid`).

The OBP file is located under a directory path that is specific to the SunOS release. SunOS 5.6 corresponds to the Solaris 2.6 operating environment, SunOS 5.7 corresponds to the Solaris 7 operating environment, SunOS 5.8 corresponds to the Solaris 8 operating environment, and SunOS 5.9 corresponds to the Solaris 9 operating environment. You can determine your SunOS version with `uname -r`. For example, under SunOS 5.7, the OBP file is located in the following directory:

```
/opt/SUNWssp/release/Ultra-Enterprise-10000/5/7/hostobjs/obp
```

where the `/5/7` portion of the path corresponds to the SunOS version number. If your release contains a different version of the operating system, that portion of the path will be different.

The primary task of OBP is to boot and configure the operating system from either a mass storage device or from a network. OBP also provides extensive features for testing hardware and software interactively. As part of the boot procedure, OBP probes all the SBus slots on all the system boards and builds a device tree. This device tree is passed on to the operating system. The device tree is ultimately visible using the command `prtconf` (for more information, see the SunOS `prtconf`(1M) man page).

OBP also interprets and runs FCode on SBus cards, which provides loadable, simple drivers for accomplishing boot. In addition, it provides a kernel debugger, which is always loaded.

The following sections describe how the `obp_helper` daemon and `download_helper` file control the OBP.

`obp_helper` Daemon

`obp_helper`(1M) is responsible for starting processors other than the boot processor. It communicates with OBP through bootbus SRAM(BBSRAM), responding to requests to supply the time-of-day, get or put the contents of the pseudo-EEPROM, and release slave processors when in multiprocessor mode. To release the slave processors, `obp_helper`(1M) must load `download_helper` into the BBSRAM of all the slave processors, place an indication in BBSRAM that it is a slave processor, then start the processor by releasing the bootbus controller reset.

The `bringup`(1M) command starts `obp_helper`(1M) in the background, which kills the previous `obp_helper`(1M), if one exists. `obp_helper`(1M) runs `download_helper` and subsequently downloads and runs OBP.

For more information, see the `obp_helper`(1M), and `bringup`(1M) man pages and "download_helper File" on page 107.

`download_helper` File

`download_helper` enables programs to be downloaded to the memory used by a domain instead of BBSRAM. This provides an environment in which host programs can run without having to know how to relocate themselves to memory. These programs can be larger than BBSRAM.

`download_helper` works by running a protocol through a mailbox in BBSRAM. The protocol has commands for allocating and mapping physical to virtual memory, and for moving data between a buffer in BBSRAM and virtual memory. When

complete, the thread of execution is usually passed to the new program at an entry point provided by the SSP. After this occurs, `download_helper` lives on in BBSRAM so it can provide reset handling services. Normally, you do not need to be concerned with the download helper; it is used only by the `obp_helper`(1M) daemon. See the `obp_helper`(1M) man page for more information.

POST

Power-on self-test (POST) probes and tests the components of uninitialized Sun Enterpirse 10000 system hardware, configures what it deems worthwhile into a coherent initialized system, and hands it off to OpenBoot PROM (OBP). POST passes to OBP a list of only those components that have been successfully tested; those in the `blacklist`(4) file are excluded.

`hpost`(1M) is the SSP-resident executable program that controls and sequences the operations of POST. `hpost`(1M) reads directives in the optional file `.postrc` (see `postrc`(4)) before it begins operation with the host.

⚠ **Caution –** Running hpost(1M) outside of the bringup(1M) command can cause the system to fail. hpost(1M), when run by itself, does not check the state of the platform, and causes fatal resets.

POST looks at `blacklist`(4), which is on the SSP, before preparing the system for booting. `blacklist`(4) specifies the Sun Enterprise 10000 components that POST must not configure.

POST stores the results of its tests in an internal data structure called a *board descriptor array*. The board descriptor array contains status information for most of the major components of the Sun Enterprise 10000 system, including information about the UltraSPARC™ modules.

POST attempts to connect and disconnect each system board, one at a time, to the system centerplane. POST then connects all the system boards that passed the tests to the system centerplane.

Environment Variables

Most of the necessary environment variables are set when the `ssp` user logs in. TABLE 10-3 describes the environment variables.

Note – Do not change the values for the following environment variables, except for SUNW_HOSTNAME.

TABLE 10-3   Environment Variables

SUNW_HOSTNAME	Name of the domain controlled by the SSP. You set this variable to the host name of the domain on which you are performing operations.
SSPETC	Path to the directory containing miscellaneous SSP-related files.
SSPLOGGER	Path to the directory containing the platform logs and directories for domain logs.
SSPOPT	Path to the SSP package binaries, libraries, and object files.
SSPVAR	Path to the directory where modifiable files reside.

Miscellaneous SSP Procedures

This appendix describes how to do the following:

- Change the SSP name
- Deconfigure the host
- Deconfigure the SSP

Changing the SSP Name

If you need to change the name of your SSP, note that you must modify numerous files on both the SSP and the domains.

▼ To Rename the SSP

1. **Do one of the following:**

   - If you are renaming the main SSP, replace the SSP name with the new name for the main SSP in the following files:

     - `/etc/hosts`
     - `/etc/nodename`
     - `/etc/hostname.`*interface*
     - `/etc/net/ticlts/hosts`
     - `/etc/net/ticots/hosts`
     - `/etc/net/ticotsord/hosts`

   - If you are renaming the spare SSP, replace the name of the spare SSP with its new name in the files listed above and the
     `/$SSPVAR/.ssp_private/ssp_to_domain_hosts` file.

2. **On each domain, replace the old SSP name with the new name in the** `/etc/hosts` **and the** `/etc/ssphostname` **files.**

3. **Reboot the SSP.**

Deconfiguring a Domain

The following procedure undoes a domain configuration.

▼ To Deconfigure a Host

1. **For the domain to be deconfigured, retain a copy of the** `/etc/vfstab` **if the system was pre-configured.**

2. **Log into the domain as superuser and deconfigure the domain.**

```
#  /usr/sbin/sys-unconfig
```

3. **Repeat Step 1 and Step 2 on all domains that are to be deconfigured.**

Note – Each deconfigured domain is shut down automatically.

Deconfiguring the SSP

The deconfiguration of the SSP causes the following environment variables to be removed from the SSP:

- `/tftpboot`
- `${SSPVAR}/.ssp_private/cb_config`
- `${SSPVAR}/.ssp_private/domain_config`
- `${SSPVAR}/.ssp_private/domain_history`
- `${SSPVAR}/.ssp_private/ssp_to_domain_hosts`

> **Note –** Be sure to deconfigure the domains (explained in the previous procedure), before deconfiguring the SSP. Also, if you plan to reuse the SSP that you are deconfiguring, run `ssp_backup`(1M) before deconfiguration.

▼ To Deconfigure the SSP

1. Log in as user `ssp`, and set SUNW_HOSTNAME to switch the platform name.

```
ssp% domain_switch platform_name
```

2. At the prompt, type:

```
ssp% domain_remove -d domain_name
```

3. Repeat Step 2 for each domain.

4. Log in to the SSP as superuser and type:

```
ssp# /opt/SUNWssp/bin/ssp_unconfig
ssp# /usr/sbin/sys-unconfig
```

> **Note –** The `sys-unconfig` command automatically shuts down the SSP.

Glossary

Alternate Pathing (AP)   AP enables you to set up an alternate path to system components in case of failure, repair, or replacement.

AP   See *Alternate Pathing* (AP).

application specific integrated circuit (ASIC)   In the Sun Enterprise 10000 system, any of the large main chips in the design, including the UltraSPARC processor and data buffer chips.

arbitration stop   A condition that occurs when one of the ASICs on a Sun Enterprise 10000 system detects a parity error or equivalent fatal system error. Bus arbitration is frozen, so all bus activity stops. The system is down until the SSP detects the condition by polling the Control and Status Registers CSRs of the Address Arbiter ASICs through JTAG, and clears the error condition.

ASIC   See *application specific integrated circuit* (ASIC).

Automated Dynamic Reconfiguration (ADR)   The dynamic reconfiguration of system boards accomplished through commands that can be used to automatically attach, move, or detach system boards and obtain board status information. You can run these commands interactively or in shell scripts.

automatic failover   The automatic switchover of the main SSP to its spare or the primary control board to its spare when a failure in the operation of the main SSP occurs.

BBSRAM   See *bootbus SRAM* (BBSRAM).

blacklist A text file that hpost(1M) reads when it starts up. The blacklist file specifies the Sun Enterprise 10000 system components that are not to be used or configured into the system. The default path name for this file can be overridden in the .postrc file (see postrc(4)) and on the command line.

board descriptor array The description of the single configuration that hpost(1M) chooses. It is part of the structure handed off to OBP.

bootbus A slow-speed, byte-wide bus controlled by the processor port controller ASICs, used for running diagnostics and boot code. UltraSPARC starts running code from bootbus when it exits reset. In the Sun Enterprise 10000 system, the only component on the bootbus is the BBSRAM.

bootbus SRAM (BBSRAM) A 256-Kbyte static RAM attached to each processor PC ASIC. Through the PC, it can be accessed for reading and writing from JTAG or the processor. Bootbus SRAM is downloaded at various times with hpost(1M) and OBP startup code, and provides shared data between the downloaded code and the SSP.

command synchronization The recovery of user- created commands interrupted by an automatic failover.

control and status register (CSR) A general term for any embedded register in any of the ASICS in the Sun Enterprise 10000 system.

CSR See *control and status register* (CSR).

data synchronization The process of synchronizing SSP configuration and user-created files between the main and spare SSP for failover purposes.

DIMM See *dual in-line memory module* (DIMM).

domain A set of one or more system boards that acts as a separate system capable of booting the OS and running independently of any other domains.

DRAM See *dynamic RAM* (DRAM).

dual in-line memory module (DIMM) A small printed circuit card containing memory chips and some support logic.

dual power grid Redundant power supplies on the Sun Enterprise 10000 system. The power supplies are divided into two grids, with each grid wired to independent AC sources. Up to eight power supplies are available for each grid, for a total of 16 power supplies.

dynamic RAM (DRAM) Hardware memory chips that require periodic rewriting to retain their contents. This process is called "refresh". In the Sun Enterprise 10000 system, DRAM is used only on main memory SIMMs and on the control boards.

Dynamic Reconfiguration (DR) The logical attachment and detachment of system boards to and from the operating system without causing machine downtime. This feature is used to add a new system board, reinstall a repaired system board, or modify the domain configuration on the Sun Enterprise 10000 system.

Ecache See *external cache* (Ecache).

external cache (Ecache) A 0.5-Mbyte to 4-Mbyte synchronous static RAM second-level cache local to each processor module. Used for both code and data. This is a direct-mapped cache.

JTAG A serial scan interface specified by IEEE standard 1149.1. The name comes from Joint Test Action Group, which initially designed it. See JTAG+.

JTAG+ An extension of JTAG, developed by Sun Microsystems Inc., which adds a control line to signal that board and ring addresses are being shifted on the serial data line. Often referred to simply as JTAG.

InterDomain Networks (IDN) High-speed networking supported between dynamic system domains within a single Sun Enterprise 10000 platform. Domains can communicate with each other using standard networking interfaces, such as Transmission Control Protocol/Internet Protocol (TCP/IP).

IP Multipathing (IPMP) Internet Protocol multipathing. Enables continuous application availability by load balancing failures when multiple network interface cards are attached to a system. If a failure occurs in a network adapter, and if an alternate adapter is connected to the same IP link, the system switches all the network accesses from the failed adapter to the alternate adapter. When multiple network adapters are connected to the same IP link, any increases in network traffic are spread across multiple network adapters, which improves network throughput.

OBP See *OpenBoot PROM* (OBP).

OpenBoot PROM (OBP) A layer of software that takes control of the configured Sun Enterprise 10000 system from hpost(1M), builds some data structures in memory, and boots the operating system.

POST See *power-on self-test.*

power-on self-test (POST) A test performed by hpost(1M). This is the program that takes uninitialized Sun Enterprise 10000 system hardware and probes and tests its components, configures what seems worthwhile into a coherent initialized system, and hands it off to OBP.

.postrc A text file that controls options in hpost(1M). Some of the functions can also be controlled from the command line. Arguments on the command line take precedence over lines in the .postrc file, which takes precedence over built-in defaults. hpost -? postrc gives a terse reminder of the .postrc options and syntax. See postrc(4).

SBus A Sun Microsystems Inc. designed I/O bus, now an open standard.

SRAM See *static RAM (SRAM).*

static RAM (SRAM) Memory chips that retain their contents as long as power is maintained.

SSP See *System Service Processor (SSP).*

System Service Processor (SSP) A workstation or server containing software for controlling power sequencing, diagnostics, and booting of a Sun Enterprise 10000 system.

UltraSPARC The UltraSPARC processor is the processor module used in the Sun Enterprise 10000 system.

Index

control board executive (CBE), 4, 85
control board failover
   complete, 89, 90
   partial, 88
   recovery tasks, 92
   setfailover command, 89
   showfailover command, 91
   status, 91
control board server (CBS), 86, 100
control menu
   Hostview, 13
creating domains
   command line, 26
   Hostview, 24

D

daemons
   datasyncd, 70, 76, 106
   edd, 98
   fad, 101
   fod, 70, 101
   ntpd, 98
   obp_helper, 97
   SSP, 97
   xntpd, 98
data synchronization, 70, 76
   backup file, 77
   setdatasync command, 76
   showdatasync command, 80
   status, 80
DIMM, 116
domain_remove, 30
domain_switch, 31
domains, 1
   bringing up, 30
   creating, 24
      domain_create, 26
   domain name, 25
   removing, 30
   renaming, 34
   status of, 31
download_helper, 107
DRAM, 117
dual control boards, 4
dynamic reconfiguration, 2

E

edd, 98
edit menu
   Hostview, 13
environment variables, 108
   SSPETC, 109
   SSPLOGGER, 109
   SSPOPT, 109
   SSPVAR, 109
   SUNW_HOSTNAME, 109
event detector daemon, 98
external cache, 117

F

failover, 69
   causes, 71
   command synchronization, 69, 81
   control board, 88
   controlling, 72
   data synchronization, 69, 76
   detection points, 102
   modifying SSP resources, 74
   monitoring, 69
   recovery tasks, 84
   setfailover command, 72
   showfailover command, 75
   SSP, 67
   status, 75
failure button
   Hostview, 18
fan button
   Hostview, 17
Fan Status Display window, 57
fan tray display, 59
fans
   monitoring in Hostview, 57

H

help menu
   Hostview, 15
help window
   Hostview, 15
Hostview, 10