Sun microsystems

Sun Fire™ Link Fabric
Administrator's Guide

Adobe PostScript

Contents

Preface

This document describes the tools and methods to be used in configuring and managing clusters of Sun Fire™ 6800 and 15K/12K systems that are connected by the Sun Fire Link interconnect technology.

How This Book Is Organized

Chapter 1 introduces the principal software components that play a role in the configuring and administration of Sun Fire Link clusters.

Chapter 2 explains how to initialize adminstrative security for a Sun Fire Link cluster.

Chapter 3 describes the procedure for creating cluster *fabrics*, logical containers for holding configuration building blocks.

Chapter 4 describes the procedure for integrating Sun Fire Link managment software into the Sun Management Center administrative domain.

Chapter 5 explains how to populate Sun Fire Link fabrics with configuration components (nodes and switches).

Chapter 6 describes the procedure for creating and configuring Sun Fire Link *partitions*, a logical network of cluster nodes that are able to communicate with each other over the Sun Fire Link interconnect.

Chapter 7 explains how to delete nodes, switches, partitions, and fabrics.

Chapter 8 describes the procedure for acquiring information about the state of communication paths, called *links*, between nodes.

Chapter 9 explains how to use the Sun Fire Link graphical user interface to manage a Sun Fire Link cluster.

Chapter 10 explains how to collect status and error information that can be useful for troubleshooting problems in a Sun Fire Link cluster.

Appendix A explains how to use the Sun Fire Link command-line interface to configure a Sun Fire Link cluster.

Appendix B contains a sample Sun Fire Link configuration file that can be used as a template for configuring Sun Fire Link clusters by hand.

Using UNIX Commands

This document might 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:

- *Solaris Handbook for Sun Peripherals*
- AnswerBook2™ online documentation for the Solaris™ operating environment
- Other software documentation that you received with your system.

Typographic Conventions

Typeface[*]	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. Replace command-line variables with real names or values.	Read Chapter 6 in the *User's Guide*. These are called *class* options. You *must* be superuser to do this. To delete a file, type `rm` *filename*.

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

Shell Prompts

Shell	Prompt
C shell	*host-name*%
C shell superuser	*host-name*#
Bourne shell and Korn shell	$
Bourne shell and Korn shell superuser	#

.

Related Documentation

Application	Title	Part Number
Task Map	*Task Map for Getting Started With Sun Fire Link Networks*	816-0041
Hardware installation	*Sun Fire Link Hardware Installation Guide*	806-1396
Software installation (with Sun MC 3.0)	*Sun Fire Link Software Installation Guide*	806-1401
Software installation (with Sun MC 3.5)	*Sun Fire Link Software Installation Guide*	817-0401
Overview	*Sun Fire Link System Overview*	816-0697
Administration	*Sun Fire Link Fabric Administrator's Guide*	806-1405
Service	*Sun Fire Link Service Manual*	805-7363
Switch	*Sun Fire Link Switch Installation and Service Manual*	806-1397

Accessing Sun Documentation

You can view, print, or purchase a broad selection of Sun documentation, including localized versions, at:

`http://www.sun.com/documentation`

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can email your comments to Sun at:

`docfeedback@sun.com`

Please include the part number (806-1405-11) of your document in the subject line of your email.

Sun Fire Link Software Components

This chapter provides an overview of the Sun Fire Link software components that are of interest for managing a Sun Fire Link cluster. It contains the following sections:

Sun Fire Link Administration Overview

You configure and manage Sun Fire Link clusters using the services of a software module called the Sun Fire Link Manager (FM). The FM provides centralized management of the Sun Fire Link clustering resources in the Sun Fire system chassis.

The FM runs on a central administration host called the *management station*, which is external to the Sun Fire Link cluster. This system is connected to the hosts in the Sun Fire Link cluster by a TCP/IP network. See FIGURE 1-1.

The FM includes add-ons to the Sun Management Center (Sun MC) software. These add-ons extend the administration capabilities of Sun Management Center software, allowing you to manage Sun Fire Link clusters as part of a Sun Management Center administrative domain. The Sun Fire Link management station and the Sun Management Center server can be the on the same host system or they can be on separate systems.

The FM add-ons support an FM-specific set of Sun Management Center console windows and dialog boxes, which are used exclusively to monitor and control Sun Fire Link clusters and components. This FM-specific subset of the Sun Management Center console is sometimes referred to in Sun Fire Link documentation as the Sun Management Center/FM *console*.
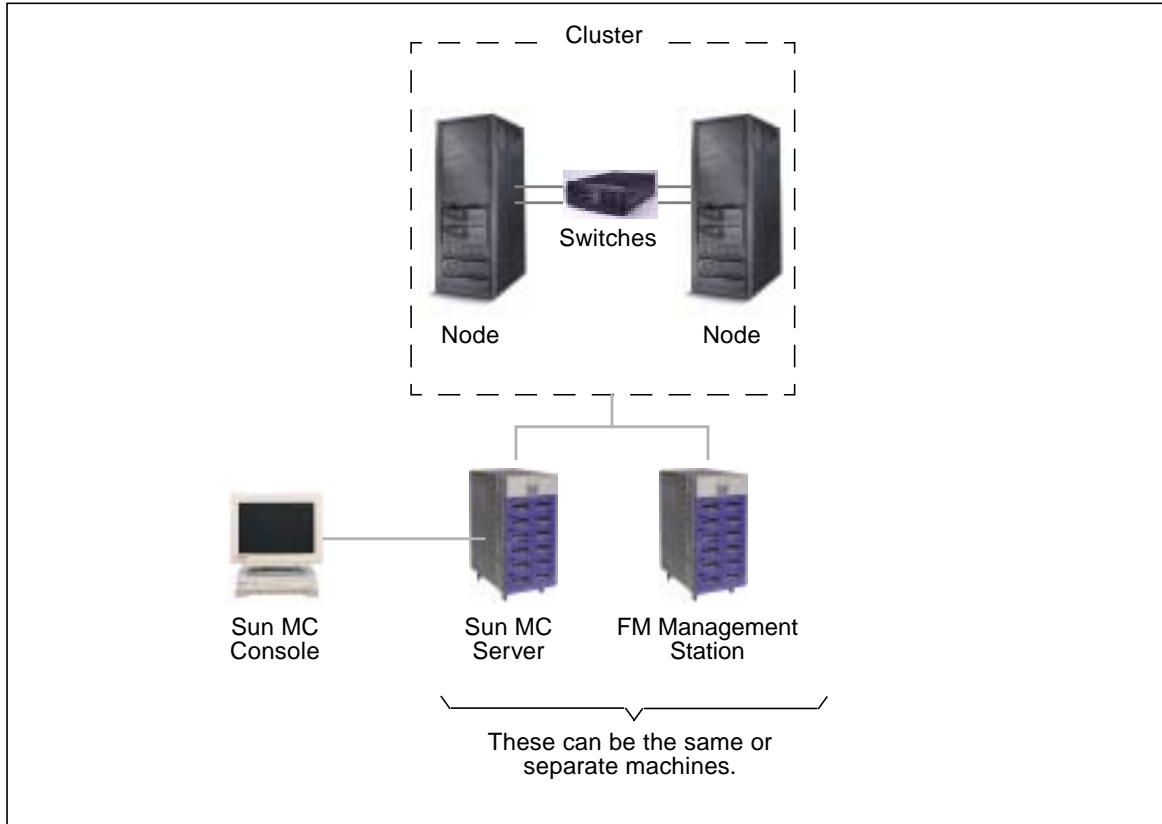
FIGURE 1-1 Sun Fire Link Cluster With Sun Management Center Server and FM Management Station

The FM also supports a command-line interface, which can be used in place of the Sun Management Center server and the FM console. However, because the Sun Management Center software provides a more effective and convenient means to administer Sun Fire Link clusters, this manual focuses on the Sun Management Center console interface. Use of the command-line interface is explained in Appendix A.

Sun Fire Link Administration Software

Software Architecture

FIGURE 1-2 illustrates the major software components in the Sun Fire Link system and shows the principal control and information paths between them.

The Sun Fire Link administration software applies the Sun Management Center console-server-agent architecture to the support of Sun Fire Link clusters. The following is a brief description of this arrangement.

- The Sun Management Center console is the user interface through which you control and monitor the Sun Fire Link cluster components. FM-specific windows are accessed by opening a detailed view of a Sun Fire Link component called a *fabric*. Sun Fire Link fabrics are described in "Sun Fire Link Manager" on page 5 and "Creating and Activating Sun Fire Link Fabrics" on page 9.

- The FM is the Sun Fire Link equivalent to the Sun Management Center server. The FM executes Sun Fire Link administration tasks you initiate by sending instructions to Sun Fire Link proxies that are running on managed components. The FM also forwards configuration, status and error information it receives from the agents to the console.

- WRSM proxies running on each node in the Sun Fire Link cluster carry out the instructions they receive from the FM. They also report local status and other management information to the FM. In addition, there are agent modules running on the FM Management Station that serve as interfaces to the FM console, Sun Management Center server, and Switch System Controllers (SSCs), respectively.

More detailed descriptions of the Sun Fire Link software entities are provided later.
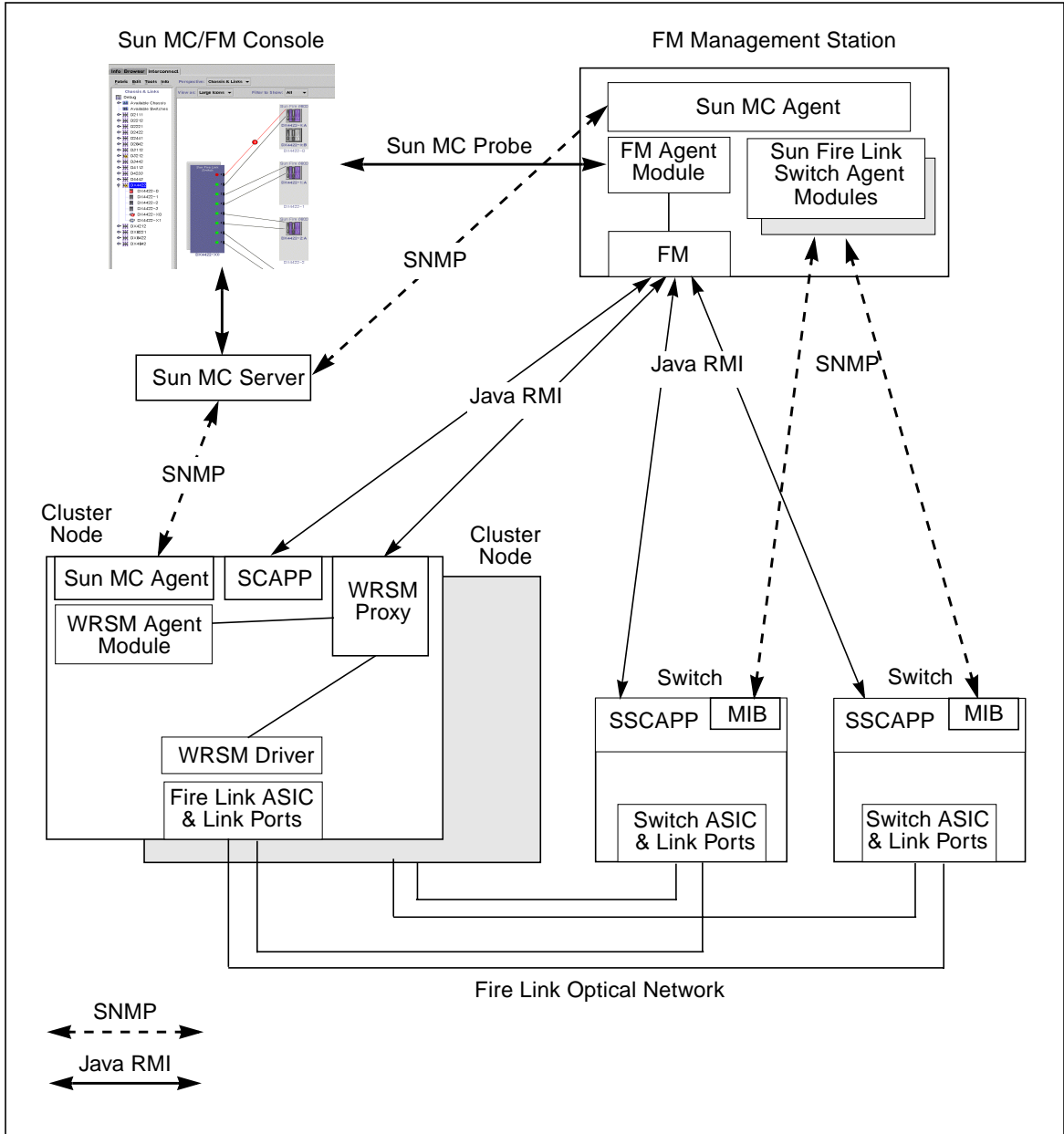
FIGURE 1-2 Sun Fire Link Software Architecture (Including Sun Management Center Server and Console)

Sun Management Center/FM Console

The Sun Management Center/FM console provides user access to Sun Fire Link objects that are being managed by the FM. It's principal responsibilities include:

- Providing a graphical user interface to the FM for configuring and managing Sun Fire Link networks.
- Collecting status from the FM.
- Displaying the status of network links.
- Displaying alarm information regarding failed or degraded Sun Fire Link components.

The Sun Management Center/FM console sends cluster administration instructions to the FM through the FM agent module, using the standard Sun Management Center SNMP interface. It also receives status and other information about Sun Fire Link objects through the same FM/FM agent module path.

Sun Management Center Agent and FM Agent Module

The FM agent module and the Sun Management Center agent provide interface support to the FM, handling communication between the FM and the Sun Management Center and FM consoles.

Sun Fire Link Manager

The Sun Fire Link Manager (FM) executes Sun Fire Link cluster management tasks that are initiated from the FM console. Its responsibilities include:

- Managing a set of Sun Fire Link nodes and switches that are in a logical collection referred to as a *fabric*.
- Evaluating the state of the existing physical network connections between the nodes and switches in a fabric and calculating valid network configurations based on these evaluations.
- Setting up and modifying network configurations.
- Configuring network *partitions*.
- Monitoring the state of physical *links* and logical *routes* within a Sun Fire Link fabric.
- Passing status and other information received from WRSM proxies to the Sun Management Center/FM console.

In the Sun Fire Link context, a fabric is an abstract entity used by administrators to manage a specified set of nodes and switches in a Sun Fire Link cluster.

A Sun Fire Link partition is a set of a fabric components that have been configured into an RSM-based network. That is, the FM creates a partition by enabling the pathways that connect a specified set of nodes, so processes running on nodes in the partition can communicate with each other.

The term *link* refers to a Sun Fire Link optical cable connection between two Sun Fire Link optical modules. The cable can connect two nodes directly or a node to a switch. These are called direct-connect and switched configurations, respectively.

The term *route* refers to a logical communication path between two nodes in a Sun Fire Link cluster. A route can consist of one, two, or four physical links. If a link component in a route fails during operation, the operation will be able to continue at a lower performance level using the route's remaining links. Also, if a route includes Sun Fire Link switches, the logical connections can be redirected to steer a route around failed components.

Sun Fire Link Manager Agents

An FM agent is installed on each standalone hardware entity that is managed by and/or reports to the FM. These entities include the node System Controllers, each Solaris instance, and all Sun Fire Link switches. The FM agent running on Solaris is called the WRSM proxy.

The FM and the FM agents have a client/server relationship where the FM is the client. FM agents handle configuration requests from the FM. This includes the following responsibilities:

- Accepting configuration requests from the FM.
- Forwarding FM requests to the local Sun Fire Link device driver.
- Storing the current configuration for use on reboot.

Sun Fire Link Device Drivers

Sun Fire Link device drivers (labeled "WRSM driver" in FIGURE 1-2) manage the local Sun Fire Link *ASIC*s and optical link interfaces on the System Controllers, switches, and Solaris instances. ASICS (Application Specific Integrated Circuits) are the complex circuit chips that execute the interface management operations.

The Sun Fire Link device drivers have the following responsibilities:

- Communicating with Sun Fire Link device drivers on other nodes to identify network link endpoints
- Installing network route maps

- Programming striping levels
- Monitoring link status
- Generating link and route events for reporting to the Sun Management Center agent, which forwards them to the Sun Management Center server and the Sun Management Center/FM console

SNMP Agents

SNMP agents collect network status for the Sun Management Center console. The agents are responsible for:

- Polling for status.
- Notifying the Sun Management Center server of events with SNMP traps.
- Returning information requested by the Sun Management Center server.

Sun Management Center Agent/System Controller Proxy, Switch Proxy

A separate Sun Management Center proxy runs for each agent that is not part of the Sun Managment Center software. These proxies provide interfaces between the Sun Management Center server and the SNMP agents for the System Controller and the Switch System Controller.

Summary of Sun Fire Link Administration Tasks

This section provides a quick overview of the principal activities involved in configuring and managing Sun Fire clusters. FIGURE 1-3 provides a capsule view. Descriptions of the tasks follow the figure.

Note – When entering text, whether in supplying configuration information to the Sun Management Center GUI or editing the XML-based fabric configuration files, you must use only ASCII characters. For example, do not enter domain or partition names that include non-ASCII characters.
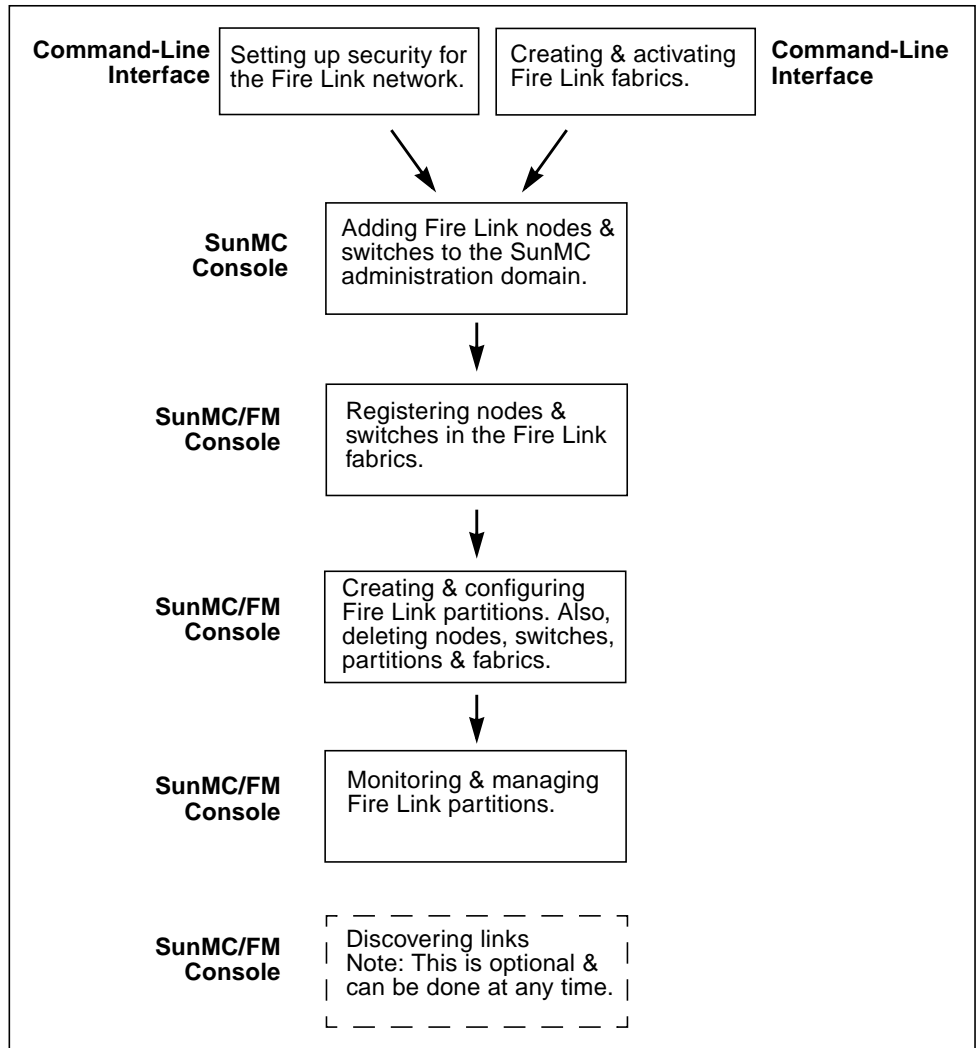
FIGURE 1-3 Principal Sun Fire Link Administration Tasks

Setting Up Network Administration Security

A password-protected network access group, sfladmin, must be created and added to /etc/group. The group must contain the login names of everyone authorized to have administrative access to the Sun Fire Link network. This is done using Solaris commands and methods for editing files and setting passwords. The procedures for adding this layer of security to the Sun Fire Link network are described in Chapter 2.

Creating and Activating Sun Fire Link Fabrics

The first step in managing a Sun Fire Link cluster is to create and activate one or more fabrics. A fabric is a logical container that provides a convenient way to manage a specified set of nodes and switches as a named group. At this stage, you simply create an empty container and start a set of associated daemons. A set of Sun Fire Link commands are provided for creating, activating, deactivating, and removing Sun Fire Link fabrics. This process is described in Chapter 3.

Note – All the Sun Fire Link cluster resources in a given Sun Fire system chassis will be contained in the same fabric. This ensures that there are no configuration conflicts when creating multiple network partitions.

Adding Sun Fire Link Components to a Sun Management Center Administration Domain

If you are using Sun Management Center software to manage the Sun Fire Link network, you must load and discover the FM agent module on the FM host so that Sun Management Center software can create composite objects for each of the fabrics that is to be managed.

You also use the Sun Management Center Discovery Manager to discover all the nodes and switches in the Sun Fire Link cluster that is to be managed. This procedure allows Sun Management Center software to access driver-level node and switch information that is not available through the FM.

Registering Nodes and Switches in the Sun Fire Link Fabrics

Once you have created and activated a fabric, you specify which nodes and switches you want to be in that fabric. For this step, you need to know the IP address of the Solaris domain (WRSM proxy) and the security credentials in each node that you plan to register. If your cluster includes switches, you will also need to know the community password and security credentials that are applicable to the switches. This is described in Chapter 5.

Creating and Configuring Sun Fire Link Partitions

Once you have registered nodes and switches in a fabric, you can create and configure partitions. A Sun Fire Link partition is a set of nodes and switches (if used) that are configured to communicate with each other over logical Sun Fire Link routes that have been set up and are managed by the FM. You perform this step by creating a partition with a logical name and then adding nodes and switches to that partition.

You also specify the level of striping that you want the FM to use when setting up the routes between the nodes. The choices are:

- 1 – Each route consists of a single bidirectional physical link; in other words, data are not striped.
- 2 – Each route consists of two bidirectional links. If one link fails, communication continues over the remaining link.
- 4 – Each route consists of four bidirectional links. If one link fails, communication continues over the remaining links.

The procedures for creating partitions are described in Chapter 6. Procedures for deleting partitions, nodes, and fabrics are described in Chapter 7.

Note – The terms *cluster* and *partition* are synonymous in the Sun Fire Link context.

Discovering Links

You can use the link discovery method to get a report on the current state of the connections between the nodes and switches in the cluster. This capability is provided as an optional information collection tool. This method is not needed for any cluster setup or configuration task. This is described in Chapter 8

Monitoring and Managing Sun Fire Link Partitions

When the Sun Fire Link cluster is in use—that is, when there are applications running on the cluster—you use the Sun Management Center/FM console to monitor the status of the Sun Fire Link network, change its configuration, and diagnose link problems that might occur. You use the Sun Management Center/FM console to manage the Sun Fire Link cluster in much the same way you use the Sun Management Center console to manage Solaris hosts. Instructions for monitoring partitions are provided in Chapter 9.

Setting Up Secure Administrator Access to Sun Fire Link Clusters

Administrative security for Sun Fire Link clusters is based on the following controls:

- Administration station (FM host) – A special Sun Fire Link user name, `sfluser`, is required for access to the FM. The `sfluser` name must be included in the Sun Fire Link administration group `sfladmin`, which is specified in `/etc/group`. See "To Set Up Security on the FM Management Station" on page 12 for details.
- WRSM proxy – Access to Sun Fire Link domains is restricted to users whose user names are specified in the group `sfladmin`. An associated password is also required. See "To Set Up WRSM Proxy Security on Cluster Nodes" on page 12 for details.
- Domain consoles – A password is required for access to each domain console in a cluster. For Sun Fire 6800 sytems, user names are not required. For Sun Fire 15K/12K systems, a user name as well as a password are required.
- Switch System Controller (SSC)command-line interface – A password is required to access the SSC command-line interface. This password is ordinarily established when the switch is installed. See the *Sun Fire Link Switch Installation and Service Manual* for more information.
- SSC FM interface – A community password is required to enable access to the switches through the FM. See "To Provide an RMI Password for the FM/SSC Interface" on page 14 for details.

Note – All these security procedures must be performed before the Sun Fire Link software can be used. Failure to do so will result in insufficient privileges, leading to access problems that will prevent Sun Fire Link operations from functioning properly.

▼ To Set Up Security on the FM Management Station

1. Log in to the FM host.

```
% rlogin node-name
password: password
%
```

2. Become superuser.

```
% su
password: su-password
#
```

3. Edit `/etc/group`, **adding the user name** `sfluser` **to the** `sfladmin` **group.**

```
/etc/group
root::0:root
other::1:
bin::2:root,bin,daemon
sys::3:root,bin,sys,adm
sfladmin::4:sfluser
```

4. Create the user `sfluser` **belonging to the group** `sfladmin`.

```
# useradd –g sfladmin [other-options] sfluser
```

▼ To Set Up WRSM Proxy Security on Cluster Nodes

1. Log in to one of the cluster nodes.

```
% rlogin node-name
password: password
%
```

2. **Become superuser.**

```
% su
password: su-password
#
```

3. **Edit** `/etc/group`**, adding a list of login names of the users who are to have access to cluster domains.**

   In the following example, the user names `fmuser1` and `fmuser2` are added to the `sfladmin` group.

```
/etc/group
root::0:root
other::1:
bin::2:root,bin,daemon
sys::3:root,bin,sys,adm
sfladmin::4:fmuser1,fmuser2
```

4. **Create the users specified in Step 3 with membership in the** `sfladmin` **group.**

```
# useradd -g sfladmin [other-options] fmuser1
# useradd -g sfladmin [other-options] fmuser2
```

5. **Assign each new user a password.**

6. **Repeat Step 1 through Step 5 for the other domains in the Sun Fire Link cluster.**

Note – If you edit the XML configuration file by hand, you must enter these user names and corresponding passwords in the `<host_user> </host_user>` and `<host_password> </host_password>` fields in the file. If you use the Sun Management Center GUI facility to create fabrics, the configuration file will be created automatically, including these names and passwords.

▼ To Create a Password for Domain Consoles

● **Create a domain shell password for each domain. Refer to the** *Sun Fire 6800/4810/4800/3800 Systems Platform Administration Manual* **or the** *System Management Services (SMS) Administrator Guide for the Sun Fire 15K/12K Systems* **for instructions.**

You will need to specify these domain console passwords when you register these cluster nodes in a fabric.

- When using Sun Management Center/FM console to configure a fabric, you enter these passwords in the node registration dialog box. See "To Register a Node" on page 33 for details.
- If you configure a fabric by editing an XML configuration file, use this password in the `<sc_password></sc_password>` field associated with the cluster node. See "Creating an XML Configuration File" on page 175 for details.

Note – These passwords belong only in the cluster node definition sections of the XML configuration file. Do not enter them in the `<sc_password></sc_password>` field in a section that defines a switch.

▼ To Provide an RMI Password for the FM/SSC Interface

To control FM access to the SSCs in a cluster, create an RMI community password, as described below:

1. **Connect to the SSC on one of the Sun Fire Link switches via its serial port. It is called** `switch1` **in the following example.**

   For the requested password, enter the password that was specified as part of the switch installation procedure. Refer to the *Sun Fire Link Switch Installation and Service Manual* for more information about this password.

   ```
   # site-specific-command switch1
   Enter Password: switch_cli_password
   switch1:SSC>
   ```

2. **Run the** `rmi_password` **command and, when prompted, enter the community password for the switch.**

   ```
   switch1:SSC> rmi_password
   Enter Password:
   ```

3. **Repeat Step 1 and Step 2 on the other switches in the cluster.**

   You will need to specify these switch controller passwords when you register the switch nodes in a fabric.

- When using Sun Management Center/FM console to configure a fabric, you enter these RMI passwords in the switch node registration dialog box. See "To Register a Node" on page 33 for details.
- If you configure a fabric by editing an XML configuration file, specify this password in the `<sc_password></sc_password>` field associated with the switch. See "Creating an XML Configuration File" on page 175 for details.

Note – Use these passwords only in the switch definition sections of the XML configuration file. Do not enter them in the `<sc_password></sc_password>` field in a section that defines a cluster node.

Creating and Activating Sun Fire Link Fabrics

Before you can start configuring Sun Fire Link partitions, you must create and activate a fabric, which is described in this chapter. This chapter also explains how to:

- Initialize, reset, and verify fabrics.
- List fabrics that already exist.
- Stop the daemons for a specified fabric.
- Stop the daemons for all fabrics with a single command.
- Delete a fabric.

If you will be using Sun Management Center software to manage your Sun Fire Link cluster, you must create a composite object for the fabric in Sun Management Center administrative domain. This is described in Chapter 4.

You must also populate the fabric with the nodes and switches that will be used in the creation of partitions. The simplest way to do this is by using the Sun Management Center console. This approach is described in Chapter 5. Alternatively, you can edit an XML fabric configuration file by hand. This approach is described in Appendix A and Appendix B.

Note – For Sun Fire 15K/12K systems, only one fabric can be created per chassis. Multiple fabrics can be created on a single Sun Fire 6800 chassis.

Use the FM command-line interface (CLI) to create, start, list, stop, and delete fabrics. The basic CLI procedures for these operations are described below. More detailed descriptions of their syntax are provided in Appendix A. The commands are also described in the man page fabric(1m).

> **Note –** The FM CLI also supports three commands that can be used to configure partitions, retrieve information about fabrics, and verify fabric connectivity. These commands are `wcfmconf(1m)`, `wcfmstat(1m)`, and `wcfmver(1m)`, respectively. Use of these commands is discussed in Appendix A. The commands also have man pages.

▼ To Create a Fabric

1. **If you are not already logged in to the FM server as superuser, do so now.**

2. **Go to the** `/opt/SUNWwcfm/bin` **directory.**

   ```
   fm-host# cd /opt/SUNWwcfm/bin
   ```

3. **Enter the** `createfabric` **command, specifying the name of the fabric as an argument.**

   For example, to create a fabric named `fabric1`, enter

   ```
   fm-host# ./createfabric fabric1
   Created fabric log directory "/var/opt/wcrsmct/SUNWwcfm/config/
   fabric1/log"
   Created fabric data directory "/var/opt/wcrsmct/SUNWwcfm/config/
   fabric1/cfg"
   Created policy file "/var/opt/wcrsmct/SUNWwcfm/config/fabric1/
   cfg/fabric1.policy"
   ```

   Creating the fabric also causes to two fabric-specific directories, `log` and `cfg`, and a fabric-specific policy file to be created. The two directories and policy file have as their root the base directory that is created when the Sun Fire Link manager software is set up following installation. In the previous example, `/var/opt/wcrsmct` is the base directory. The roles of the `log` and `cfg` directories and policy file are described below:

   - Messages relating to the fabric are written to a file in the `log` directory. The file is named *fabric-name*.log, where *fabric-name* is the name of the fabric.
   - When the FM configures the fabric by enabling links between network endpoints, it creates an XML-formatted file that defines the partition and topology aspects of the configuration. This file is stored in the `cfg` directory and is given the name of the fabric, with the extension ".xml". For example, the XML configuration file for the fabric `testfab` would be named `testfab.xml`.

■ The *fabric-name*`.policy` file is created to grant required code permissions to the FM.

▼ To Start a Fabric

Note – A fabric must be created before it can be started.

---

1. **If you are not already logged in to the FM server as superuser, do so now.**

2. **Go to the** `/opt/SUNWwcfm/bin` **directory.**

```
fm-host# cd /opt/SUNWwcfm/bin
```

3. **Enter the** `startfabric` **command, specifying the name of the fabric to be started.**

   In this example, the fabric is `fabric1`.

```
fm-host# ./startfabric fabric1
...
Fabric "fabric1" configured correctly
RMI registry up and running on port 1099
...
Fabric Manager Started
...
fabric1 Bound To Registry
```

Note – If the shell prompt does not appear after the *fabricname* Bound To Registry message is displayed, press the Return key.

Running `startfabric` will start an FM daemon instance dedicated to the fabric and will also start the rmiregistry if it has not already been started.

▼ To Verify That a Fabric Is Running

The `listfabrics` command identifies all the fabrics that are currently running.

1. **If you are not already logged in to the FM server as superuser, do so now.**

2. **Go to the** `/opt/SUNWwcfm/bin` **directory.**

```
fm-host# cd /opt/SUNWwcfm/bin
```

3. **Enter the** `listfabrics` **command.**

```
fm-host# ./listfabrics
```

This command only lists fabrics that are actually running. If a fabric has been created but not started, it will not appear in the command output.

The names of running fabrics appear in the indented entry following the words: "`Registry Contains...`"

The following example shows `listfabrics` output for the case where no fabrics are running and there is no rmiregistry.

```
fm-host# ./listfabrics
Retrieving Objects From: //localhost:1099/
Registry does not exist.
```

The next example shows `listfabrics` output for the case where the rmiregistry exists, but no fabrics are running. This will happen when `stopfabric` is used to stop the sole running fabric, which results in having no running fabrics, but the rmiregistry is not removed.

```
fm-host# ./listfabrics
Retrieving Objects From: //localhost:1099/
Registry Contains 0 Objects
```

Note – If `killfabrics` is used to stop all running fabrics, the rmiregristry will be removed. See "To Stop a Fabric" on page 21 and "To Stop All Running Fabrics at Once" on page 22 for more information on `stopfabric` and `killfabrics`.

The next example shows listfabrics output for the case where two fabrics are running, fabric1 and fabric2.

```
fm-host# ./listfabrics
Retrieving Objects From: //localhost:1099/
Registry Contains 2 Objects
        rmi://localhost:1099/fabric1
        rmi://localhost:1099/fabric2
```

The next example shows listfabrics output for the case where a fabric is stopped in an unfinished state. In this case, fabric2 is shown as Not Responding. This can happen, for example, if the fabric is stopped by a fault event of some kind.

```
fm-host# ./listfabrics
Retrieving Objects From: //localhost:1099/
Registry Contains 2 Objects
        rmi://localhost:1099/fabric1
        rmi://localhost:1099/fabric2 (Not Responding)
```

When you see a fabric in this state, use the stopfabric command to remove it from the rmiregistry. See "To Stop a Fabric" on page 21 for information on the the stopfabric command. Use startfabric to restart the fabric and reenter it in the rmiregistry.

▼ To Stop a Fabric

1. **If you are not already logged in to the FM server as superuser, do so now.**

2. **Go to the** /opt/SUNWwcfm/bin **directory.**

```
fm-host# cd /opt/SUNWwcfm/bin
```

3. **Enter the** `stopfabric` **command, specifying the fabric name as an argument. This fabric must have been previously created and started using** `createfabric` **and** `startfabric`**.**

   In this example, the instance daemon for `fabric1` will be stopped and the `fabric1` entry will be removed from the rmiregistry.

```
fm-host# ./stopfabric fabric1
Found FM at [rmi://localhost:1099/fabric1]
Fabric "fabric1" stopped, the FM process may not exit for several seconds
```

   In the following example, `fabric2` instance daemon had been killed by some fault event and `stopfabric` will only remove the `fabric2` entry from the rmiregistry.

```
fm-host# ./stopfabric fabric2
Found FM at [rmi://localhost:1099/fabric2]
Fabric "fabric2" not responding. Removing it from the registry.
```

   Note – Be sure to allow time for the FM process to exit before deleting the fabric (see the last line in the previous output message).

▼ To Stop All Running Fabrics at Once

The `killfabrics` command stops every running fabric. It also removes the rmiregistry.

Note – The `killfabrics` command does not remove the fabric data directories associated with the fabrics being stopped. If desired, the fabric data directories can be removed individually by using the `deletefabric` command.

1. **If you are not already logged in to the FM server as superuser, do so now.**

2. **Go to the** `/opt/SUNWwcfm/bin` **directory.**

```
fm-host# cd /opt/SUNWwcfm/bin
```

3. **Enter the** `killfabrics` **command. In this example,** `fabric1` **and** `fabric2` **are killed.**

```
fm-host# ./killfabrics
Killed Fabric Manager instance for fabric "fabric1", PID 18599
Killed RMI Registry, PID 18589 on port 1099
Killed Fabric Manager instance for fabric "fabric2", PID 18615
Killed RMI Registry, PID 18615 on port 1099
```

▼ To Delete a Fabric

1. **If you are not already logged in to the FM server as superuser, do so now.**

Note – You do not need to stop a fabric before deleting it. If you specify a fabric to be deleted that is running, the `deletefabric` command will cause it to be stopped before deleting it.

2. **Go to the** `/opt/SUNWwcfm/bin` **directory.**

```
fm-host% cd /opt/SUNWwcfm/bin
```

3. **Enter the** `deletefabric` **command, specifying the name of the fabric to be deleted.**

   In this example, the instance daemon for `fabric1` has been stopped. The fabric data directory for `fabric1` will be deleted.

```
fm-host# ./deletefabric fabric1
A fabric was found in "/var/opt/wcrsm/SUNWwcfm/config/fabric1"
Do you wish to remove this fabric? [y or n]: y
Removing fabric "fabric1", directory removed "/var/opt/wcrsm/SUN
Wwcfm/config/fabric1"
```

The following example shows the output when the deletefabric command
specifies a fabric that is running. The instance daemon for fabric1 will be stopped
and the fabric data directory will be removed.

```
fm-host# ./deletefabric fabric1
Found FM at [rmi://localhost:1099/fabric1]
Fabric "fabric1" stopped, the FM process may not exit for several seconds

A fabric was found in "/var/opt/wcrsm/SUNWwcfm/config/fabric1"
Do you wish to remove this fabric? [y or n]: y
Removing fabric "fabric1", directory removed "/var/opt/wcrsm/SUNWwcfm/config/
fabric1"
```

Discovering the FM Agent Module and Sun Fire Link Proxy Agents

This chapter describes the procedure for creating a composite object in the Sun Management Center administrative domain for each Sun Fire Link fabric. See "Loading FM Agent Modules and Discovering Fabrics" on page 25 for the procedure.

This chapter also explains how to connect the Sun Management Center server to the WRSM agents on the nodes and SNMP agents on the switches so it can receive event notification and `kstat` results. See "Discovering the Nodes and Switches in the Sun Fire Link Cluster" on page 28.

Loading FM Agent Modules and Discovering Fabrics

Before a fabric can be configured by the Sun Fire Link Manager (FM), it must be established as composite object in the Sun Management Center administrative domain. To do this, you must:

- Discover the FM host. See "To Discover the FM Host System" on page 26.

- For each fabric to be managed, load a separate instance of the FM agent module onto the FM host. See "To Load the FM Agent Module" on page 27.

- Rediscover the FM host. This will create a composite object on the Sun Management Center server for each fabric. See "To Rediscover the FM Host" on page 27.

The procedural details are presented below.

▼ To Discover the FM Host System

Use the Sun Management Center Discovery Manager to discover the server that is host to the Sun Fire Link Manager. This *may* be the same server that you are using as the Sun Management Center server. The FM host is also referred to as the FM Management Station.

1. **Start the Sun Management Center program and open the Sun Management Center console.**

2. **Open the Discovery Requests window.**

   From the Tools pull-down menu, select Discover. If no requests have been previously created, the window appears blank.

3. **Click the Add button in the Discovery Requests window.**

   The New Discover Request dialog box appears.

4. **Fill in the fields described below.**

   - Enter a Request Name

     Enter the host name by which the FM server is known.

   - Enter the IP address of the FM host in both the Start IP Address and End IP Address boxes.

   - Check Use Default Port (161)

     Be sure the box is checked.

5. **Click OK to add the request.**

   A confirmation window appears.

6. **Click Yes to start the discovery immediately.**

   The discovery request appears in the Discovery Request window, highlighted in blue. The Status field changes quickly from New to Queued to Running.

   When the discovery completes, the status field changes to "Succeeded":

7. **Close the Discovery Request window.**

   Graphic symbols representing the FM host object appear in both the tree and topology panes of the Sun Management Center console screen.

Note – For a more comprehensive coverage of the Sun Management Center discovery process, see the Discovery Manager description in the *Sun Management Center Software User's Guide.*

▼ To Load the FM Agent Module

After the FM host has been discovered by the Sun Management Center software, load a separate instance of the FM agent module for each fabric that has been created.

1. **Open the Load Module dialog box for the FM host.**

   In the Sun Management Center topology pane, right-click on the FM host icon and highlight Load Module in the resulting pop-up menu.

   Note – You can also select the Load Module command from the Tools pull-down menu.

   Selecting the Load Module command brings up the Load Module window, which lists all the modules available to the Sun Management Center software.

2. **Highlight the Sun Fire Link FM Module entry in the Load Module window and click on OK.**

   The Load Module dialog box is replaced by the Module Loader dialog box.

3. **Fill in the fields of the Module Loader dialog box and click OK when you finish.**

   The various fields in this dialog box are described below:

Instance field	Enter text that identifies which instance of the FM agent module is being loaded. This text must begin with a letter.
Description field	Enter a description of this instance. The description should associate the instance with the applicable fabric.
Fabric Name field	Enter the name you gave the fabric when you created it.
FM Port field	Always specify port 1099.

4. **Click OK when you are done filling in the Module Loader dialog box fields.**

5. **Repeat the steps in this procedure for each fabric you have created and activated.**

▼ To Rediscover the FM Host

You need to discover the FM host again so that the Sun Management Center software can communicate with the FM agent module instances you just loaded and create composite objects for the fabrics those agents support.

● **Repeat the procedure described in "To Discover the FM Host System" on page 26.**

Discovering the Nodes and Switches in the Sun Fire Link Cluster

Perform a Sun Management Center discovery operation on every node and switch in the Sun Fire Link network. You do this following the same procedure you used to discover the FM host, except that you specify the IP addresses of the nodes and switches instead of the FM server's IP address.

For convenience, the host discovery steps are repeated below in a simplified form. For a complete description of the host discovery process, see the *Sun Management Center Software User's Guide.*

▼ To Discover the Nodes and Switches

1. **Open the Discovery Requests window.**

2. **Click the Add... button in the Discovery Requests window.**

3. **Fill in the Discovery Requests fields and click OK when finished.**

4. **Click Yes in the Confirm dialog box to start the discovery immediately.**

5. **When the Discovery Request window shows that the requested discovery succeeded, repeat Step 1 through Step 4 for each of the remaining nodes and switches in the Sun Fire Link network.**

Populating Fabrics With Nodes and Switches

You populate a fabric by registering nodes and switches in that fabric. This chapter explains the procedure. It also explains how to specify the appropriate network interface when registering nodes with multiple interfaces.

Note – The same procedure is used for registering both nodes and switches, except that there are slight differences in the information you enter in the Register Nodes dialog box.

The procedures described below are based on the assumption that the following conditions are in effect:

- The fabric to be populated has been created and started as described in Chapter 3 and discovered by the Sun Management Center software as described in Chapter 4.
- You are logged in to the Sun Management Center server.
- The logical host names have been assigned to the System Controllers on the FM host and the nodes.

Note – Be sure to use logical host names for the FM host and node SCs so that, should a primary System Controller fail over to the secondary System Controller, the Sun Fire Link software will continue to operate transparently.

The procedures in this chapter include:

- "To Register a Node" on page 33
- "To Verify That the Nodes Were Successfully Registered in the Fabric" on page 38

Specifying the Appropriate Network Interface

When you register a cluster node to be in a fabric, you are asked to enter the node's host name. Ordinarily, you would specify the host name contained in the `/etc/nodename` file. This would result in the FM using the system's default interface to manage the cluster nodes.

This section discusses two cases where the FM uses a nondefault interface, including the considerations these special cases require. It also briefly describes the simplest default case to provide a comparison for the nondefault cases.

Using the Default Interface for Both FM Management and Internode Communication

If the same interface will be used for both FM management of the node as well as for node-to-node communication, specify the host name contained in `/etc/nodename` when you register the node. In the example shown in FIGURE 5-1, this would be `node1-a`.

```
          Interface for both
          FM Management          ┌ ─ ─ ─ ─ ─ ─ ─ ┐
               and       ─────── │ /etc/nodename/node1-a │
      Sun Fire Link Network       └ ─ ─ ─ ─ ─ ─ ─ ┘
```
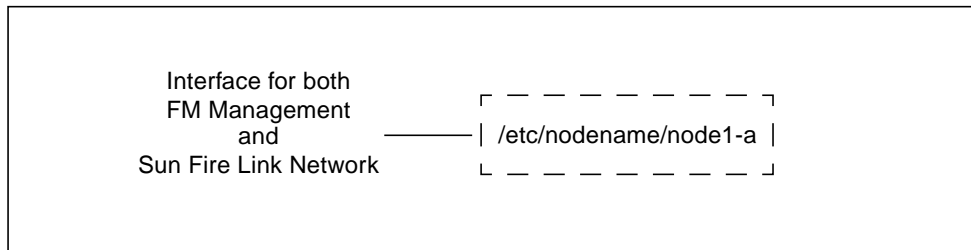
FIGURE 5-1 Schematic of a Cluster Node With a Single Network Interface

Using Separate Interfaces for FM Management and Internode Communication

FIGURE 5-2 illustrates a node that has one interface dedicated to FM management of the node and a separate interface that is used for node-to-node communication. In this example, the private FM managment network interface is `node1-a-mgmt` (`ce0`).

FM Management

node1-a-mgmt
(ce0)

Sun Fire Link Network
Communication

/etc/nodename/node1-a
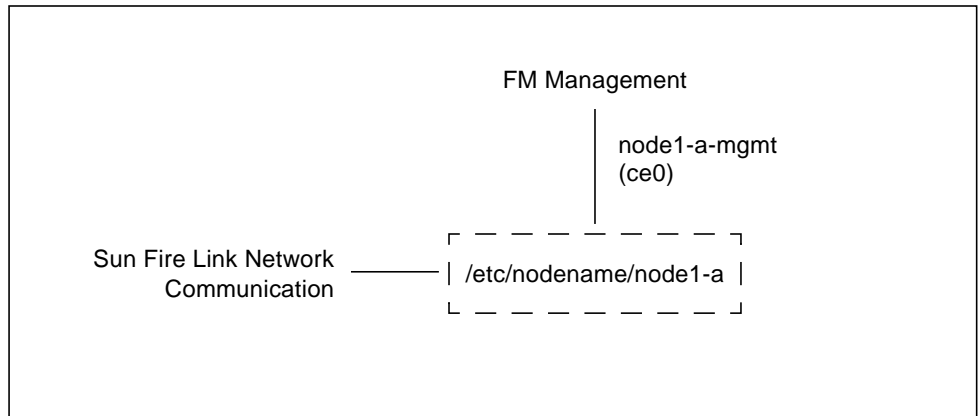
FIGURE 5-2 Schematic of a Cluster Node With a Private FM Management Interface and a Separate Node-to-Node Communication Network Interface

If you want to implement this kind of interface arrangement, keep the following in mind:

- When you register the node, specify the host name `node1-a`. This results in node-to-node traffic going over the default interface.
- The FM must have network access to this interface.
- Set up the RSM proxy on each cluster node so that FM communication will be routed through the private interface, `node1-a-mgmt` in this example. The procedure for doing this is described below.

▼ To Support a Private Interface for FM Management

1. **Examine the** `$BASEDIR/SUNWwrsmp/node_name.cfg` **file for each node to see if it matches the interface that you want the FM to use.** `$BASEDIR` **is the location of the** `SUNWwrsmp` **package.**

   If the name matches, no further action is needed. If not, go to Step 2.

2. **Edit the** `node_name.cfg` **file so that it matches the desired interface name, which is** `node1-a-mgmt` **in this example.**

3. **On each node where the** `node_name.cfg` **file was edited, stop and restart the RSM proxies.**

```
# /etc/init.d/wrsm_proxy stop
# /etc/init.d/wrsm_proxy start
```

Using a Nondefault Interface for Both FM Management and Internode Communication

FIGURE 5-3 illustrates the case where a nondefault interface, `node1-a-mgmt` in this example, is used for both FM management traffic and node-to-node communication. This might be necessary, for example, if the FM does not have network access to another interface.

Interface for both
FM Management and
Sun Fire Link Network

node1-a-mgmt
(ce0)

| /etc/nodename/node1-a |
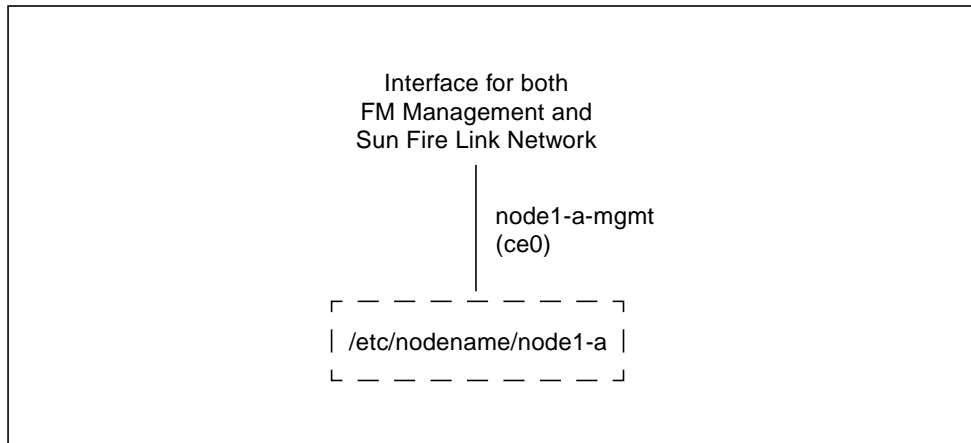
FIGURE 5-3 Schematic of a Cluster Node With FM Management and Node-to-Node Communication Sharing a Nondefault Interface

For this interface arrangement, keep the following in mind:

- When you register the node, specify the host name `node1-a-mgmt`. This results in node-to-node traffic going over this nondefault interface.
- The FM must have network access to this interface.
- Set up the RSM proxy on each cluster node so that FM communication will be routed through the same interface as the node-to-node traffic, `node1-a-mgmt`. Use the same procedure as described previously.

If Sun HPC ClusterTools software is installed on the cluster and multiple interfaces are listed in /etc/hostname, you must perform the following procedure to allow the hpc_rsmd daemon to construct network topology tables with valid contents.

▼ To Enable `hpc_rsmd` to Create Valid Topology Tables

1. **Copy the file** /tmp/.wci-hpc_config **to** /opt/SUNWhpc/etc/tmptopo **on each node.**

2. **Edit the file** /opt/SUNWhpc/etc/tmptopo **to contain the** /etc/nodename **for each node in the cluster.**

3. **Edit the file** /etc/init.d/sunhpc.hpc_rsmd **to contain the value**

```
parms=' -p /opt/SUNWhpc/etc/tmptopo'
```

4. **Remove the file** /tmp/.hpc*lock**.**

5. **Restart the** hpc_rsmd **daemon.**

```
# /etc/init.d/sunhpc.hpc_rsmd start
```

Registering Compute Nodes and Switches

▼ To Register a Node

1. **Open the Fabric Details dialog box for the fabric you are going to populate.**

   You can do this in either of the following ways. In the Sun Management Center console's main window,

   - Right click on the fabric icon and select Details from the resulting pop-up menu.
   - Double-click on the fabric icon.

2. **Open the Register Nodes dialog box.**

   Pull down the Fabric menu and select Register Nodes, as shown in FIGURE 5-4.
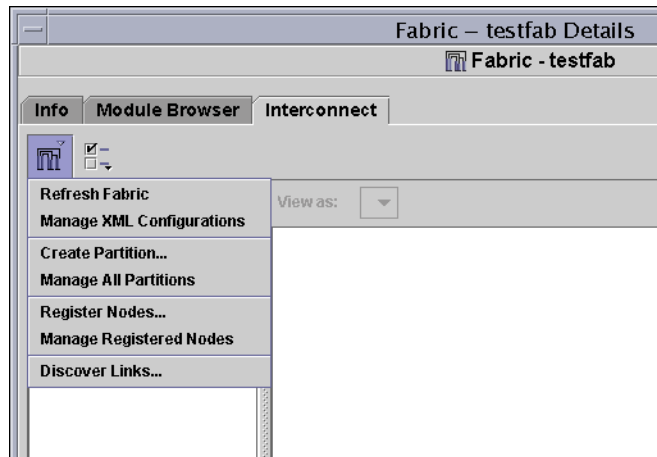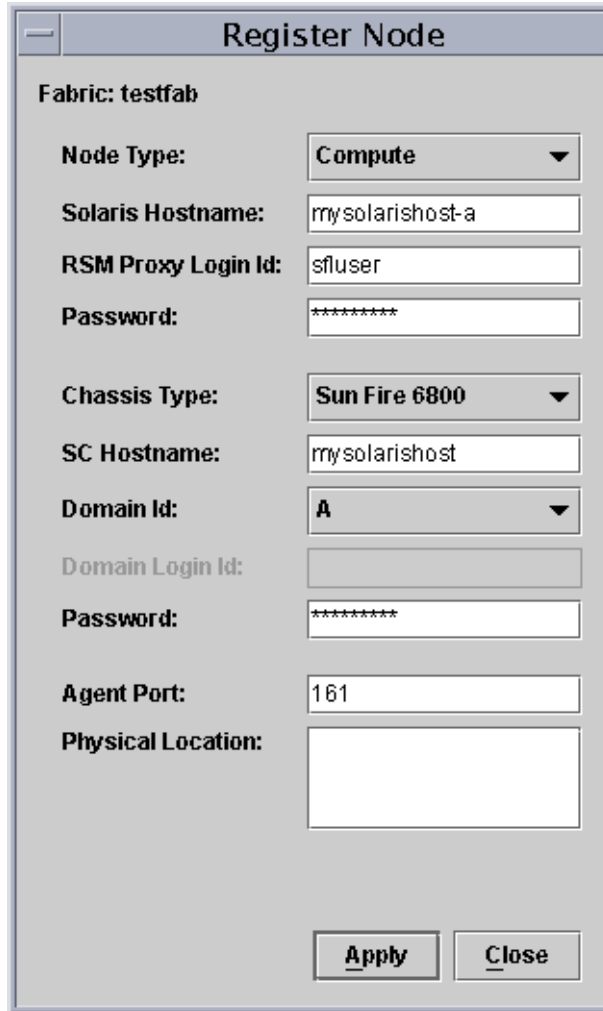
FIGURE 5-4 Fabric Menu Showing Register Nodes Menu Item

The Register Nodes dialog box appears.

3. **Fill in the fields of the Register Nodes dialog box and click on Apply when finished.**

In FIGURE 5-5, a node is being registered instead of a switch. A switch example is presented at a later point in the procedure.

FIGURE 5-5   Register Nodes Window With Fields for Registering a Compute Node

The various fields are explained below:

Node Type	To register a node, ensure that Compute is selected.
Solaris Hostname	Specify the host name of the chassis, followed by a dash and the domain ID of the domain connected to the Sun Fire Link network.
RSM Proxy Login ID	Enter your WRSM proxy username (see "To Set Up WRSM Proxy Security on Cluster Nodes" on page 12").
Password	Enter your WRSM proxy password (see "To Set Up WRSM Proxy Security on Cluster Nodes" on page 12").
Chassis Type	Select the type of system containing the node that is being registered.
SC Hostname	Enter the host name of the chassis.
Domain Login ID	This is the login name you would use for a Sun Fire 15K/12K domain. This Domain Login ID must be in the dmn*X*admin group on the System Controller and in the sfladmin group on the domain. This field is not used on Sun Fire 6800 systems.
Password	Enter the domain password. This is the password for the Domain Login ID required by Sun Fire 15K/12K domains (see "To Create a Password for Domain Consoles" on page 13").
Agent Port	This is the standard agent port for the Sun Management Center agent running on the node. Use the default value.
Physical Location	This is an optional field. You can use it to describe the physical location of the node or switch.

4. **Click on Apply when you have finished entering the required information.**

5. **Repeat Step 3 and Step 4 for each node that you want to be a member of the fabric.**

6. **If you want to register Sun Fire Link switches in the fabric, change the Node Type to Sun Fire Link Switch.**

   This causes the Register dialog box to display a slightly different set of fields, as shown in FIGURE 5-6.

FIGURE 5-6   Register Nodes Window With Fields for Registering a Switch Node

7. Fill in the fields of the Register dialog box as they appear for registering a switch.

- SC Hostname – Enter the host name of the System Controller.

The dialog box for registering switches is essentially the same as for registering nodes except the following are disabled because the Solaris operating environment is not running on the switches:

- Solaris Hostname
- RSM Proxy Login Id
- Password

■ Domain ID

8. **Click on Apply when you have finished entering the required switch information.**

9. **Repeat Step 6 and Step 7 for each switch that you want to be a member of the fabric.**

▼ To Verify That the Nodes Were Successfully Registered in the Fabric

Perform the following steps to verify that the nodes and switches are now members of the fabric. Begin in the Nodes & Routes perspective setting, which is the default.

1. **Open the Fabric Details window and select the Interconnect tab (the default selection).**

2. **Double-click on the Available Nodes folder.**

   a. **Verify that the list of nodes displayed below the opened folder is correct.**

   b. **Verify that icons for all the registered nodes are present in the topology panel.**

3. **In either the tree or topology panel, double-click each node icon.**

   This displays properties information about the selected node. Verify that the node information is correct.

4. **Perform Step 1 through Step 3 on the Available Switches folder.**

5. **Change the Perspective selection to Chassis & Links.**

6. **Double-click the Available Chassis icon to expand the folder and show the topology panel.**

   Verify that the list of nodes displayed below the folder is correct.

7. **Verify that the registered nodes are represented in the topology panel.**

8. **In either the tree or topology panel, double-click each chassis icon.**

   Verify that the displayed chassis information is correct.

9. **Pull down the Fabric menu and click on Node Registry.**

   Verify that all information about the nodes is correct.

10. **In the Fabric Details window, select the Browser tab.**

11. **Double-click on Sun Fire Link FM Configuration.**

12. Double-click on Fabric Information.

Verify that the Member Table contains the registered nodes.

Creating and Configuring Partitions

This chapter provides instructions for creating partitions and managing their configurations.

You can create and configure a partition in either of two ways:

- Use an XML-based configuration file. This file can be developed by editing a template file. Or, if a suitable configuration file already exists, you can import it as is or edit the file to meet the current partition configuration requirements.

- Let the FM software produce an optimal configuration automatically. You simply specify which nodes you want to be in the partition and the desired striping level. The FM then surveys the links that are available to the specified nodes and configures the optimal set of connections for the partition.

This chapter described the automatic configuration by the FM method. Manual creation of an XML configuration file is described in Appendix B.

Choosing Partition Characteristics

This section discusses choices you need to make regarding certain characteristics of the partition.

Partition Topology

When you create a new partition, you are asked to specify whether the partition includes switches. Your input choices are:

- Direct connect
- Sun Fire Link switches

Link Striping Level Policy

You are also asked to specify the level of striping to be used in the partition. When you do, keep the following in mind:

- There are two types of striping in a Sun Fire Link cluster, *link striping* and *ASIC striping* (also called WCI striping).

  - Link striping consists of dividing the data in a given transfer operation between the two optical ports on a single Sun Fire Link assembly. By itself, link striping provides level 2 striping.

  - ASIC striping consists of dividing the data in a given transfer operation between the two Sun Fire Link assemblies in a node. By itself, ASIC striping provides level 2 striping.

  - If link and ASIC striping are both used, the result is referred to as level 4 striping.

- If the partition you are creating will be used for Sun HPC ClusterTools applications, specify level 4 striping. While this level is not required, it will provide maximum bandwidth for data transfers between nodes. It also allows operations to continue at a reduced bandwidth if one or more links on a node should fail.

- If the partition will be used for Sun Cluster applications, specify level 2 striping. The FM will automatically implement link striping but not ASIC striping. This will allow the two Sun Fire Link assemblies in each node to be configured in a dual-controller fashion. This dual-controller mode is required to support Sun Cluster availability requirements. The single- and dual-controller concept is discussed below.

- In a three-node, direct-connect configuration, level 2 striping is the maximum possible because of the limited cabling topologies available.

Note – This description of striping choices is based on the assumption that each domain contains two Sun Fire Link assemblies and that each assembly has two operational link ports. Reducing either the number of assemblies or the number of available link ports will restrict the available striping levels accordingly.

Single- vs. Dual-Controller Domains

In this context, the term *controller* refers to a logical entity that is created by the RSM drivers to manage the optical links local to the node. In a Sun HPC ClusterTools environment, each node will have a single RSM controller. In a Sun Cluster environment, each node will have two RSM controllers.

This section describes two procedures for configuring Sun Fire Link clusters—one procedure for configuring single-controller clusters and the other for configuring dual-controller clusters. The key features of these two configuration types are explained below.

Single-Controller Summary

Single-controller clusters are used for Sun HPC ClusterTools applications—that is, clusters that have Sun HPC ClusterTools software running on the nodes.

In this type of configuration, the RSM™ software in each domain maintains a single logical controller to manage both Sun Fire Link ASICs used by that domain. This means each domain has all four optical ports available for sending and receiving messages across the Sun Fire Link network. In other words, a single-controller configuration supports four-way striping (two-way WCI striping plus two-way link striping). FIGURE 6-1 and FIGURE 6-2 show examples of a single-controller cluster.

Note – The term *WCI*, which appears in some Sun Fire Link output messages and in some Sun Management Center/FM console icon labels, is equivalent to the Sun Fire Link ASIC.
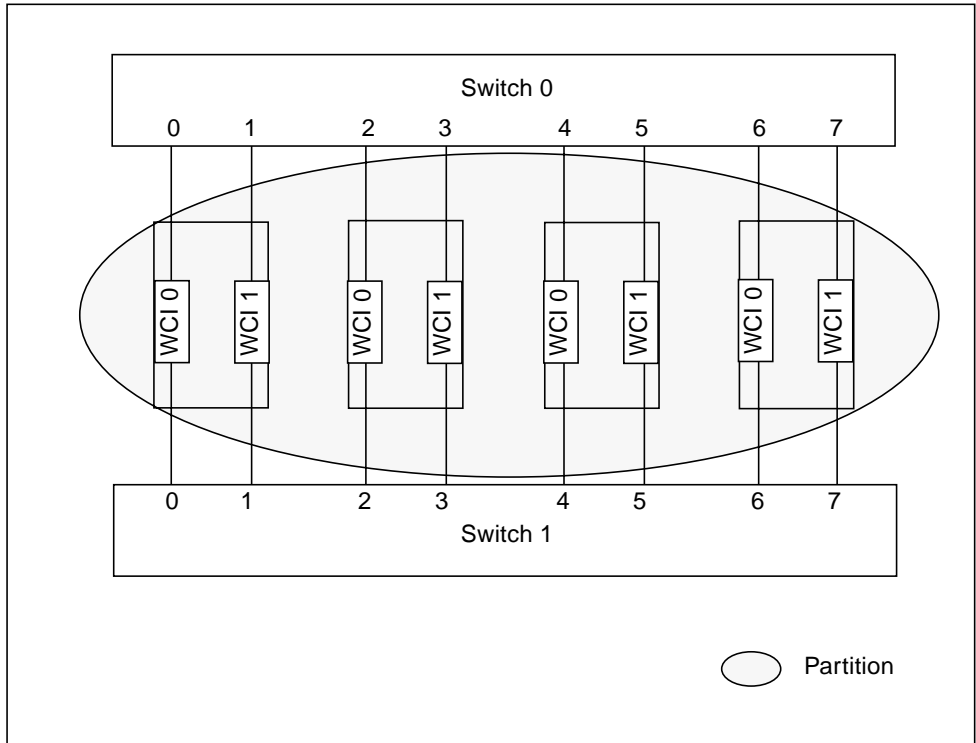
FIGURE 6-1 Single-Controller Example – Four Nodes, Two Switches, One RSM Controller per Node

Switch 2

0   1   6   7

0   1   Switch 0   6   7

WCI 0   WCI 1   WCI 0   WCI 1   · · ·   WCI 0   WCI 1   WCI 0   WCI 1

0   1   Switch 1   6   7

0   1   Switch 3   6   7

Partition
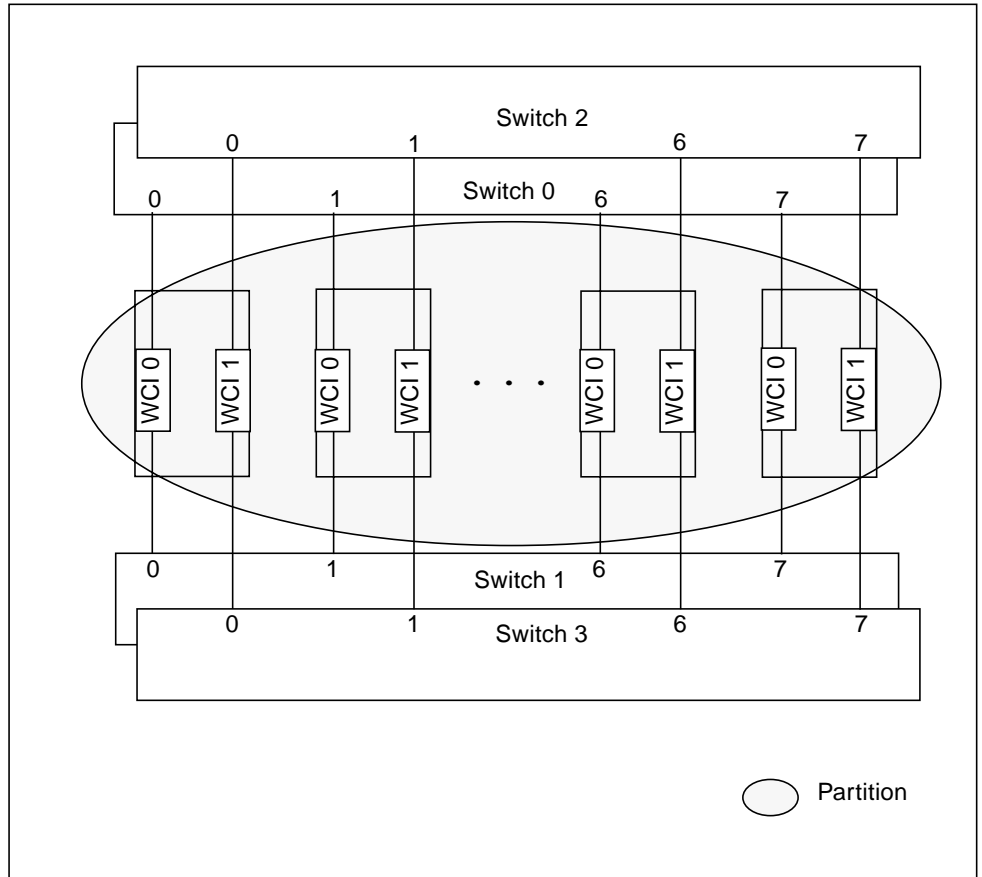
FIGURE 6-2   Single-Controller Example – Eight Nodes, Four Switches, One RSM Controller per Node

Dual-Controller Summary

Dual-controller clusters are used for Sun Cluster applications in which each domain must have Sun Fire Link ASIC failover support.

To achieve this, each domain's two Sun Fire Link ASICs are mapped into separate partitions, and separate RSM controllers are allocated to manage the two partitions. One ASIC and the two optical ports it controls function as the primary network interface. The other ASIC is brought into use if the primary network interface should fail. FIGURE 6-3 and FIGURE 6-4 show examples of dual-controller cluster configurations.
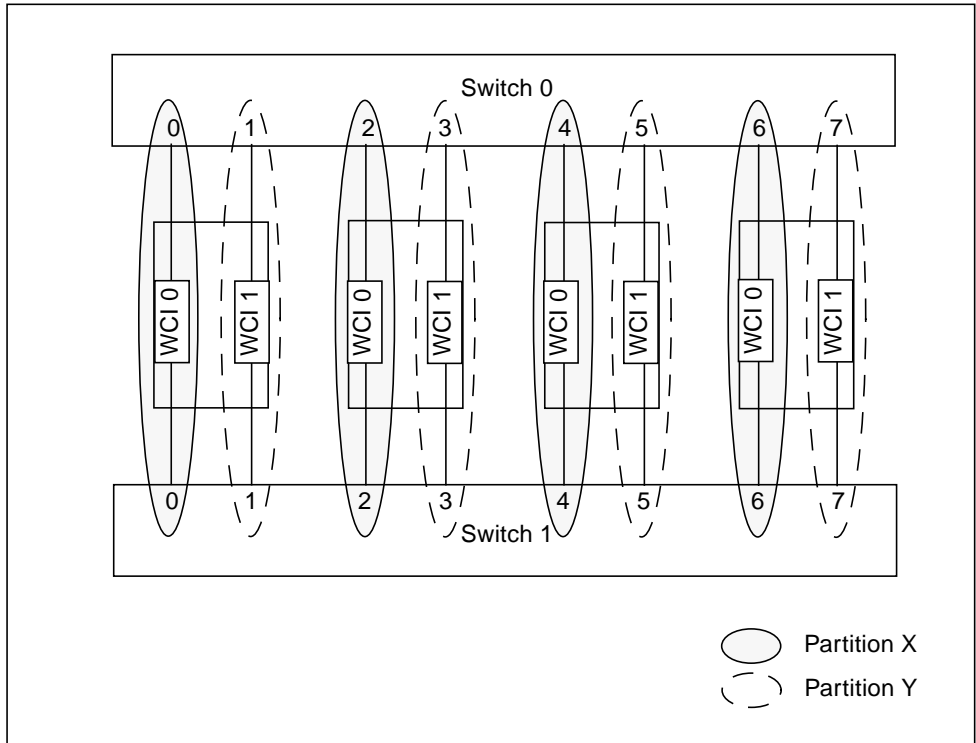
FIGURE 6-3 Dual-Controller Example – Four Nodes, Two Switches, Two RSM Controllers per Node
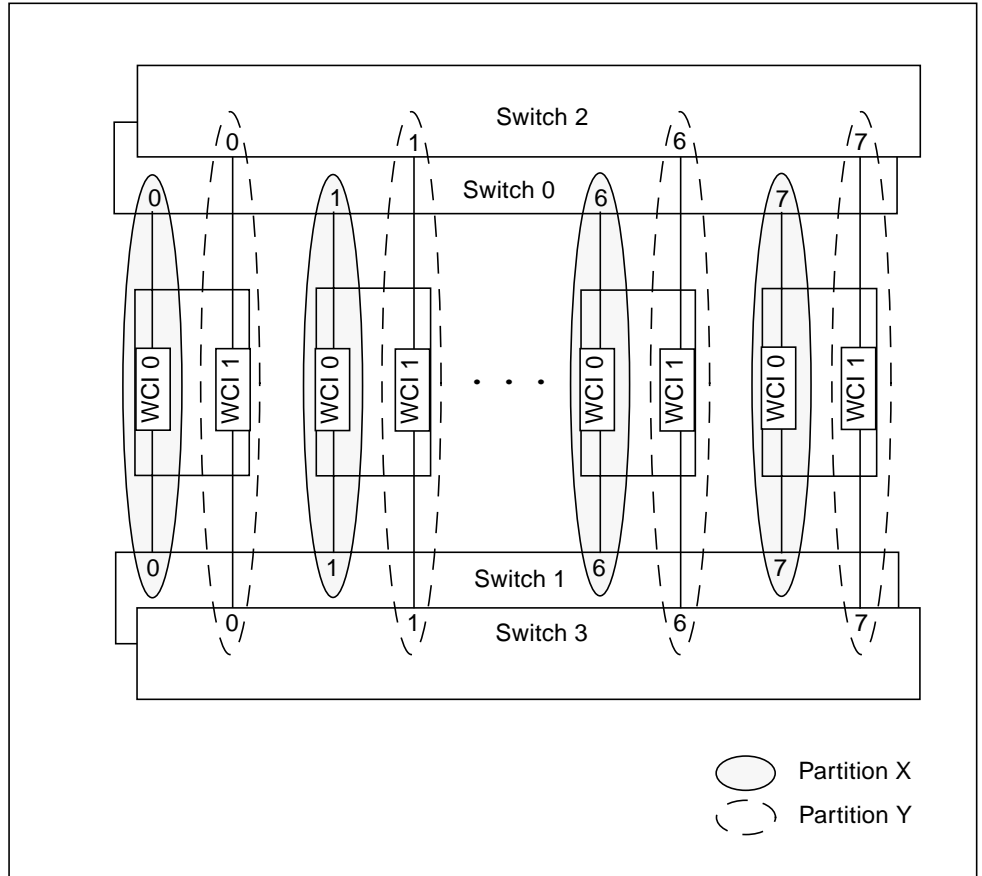
FIGURE 6-4   Dual-Controller Example – Eight Nodes, Four Switches, Two RSM Controllers per Node

Creating Partitions

When you create a partition, the main task you perform is assigning cluster nodes and switches (if used) to the partition. The level of analysis you need to apply to the task will differ, depending on whether you are configuring a single-controller or a dual-controller cluster.

In either single- or dual-controller mode, you select the topology (direct-connect or switched), the striping policy (Level 1, Level 2, or Level 4) and which nodes are to be members of the partition.

In a single-controller Sun HPC ClusterTools environment, the node and switch selection process is straightforward. You simply select the nodes and switches of interest and add them to the partition. The FM will then calculate appropriate routes that connect all the selected nodes using the specified striping level.

For dual-controller Sun Cluster environment however, you must also consider the following:

- A dual-controller configuration requires two partitions, one for each RSM controller.
- The two Sun Fire Link assemblies in each node must be divided between the two partitions, as shown in FIGURE 6-3 and FIGURE 6-4.
- If your cluster contains two switches, both switches must be shared by the two partitions. That is, each switch must be assigned to both partitions. See FIGURE 6-3.
- If your cluster contains four switches, two of the switches must be added to one partition and the other two must be in the other partition. See FIGURE 6-4.

When making your node and switch selections, use the cable routing worksheet that was created when the Sun Fire Link hardware was installed as a guide. See the *Sun Fire Link Hardware Installation Guide* for details. If you do not have a cabling worksheet, you should complete one before creating and configuring any Sun Fire Link partitions.

Note – When you pair Sun Fire Link assemblies to achieve ASIC striping in a Sun Fire 15K/12K system, the two assemblies must occupy adjacent, even-odd I/O slots. For example, it is legal to pair the assemblies in I/O slots 8 and 9, but not in I/O slots 7 and 8.

▼ To Create a Partition

1. **Open the Fabric Details window, pull down the Fabric menu, and select Create Partition. See** FIGURE 6-5 **for an illustration of this menu.**
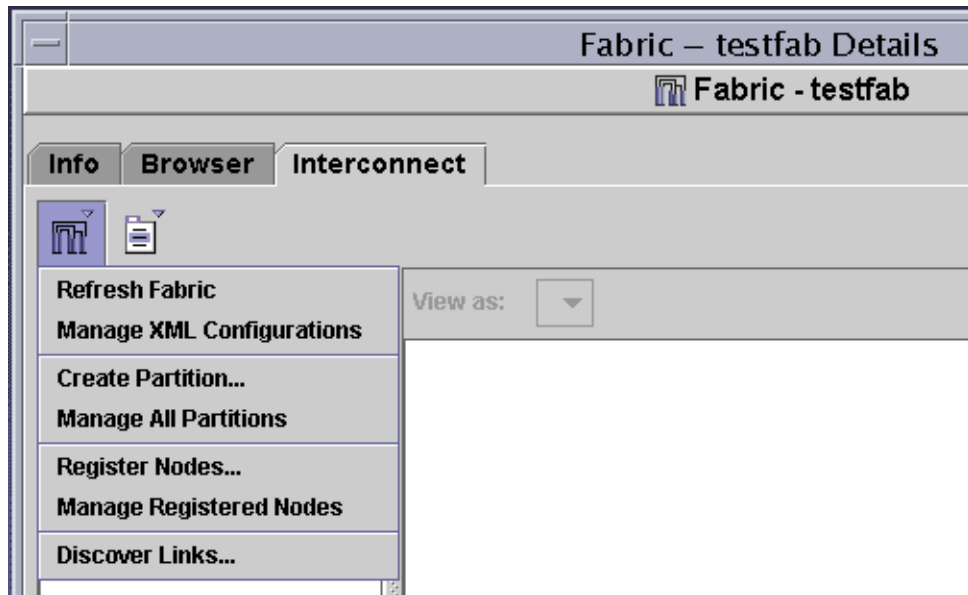
FIGURE 6-5   Fabric Menu Showing Create Partition Menu Item

A small window labeled Create Partition appears.

2. Fill in the Create Partition fields and click OK when finished. See FIGURE 6-6.
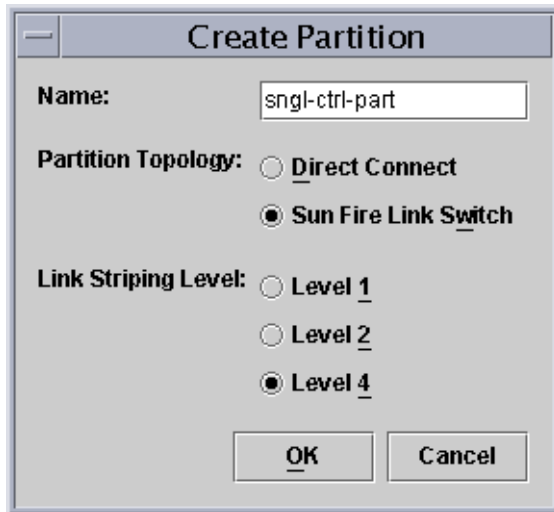
FIGURE 6-6   Create Partition Dialog

The Create Partition fields are explained below:

Name	Assign a unique name to the partition.
Partition Topology	Specify that the partition topology is to be either direct connect or through Sun Fire Link switches.
Link Striping Level Policy	For single-controller configurations (Sun HPC ClusterTools applications), specify Level 4 for maximum bandwidth. For dual-controller (Sun Cluster applications), specify Level 2 for maximum bandwidth and availability.

An icon for the new partition appears in the tree panel of the Fabric Details window. If the partition has a direct-connect topology, it appears beneath the Available Nodes heading. If it is a switch-connect partition, it appears beneath the Available Switches heading.

Adding Nodes and Switches to a Partition

You use the same Sun Management Center/FM dialog box to add both nodes and switches. The specific steps you take within that dialog box depend on whether the partition is to have a single-controller or a dual-controller configuration. Separate procedures are described for each.

▼ To Add Nodes and Switches to a Partition in a Single-Controller Configuration

1. **Right-click on the partition icon in the tree panel and select Add Node from the pop-up menu.**

   This displays the Add Node dialog box (see FIGURE 6-7).
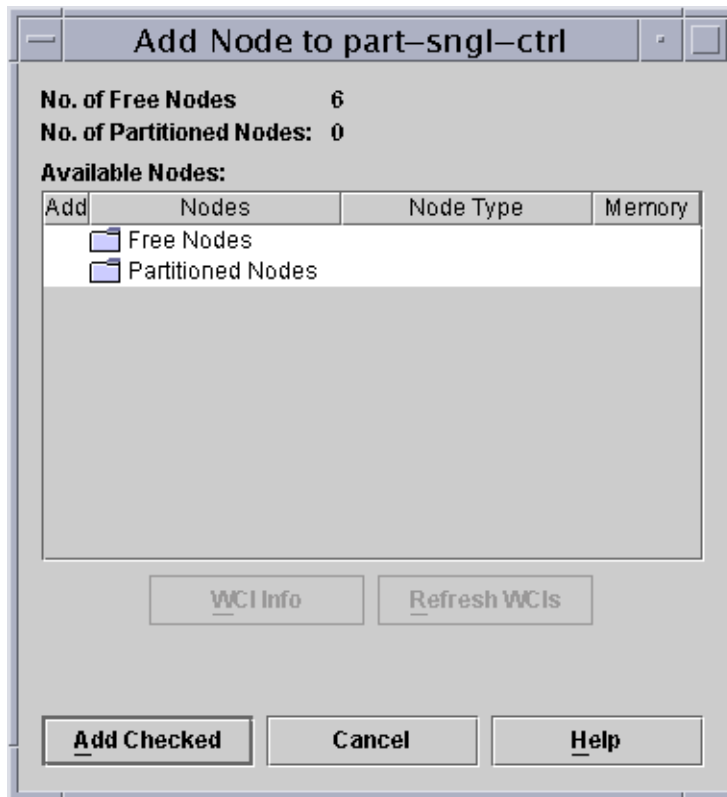
FIGURE 6-7 Add Node Dialog

2. Expand the Free Nodes folder.

This displays all the nodes that are not already members of a partition (see FIGURE 6-8).
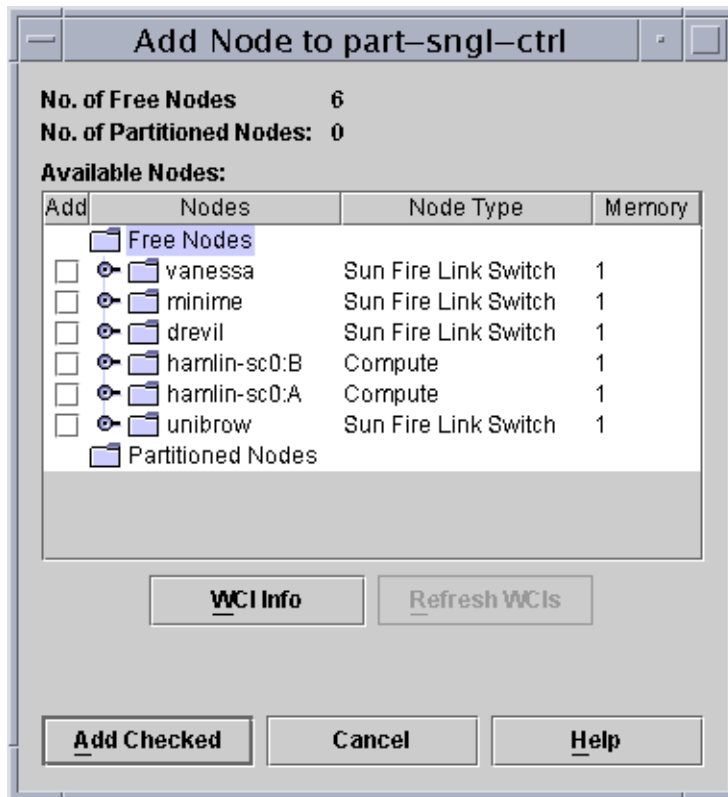
FIGURE 6-8 Add Node Dialog With Expanded Free Nodes Folder

3. **Click on the boxes to the left of every node that you want to include in the partition.**

4. **When you have checked the boxes of all the nodes and switches that you want to add to the partition, click on Add Checked.**

An example is shown inFIGURE 6-9.
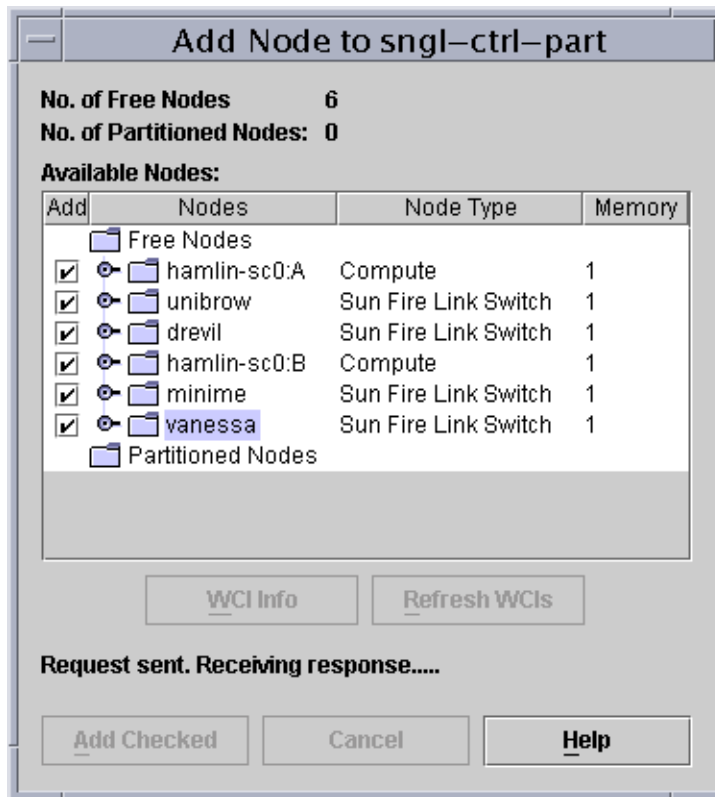
FIGURE 6-9   Add Node Dialog for Single Controller Partition With Nodes and Switches
Selected

You can now verify the partition as described in "Verifying Partition Contents" on
page 57.

▼ To Add Nodes and Switches to a Partition in a Dual-Controller Configuration

1. Create one or two partitions.

If you have already created a partition as described in "To Create a Partition" on
page 48, repeat those steps to create a second partition. If not, perform that
procedure twice. Each partition name must be unique.

2. **Right-click on one of the partition icons in the tree panel and select Add Nodes from the pop-up menu.**

This displays the Add Node dialog box.

3. **Expand the Free Nodes folder.**

This displays a list of all the nodes and switches that are not already members of a partition. Each node and switch appears as a folder whose label is the host name of the node or switch.

Note – Although the free nodes list may appear empty (as shown in the next example), Sun Cluster configurations employ only one Sun Fire Assembly per node. Since each compute node holds two assemblies, you can assign the second assembly of each compute node to a second partition.

---

4. **Expand the folder for each node that you want to include in the partition.**

For nodes, this displays icons for the Sun Fire Link assemblies in that node. This will display the Sun Fire Link assemblies (WCIs) contained in each node. These will be labeled with a slot ID (S*n*) and a WCI ID (W*n*).

5. **For nodes, click on the box to the left of every Sun Fire Link assembly that you want to include in the partition. See** FIGURE 6-10**.**

Remember that only one Sun Fire Link assembly (WCI) per node can be in the partition. Verify your choices by checking the cabling worksheet.
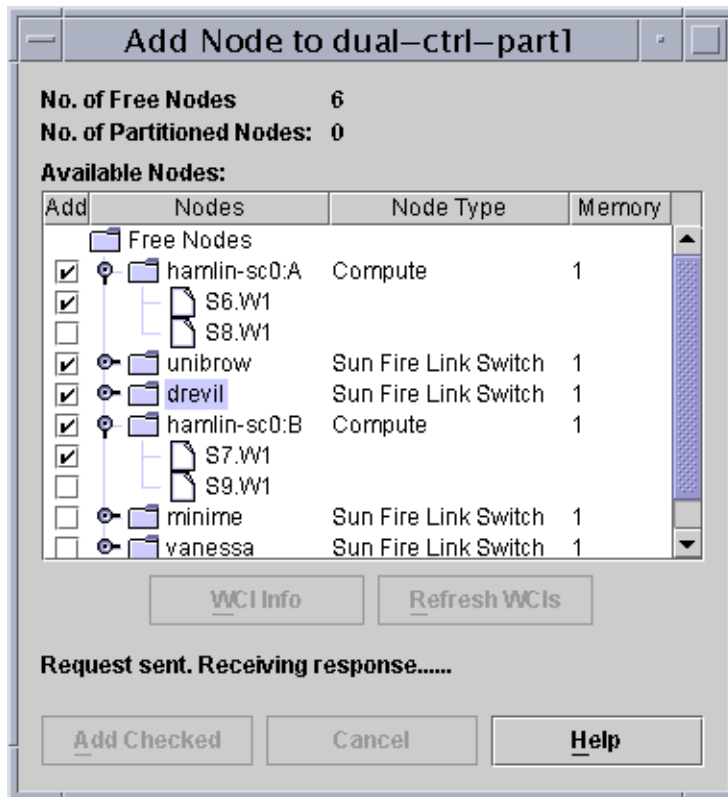
FIGURE 6-10 Add Node Dialog for a Dual-Controller Partition on a Sun Fire 6800 System With Nodes and Switches Selected

6. **For the switches, click on the box next to each switch that you want to include in the partition.**

   Remember the switch selection rules for dual-controller configurations described in "Creating Partitions" on page 47. The key points are repeated below:

   - If your cluster contains two switches, both switches must be shared by the two partitions. That is, each switch must be assigned to both partitions. See the example in FIGURE 6-3.

   - If your cluster contains four switches, two of the switches must be added to one partition and the other two must be in the other partition. See the example in FIGURE 6-4.

7. **When you have checked the boxes of all the nodes and switches that you want to add to the partition, click on Add Checked.**

8. **Reopen the Add Node dialog box for the second partition.**

   Because some of the fabric's nodes and switches were added to the first partition in Step 2 through Step 7, they will now be in the Partitioned Nodes folder instead of the Free Nodes folder.

9. **Expand the Partitioned Nodes folder.**

   This will display the nodes and switches that were added to the first partition. It may also have some other nodes and switches that were added to other partitions in different sessions.

10. **Expand the folder for each node that you want to include in the second partition.**

    Note that the selection boxes will be empty at first, so that it appears that both Sun Fire Link assemblies are available in the nodes. Ignore the fact that they are empty.

11. **Click on the nodes that you want to add to the second partition.**

    This *must* be that same set of nodes that you selected in Step 5. The FM will automatically check the boxes for the Sun Fire Link assemblies that are actually available.

12. **Verify that the correct Sun Fire Link assembly boxes were checked.**

13. **Click on the switches that you want to add to the second partition.**

    Follow the switch selection rules listed in Step 6.

    You can now verify the partition as described in "Verifying Partition Contents" on page 57.

Verifying Partition Contents

You can use the following procedures to verify that the partitions you created and configured have the correct contents and topology characteristics.

▼ To Verify a Partition's Contents and Topology

1. **From the Interconnect tab of the Fabric Details window, open the Manage Partition dialog box.**

   Pull down the Fabric menu and select Manage Partition. Verify that the partition information is correct.

2. **Open the Edit Partition Properties window.**

   Select the partition in the tree and the select Properties of *partition-name* the Edit pull-down menu. The following window appears:

   Verify that all the information about the partition is correct.

3. **Verify the information in the Partition Table.**

   a. **Select the Browser tab of the Fabric Details window.**

   b. **Double-click the WCI Fabric Manager Configuration icon in the window.**

      The view expands.

   c. **Double-click Fabric Information.**

      The Partition Table is displayed.Verify that the partition appears in the table and that the information about the partition is correct.

Deleting Nodes, Switches, Partitions, and Fabrics

The following rules govern the deletion of nodes, switches, partitions, and fabrics:

- All nodes must be deleted from a partition before any switches can be deleted.
- Switches that are not in partitions can be deleted even if there are undeleted nodes in the fabric.
- A partition must be empty before it can be deleted.
- A fabric must be empty before it can be deleted.

These deletion procedures are described below.

▼ To Delete Individual Nodes and Switches

Use the following procedure to delete individual nodes. Any switch that is not assigned to a partition can be deleted as well.

1. **In the Sun Management Center console Double-click the Fabric icon in the main Sun Management Center window.**

   The Fabric Details window appears.

2. **Select the node or switch you want to delete.**

3. **Pull down the Edit menu and select Delete** *node_name.*

▼ To Delete Nodes and Switches From a Partition

1. **In the Sun Management Center console select the Nodes and Routes perspective and open the Fabric Details window.**

   Double-click the fabric icon in the main Sun Management Center window.

   The Fabric Details window appears.

2. **Select the partition from which you want to remove a node or switch.**

3. **Launch the RSM Manage Partition dialog box.**

   Pull down the Fabric menu and select Manage Partition.

4. **In the Manage Partition dialog box, select the nodes and switches you want to delete. See** FIGURE 7-1.

   If you are planning to delete the partition, select all the nodes and switches so the partition will be empty.
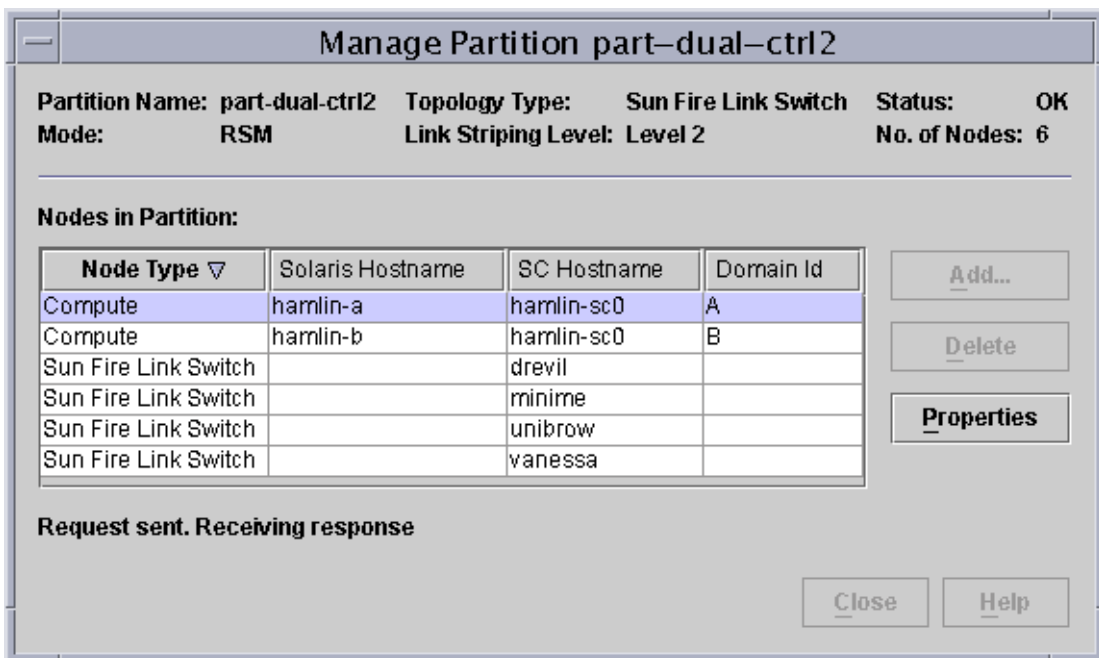
FIGURE 7-1   Manage Partition Dialog Box With List of Nodes in Selected Partition

5. **Click Delete and close the dialog box.**

   If you deleted all the nodes and switches from the partition you can now proceed to the procedure described in "To Delete a Partition" on page 61.

▼ To Delete a Partition

The following procedure is based on the assumption that the partition is empty.

1. **In the Fabric Details window, right-click the partition icon.**

   A pull-down menu appears.

2. **Select the Delete command.**

   A small window with a warning appears.

3. **Select OK to confirm.**

   The partition is deleted from the fabric, and its icon disappears from the tree panel of the Fabric Details window.

▼ To Delete a Fabric

1. **If you want to delete a fabric, perform the following procedures for all nodes, switches, and partitions in the fabric:**

   - "To Delete Individual Nodes and Switches" on page 59
   - "To Delete Nodes and Switches From a Partition" on page 60
   - "To Delete a Partition" on page 61

2. **In the Sun Management Center console, select the icon for the fabric you want to delete.**

3. **Pull down the Edit menu and select Delete.**

Discovering Links

The Sun Fire Link Manager provides a link discovery service that performs a survey of links in:

- A specified fabric.
- A specified partition.
- A specified subset of nodes and/or switches in a specified partition.
- One or more specified optical ports within specified nodes and/or switches.

When the survey is complete, the FM displays a list of the links it finds within the scope of the discovery operation. The list identifies the physical location of local and remote endpoints for each discovered link. This link discovery feature provides a simple way to evaluate the health of the connections between nodes in a partition.

The steps for executing the discover links procedure are described below.

▼ To Discover Links

1. **Ensure that:**
   - You have the Sun Management Center console open.
   - A fabric and partition have been configured.

2. **Double-click on the fabric to open the Fabric Details window.**

3. **Select the partition whose links you want to discover.**

   In the left window pane of the Fabric Details window select the icon that corresponds to the partition.

4. **Select the Fabric --> Discover Links command. See** FIGURE 8-1**.**
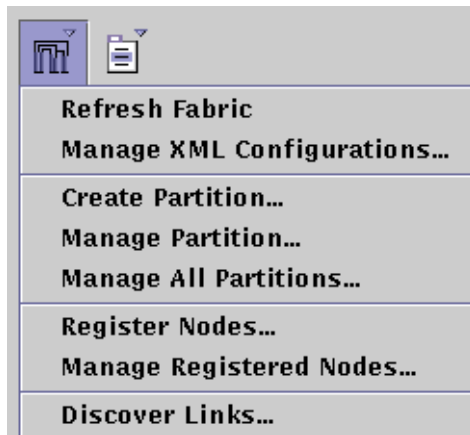
FIGURE 8-1   Fabric Menu Showing Discover Links Choice

A Link Discovery window appears. The window contains one folder for each node.

5. **Specify the scope of the link discovery that you want to have performed.**

You specify the scope of the link discovery by clicking on the box next to each component for which you want link information.

- To acquire link status at the chassis level, click on the box next to each chassis folder you are interested in.

- To acquire link status at the domain level, expand the chassis folder containing the domain of interest. See FIGURE 8-2.

- To acquire link status for an individual optical port, expand the folder for the Sun Fire Link assembly of interest.
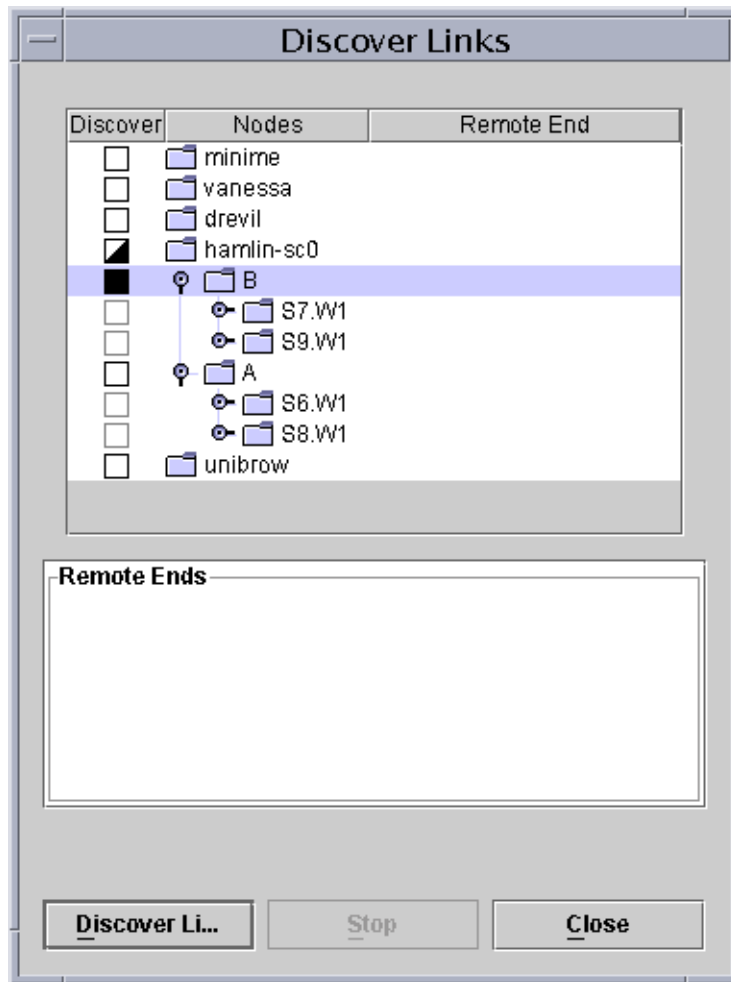
FIGURE 8-2   Discover Links Dialog Showing an Expanded Chassis Folder

6. When all the boxes of interest are selected, press the Discover Links command.

While the Sun Management Center software is discovering the links discovered, a progress message appears beneath the Discover Links command. When the links are discovered, they appear in the Remote Ends pane. See FIGURE 8-3.
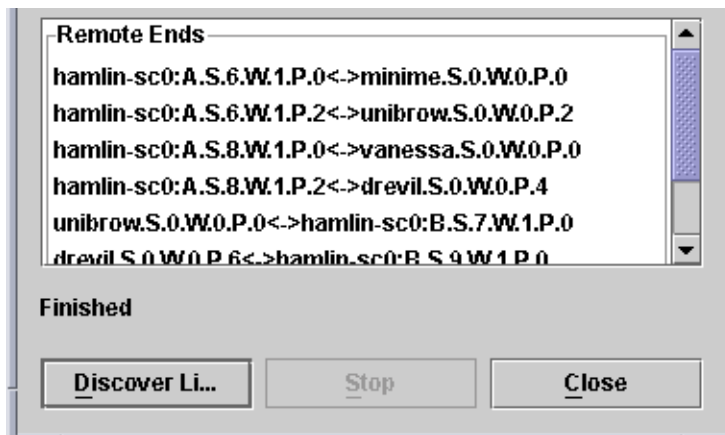
FIGURE 8-3   Remote Ends Panel

Note – Only links that are allocated to a partition will be listed. Links that are not used will not be displayed.

The following explains how to interpret the link discovery report:

■ The report contains a list of links that were discovered within the fabric. Each line describes a single link, which is a connection between two nodes in the fabric.

■ The nodes that are the endpoints of each link are separated by a double-ended arrow, `<->`.

  If the partition includes switches, one endpoint will be a compute node and the other will be a switch node. If it is a direct-connect configuration, both endpoints in each link will be a compute node.

■ Each endpoint completely describes its physical location. The descriptions consist of the following fields:

  ■ The node name. If the node is a Sun Fire Link switch, this it the name of the switch. If the node is a compute node, this is the name of the chassis that contains the node.

  ■ The domain name, if the node is a compute node. If the node is a Sun Fire Link switch, this field is omitted.

  ■ The chassis slot number. This is the physical slot within the chassis.

  ■ The Sun Fire Link ASIC address.

  ■ The optical port number. For a switch node, this number can range from 0 - 7. For a compute node, this number will be either 0 or 2.

In the main Fabric Details window, both the Nodes & Routes, and the Chassis & Links windows are updated to show the newly discovered links.

Monitoring Sun Fire Link Fabrics

Sun Fire Link Software provides information about the configuration of your fabric in properties panels. This chapter shows you how to access and interpret the properties panels.

The Sun Management Center software also provides current status information about the components of your fabric. This information appears in Fabric Details panels. This chapter shows you how to access and interpret the Details panels that convey information about fabric components.

You can gain much useful information by revisiting the dialog boxes used in configuring the fabric and its partitions.

Sun Management Center and Sun Fire Link Manager software provide even more extensive information about your fabrics from several sources.

- The Fabric Console displays:
  - Configuration data
  - Status data
  - Graphic depictions of network nodes routes; hardware chassis and links
  - Alarms
- Node Details panels display:
  - Fabric software status data
  - RSM network status data
  - Alarms
- Switch Details panels display:
  - Switch status data

Checking Fabric Information by Reviewing Configuration Dialog Boxes

You can review certain kinds of information by reopening the dialog boxes you used when configuring the fabric. For example,

- In the Add Node dialog box, you can view the following:
  - Number of free nodes in the fabric
  - Number of partitioned nodes in the fabric
  - Node types (compute or switch)

- In the WCI Info dialog box, you can view the following:
  - Nodes in your fabric, with slot, interface, and Paroli information
  - Identical information for the remote systems linked to each Paroli

- In the Managing Partitions dialog box, you can view the following:
  - Fabric name
  - Number of nodes in the fabric
  - Number of partitions in the fabric
  - Number of links in the fabric
  - Name, mode, topology type (direct connect or switch), link striping, and route status (Ok or Degraded) of each partition in the fabric

- In the Manage Registered Nodes dialog box, you can view the following:
  - Fabric name
  - For each node registered in the fabric, you can view the SC host, domain ID, Solaris hostname, chassis type, node type, and mode

- In the Manage Partition dialog box, you can view the following:
  - Partition name, topology type (direct connect or switch), Mode (RSM), route status (Ok or Degraded), and number of nodes in the partition.
  - For each node, you can view the link state, SC hostname, domain ID, chassis type, node type, and Solaris hostname.

- In the XML Configuration Manager dialog box, you can view the XML configuration for your fabric.

Note – Status and error information is also supplied in log files.

Displaying the Fabric Console

The Sun Management Center software gathers status data about Sun Fire Link hardware and software components. You can use various Sun Management Center *refresh* commands to request updates to the status data.

The Sun Management Center console displays a Details window when you double-click a fabric icon in the Default Domain window. The Fabric Details window includes menus of Sun Fire Link software commands. Taken as a whole, the Fabric Details window is also known as the *FM console.* An example of the FM console is shown in FIGURE 9-1
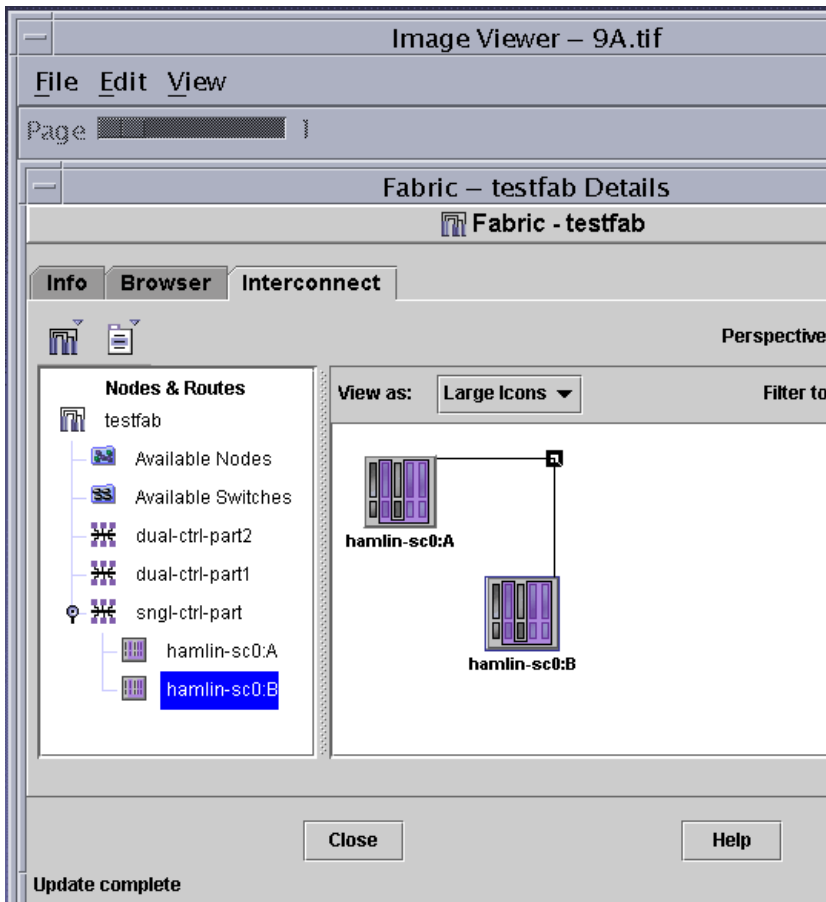
FIGURE 9-1 Sun Fire Link Fabric Console

Perspective Menu

There are two different views for each partition's topology depending on the
Perspective settings.

FIGURE 9-2 Perspective Menu

Use the Perspective pull-down menu to select one of the two perspective views of the fabric:

- Nodes & Routes view—This view shows the data routes between Sun Fire Link nodes. Each route may include multiple cable links. This is the default view.
- Chassis & Links view—This view shows the individual links (cables) that connect chassis-based nodes and switches (if present).

Tree Panel

The fabric console displays fabric components in a tree hierarchy:
Fabric -> Partitions -> Nodes (or Chassis). In addition, folders of available nodes and available switches (if present) are also displayed under the fabric tree root symbol.

Tree elements are:

- Fabric – A list of all components of the fabric. No alarm icons are displayed at this level.
- Available Nodes – A list of all registered nodes in the fabric that are available to be attached to a partition (that is, they are not yet assigned to a partition). No alarm icons are displayed at this level.
- Available Chassis – A list of all systems in the fabric that contain nodes that are available to be attached to a partition (that is, they are not yet assigned to a partition). Alarm icons are displayed at the level of the individual chassis, and to the nodes within the chassis.
- Available Switches – A list of all registered switches in the fabric that are available to be attached to a partition (that is, they are not yet assigned to a partition). No alarm icons are displayed at this level.
- Partitions – A cluster of fabric nodes and switches. When a node or switch has been allocated to a partition, it no longer appears in the *available* folders. Alarm icons are displayed at this level, and to the nodes belonging to the partition.

Note – Switches do not display alarm icons in this release.

Topology Panel

The topology panel occupies the right panel in the split pane, giving a topographical representation of the fabric. You can double-click on any item in the panel to bring up its associated information dialog box. You can select commands from the Edit menu or right-click an item to perform actions on a selected item. If an alarm occurs, the item associated with the alarm is indicated in the topology panel. You can filter the alarms to show all, degraded, or failed. You can also select from a pull-down menu to view large or small icons.

The topology panel displays high-level views of the health of partitions in Sun Fire Link networks. This information can provide a starting point for investigating problems.

Double-clicking on a chassis or a node in either panel brings up the associated Details dialog box.

Viewing Partitions Using Nodes and Routes Perspective

You can display the components the Topology panel with large icons or small icons (as shown in FIGURE 9-1 and FIGURE 9-3, respectively).
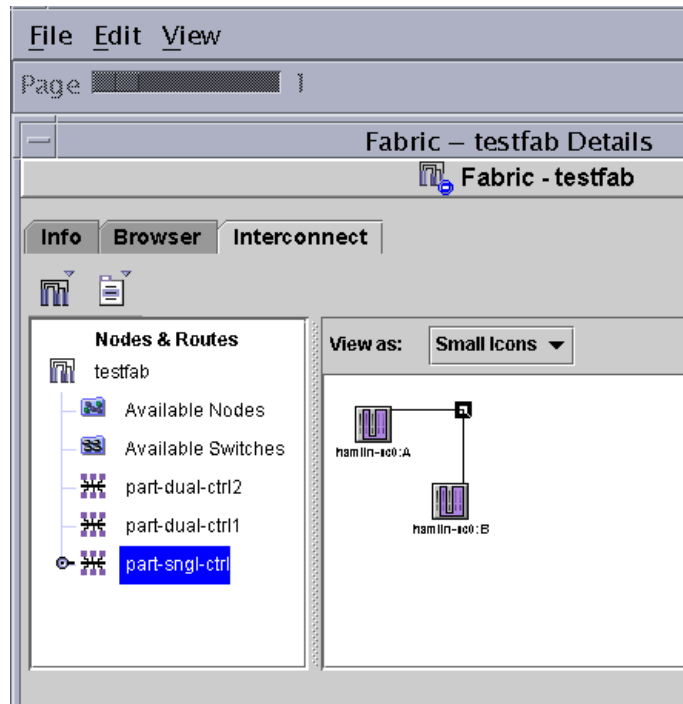
FIGURE 9-3   Two-Node Partition (Small Icons)

The Nodes and Routes view shows the data routes between nodes in a partition. These routes can contain one or more physical links (cables) with detailed route and link information accessed by double- or right-clicking the route. Route status is shown by the color of the route and by an icon indicating the following conditions:

■ OK – No links in the route have a Severity greater than OK. The route is colored black. A standard route access point (black square) is shown on the route.

■ Degraded – One or more links in the route has a Severity greater than OK, or the route does not contain enough links to meet the partition's striping requirements. The route is colored yellow. A standard Sun Management Center Warning icon replaces the route access point.

■ Failed – All links in the route have a Severity greater than OK, or the route does not contain any links at all (that is, the node is unreachable). The route is colored red. A standard Sun Management Center ERROR icon replaces the route access point.

The Nodes and Routes view displays these items:

■ Sun Fire Link nodes – Shown by white-centered node icons. Each node is labelled with partition name, node ID, and domain ID.

- Routes – Shown by solid lines connecting nodes.
- Route Access Points – Shown by squares at the corners of the route to aid in selection.
- Alarm Icons – Shown by standard Sun Management Center error icons at the corners of the routes to aid in visualization and selection.

Viewing Data Route Properties

Double-click on a route in the Nodes and Routes view to see data about the route between two nodes.

The Route Properties dialog box can help you to find out what failure is causing the route to be marked degraded or failed. FIGURE 9-4 shows the route properties for a set of links that are all healthy.
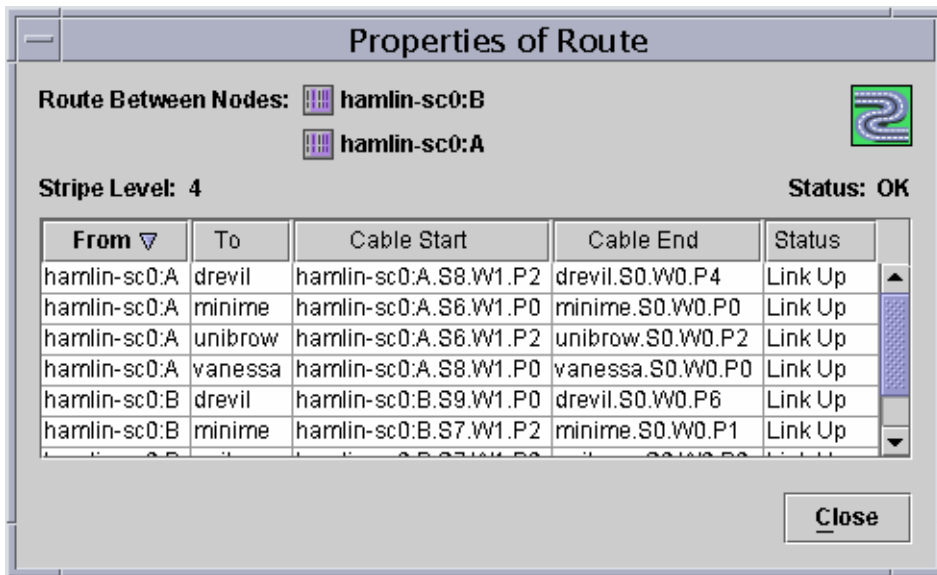
FIGURE 9-4   Route Properties Dialog Box

Route information includes:
- Nodes in this route – Each node is identified by its SC host name and domain ID.
- Stripe level
- Route status
- Port-to-port connectivity for this route:
    - From – *Node*

- To – *Node*
- Cable Start – *Node.slot.interface.paroli*
- Cable End – *Node.slot.interface.paroli*
- Link State – Up or down

Viewing Partitions Using Chassis and Links Perspective

The topology panel displays these items:

- Chassis – Grey borderless rectangles enclosing all nodes.
- Nodes – Multi-color node icons, displayed within the grey (chassis) rectangles that are the physical locations of the nodes.
- Links – Solid lines connecting nodes.
- Alarm Icons – Standard Sun Management Center error icons at the midpoint of the link.
- Striping Bracelet – Shown at the top left corner of the panel. Bracelets indicate striping level.
- Info – Informational messages are supplied at the bottom of the Topology view panels. The messages provide information about the object that the mouse cursor is over.

Note – When a partition includes switches, the Topology panel displays switches overlapping each other.

Viewing Link Properties

Double-click on a link in the Chassis and Links view to see data about that link.

The Link Properties dialog box displays:

- Nodes linked
- Stripe level
- The port-to-port connectivity:

    - Cable start – *Node.slot.interface.paroli*
    - Cable end – *Node.slot.interface.paroli*
    - Link status – Up or down
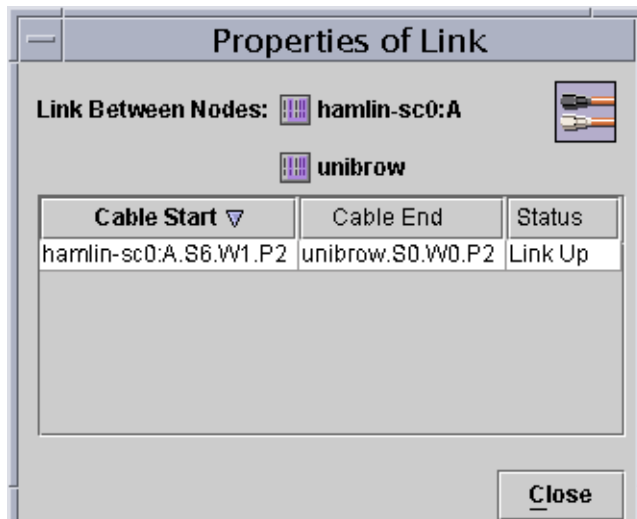
FIGURE 9-5   Link Properties Dialog Box

Displaying the Properties of Nodes in Partitions

▼ To View Node Properties

● **Double-click on a node icon to view the Node Properties dialog box for that node.**

Key node information is shown at the top, including:

- SC Host Name
- Domain Id
- SC IP Address
- Chassis Type
- Solaris Hostname
- Solaris IP Address
- Memory
- Agent Port
- Proxy Agent Hostname
- Physical Location
- WCI Info: Mode – Always RSM

- WCI Info: WCIS – The Sun Fire Link Interfaces, listed by slot
- WCI Info: Link Info (status)
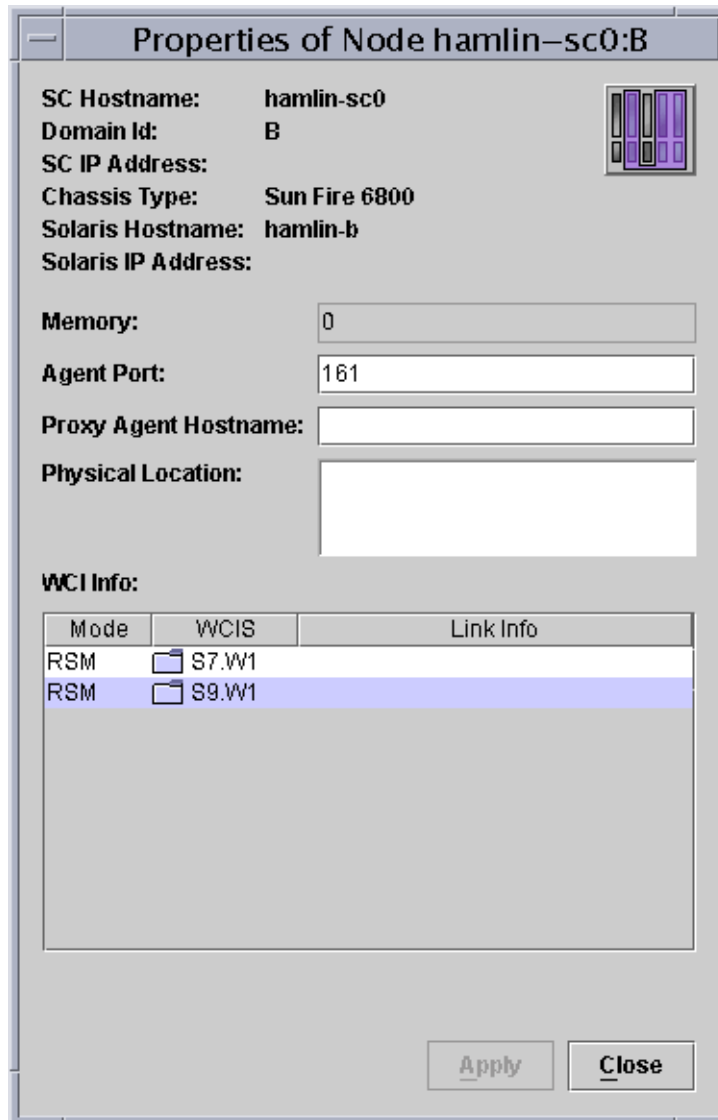
FIGURE 9-6   Node Properties Dialog Box

▼ To View Switch Properties

● **Double-click on a node icon to view the Node Properties dialog box for that node.**

Key node information is shown at the top, including:

- SC Hostname
- Domain ID
- SC IP Address
- Chassis Type
- WCI Info: Mode – Always RSM
- WCI Info: WCIS – The Sun Fire Link Interfaces, listed by slot
- WCI Info: Link Info (status)
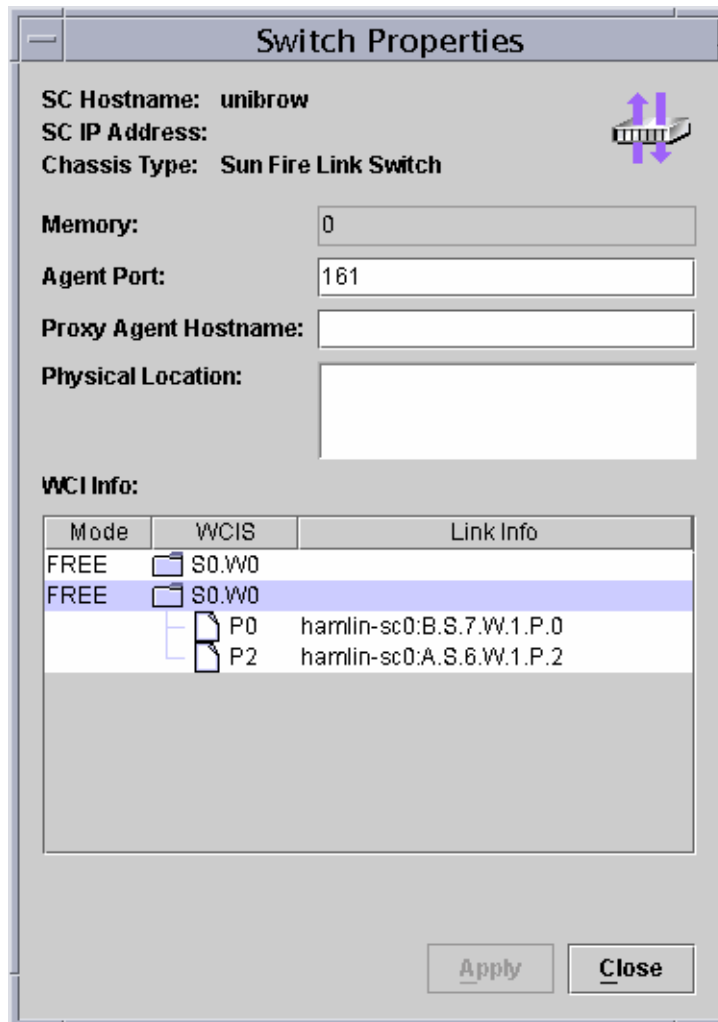
FIGURE 9-7   Switch Properties Dialog Box

▼ To View Chassis Properties

● **Double-click on the chassis shadow to see data about the Chassis.**

This data includes:

■ SC Hostname
■ SC IP Address
■ Chassis Type

- Node List:
  - Domain ID
  - Mode – RSM
  - Solaris Hostname
  - Solaris IP Address
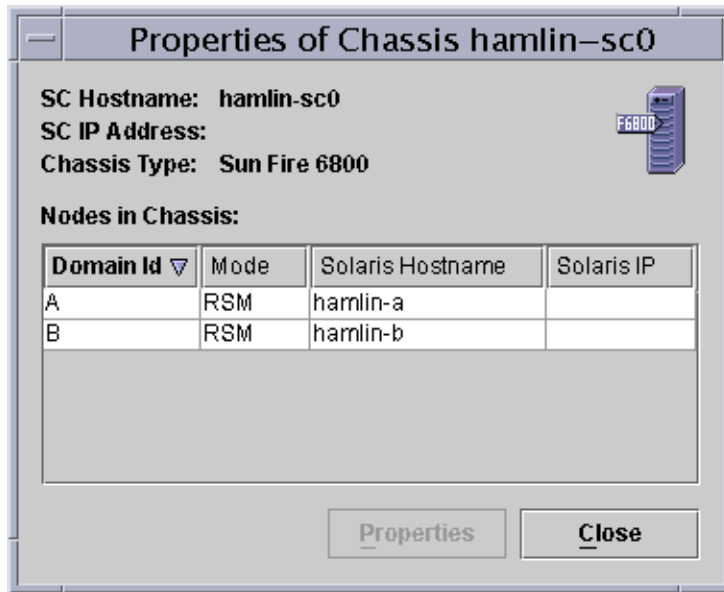
FIGURE 9-8   Chassis Properties Dialog Box

The Sun Management Center software retrieves Sun Fire Link chassis information from the data managed by agent modules residing on the nodes themselves.

The table displays all existing nodes along with any available summary information. Using the Node Details button, you can launch the Node Properties dialog box for a selected node.

Displaying Fabric Hardware Data

▼ To View Data About Fabric Hardware Devices

1. **In the Sun Management Center console, double-click a Sun Fire server system icon.**

   This brings up the Details window for that system.

2. **Select the Browser tab in the Details window.**

   The Config-Reader tree is displayed (as shown in FIGURE 9-9).
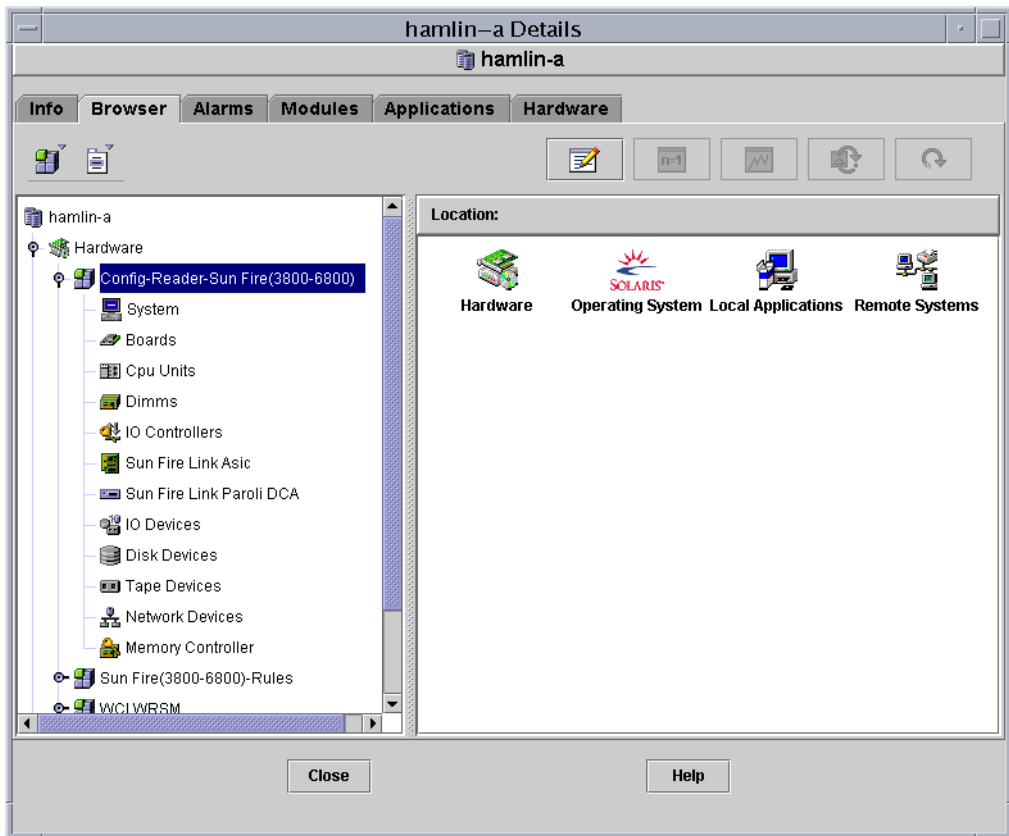
FIGURE 9-9   Config-Reader Tree

3. **Double-click Sun Fire Link–related items in the tree.**

   For example you can view tables for the Sun Fire Paroli DCA as shown in FIGURE 9-10, the Sun Fire Link ASIC, and ASICs (which include the cPCI I/O boards installed in the Sun Fire Link assemblies) as shown in FIGURE 9-11.

   You can also view the Sun Fire Link ASIC data table, as shown in FIGURE 9-12.
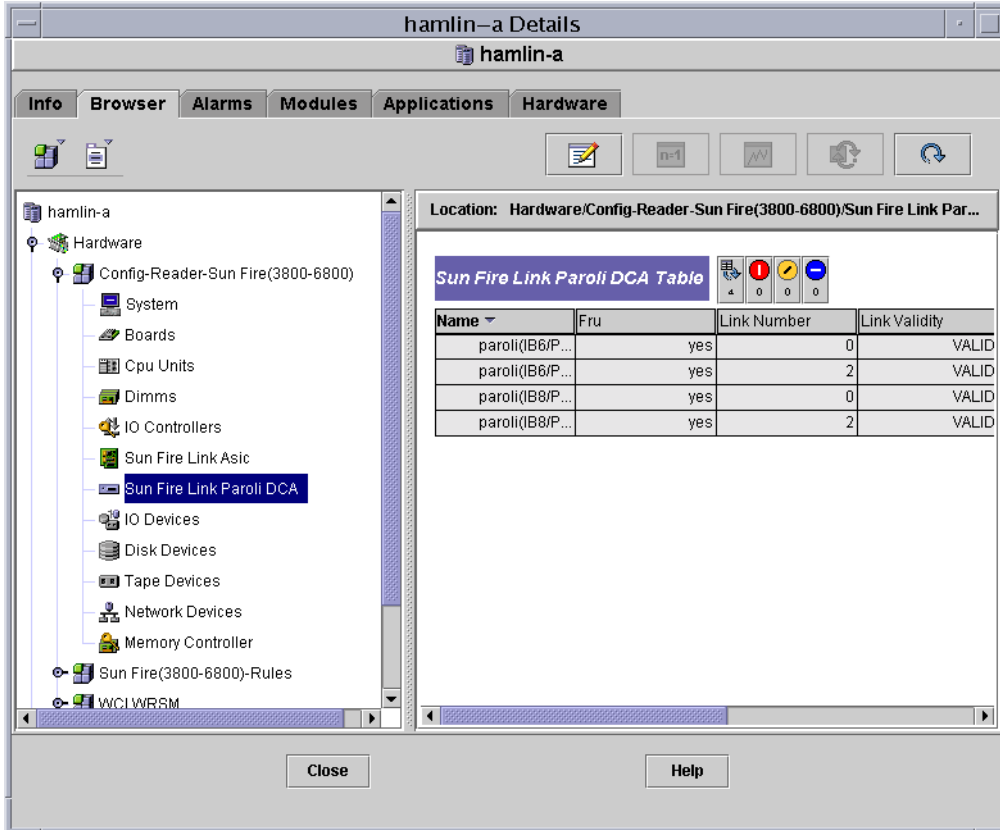
FIGURE 9-10 Config-Reader Paroli Data table
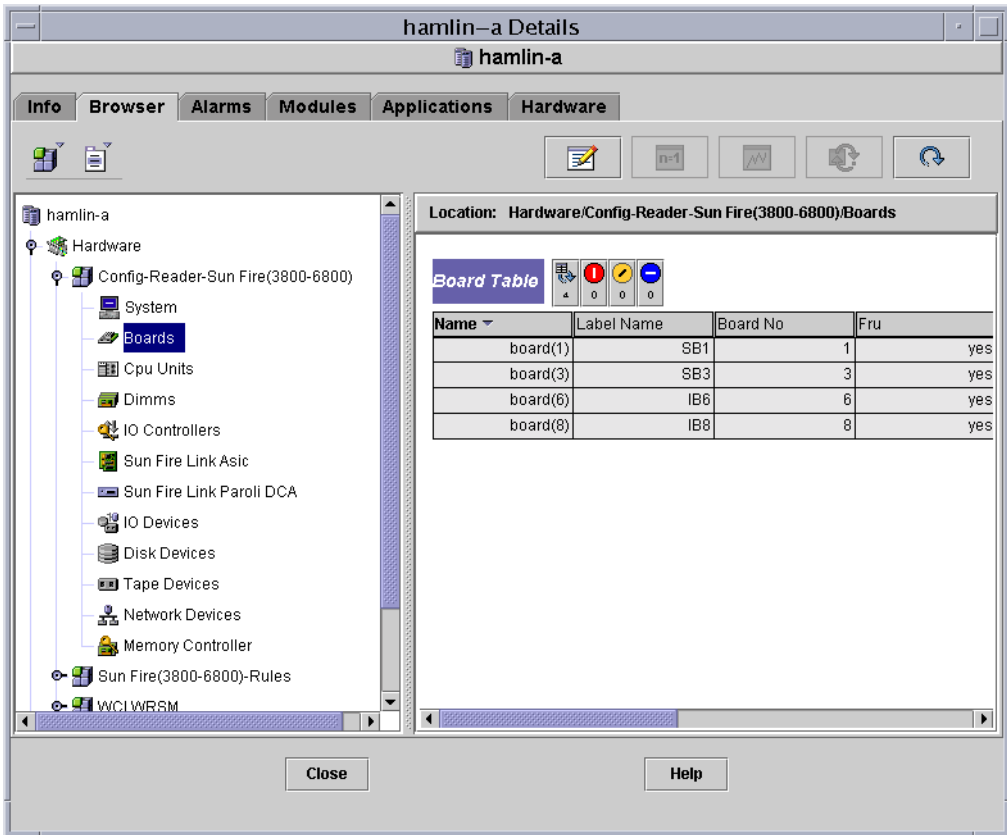
FIGURE 9-11 Config-Reader Board Data Table
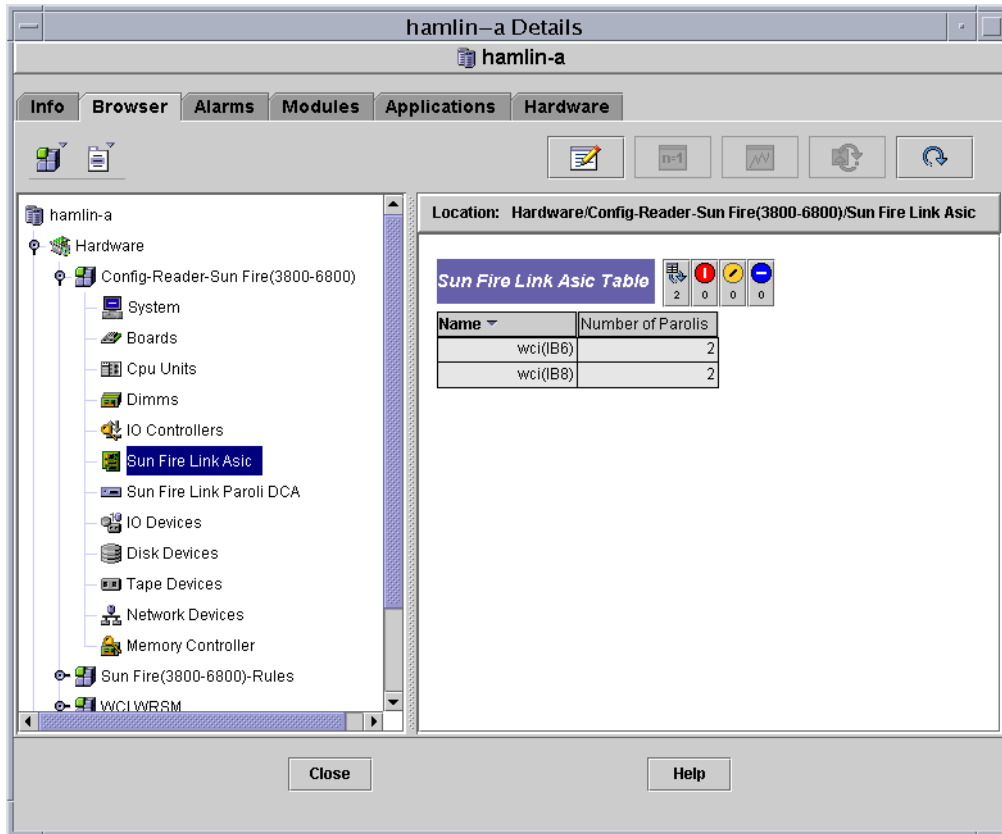
FIGURE 9-12 Config-Reader Sun Fire Link ASIC Data Table

▼ To See Physical Views of Fabric Devices

1. Click open the Hardware tab of the node details window

2. In the Views pull-down menu, select an item under either Physical View or Logical View. See FIGURE 9-13**.**
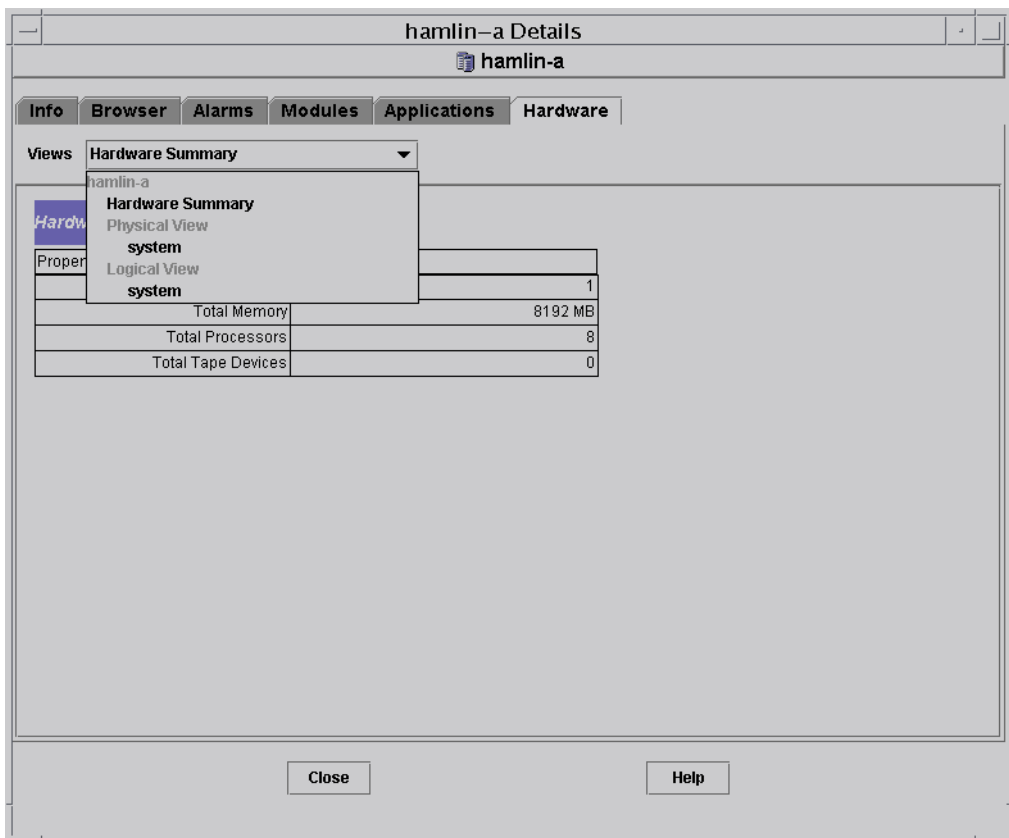
FIGURE 9-13 Hardware Summary List of Components in Hardware Details Window

If you select an entry named `system` under the heading of the Physical View, you can then move the cursor over a front or rear image of the system.

By leaving the cursor over a component, you cause the physical viewer to display information about that component. For example, the image displayed in FIGURE 9-14 is the rear view of a Sun Fire 6800 system. The cursor has highlighted Paroli IB8.

Field replaceable units (FRUs) are highlighted in the image as the cursor passes over them. If you hold the cursor still for several seconds, the corresponding properties and values are displayed in the properties window.

The properties displayed when a system is selected are:

- Name – The name of the selected component
- FRU – Whether the component is a field-replaceable unit
- Link Number – The link port number, which may be either 0 (zero) or 2.

- Link Validity – Links may be valid or invalid. Invalid states raise alarms.
- Link State – Links may be up or down. There are several alternative down states, each of which raise distinct alarms.

Link States	Alarm Level
LINK UP	No alarm
LINK DOWN	Error – red, critical
LINK NOT THERE	Info – blue, caution
WAIT FOR SC TAKEDOWN	Warning – yellow, alert
WAIT FOR SC LINK UP	Warning – yellow, alert
SC ERROR WAIT FOR LINK DOWN	Warning – yellow, alert
UNKNOWN	Error – red, critical

- Remote Link Number – The number (0 or 2) of the corresponding component at the far end of the link.
- Remote Cluster Member – The name of the node containing corresponding component at the remote link

FIGURE 9-14 Physical View of Sun Fire 6800 System
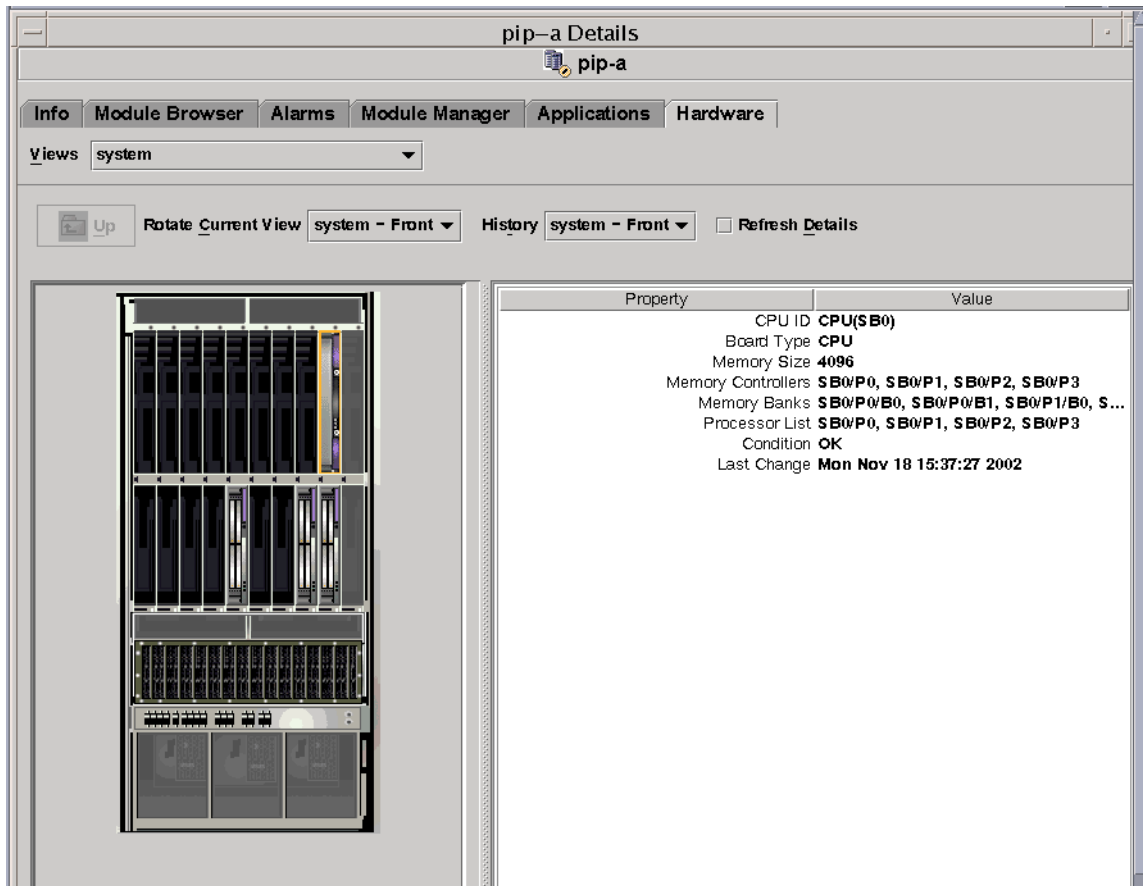
FIGURE 9-15 Physical View of Sun Fire 15K/12K System

FIGURE 9-16 shows a physical view of a Sun Fire Link switch. In this example, property values are listed for the Paroli board in slot 7.

The properties and values displayed for switch components are described in "Monitoring Switch Information" on page 101.
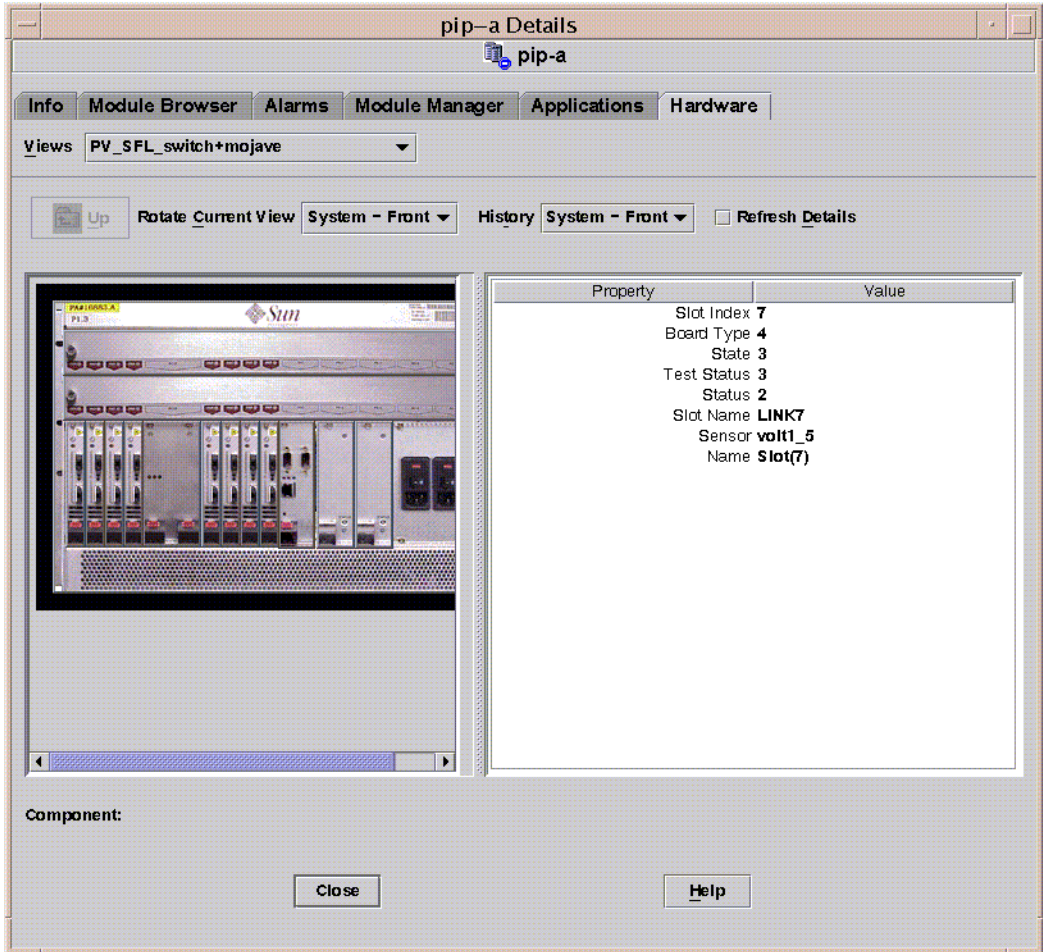
FIGURE 9-16 Physical View of Switch

If you selected the Logical View, click on an icon to display its properties in the properties window. You can double-click some icons to see various component parts, or click the Expand All button to see all the component parts in the system. Various properties and values are displayed in the Properties window.

Logical views display information about selected components in a tabular format. For example, FIGURE 9-17 shows the logical view of the board in IB8. This is the same I/O location as shown in FIGURE 9-14, except this lists details for PAR1.
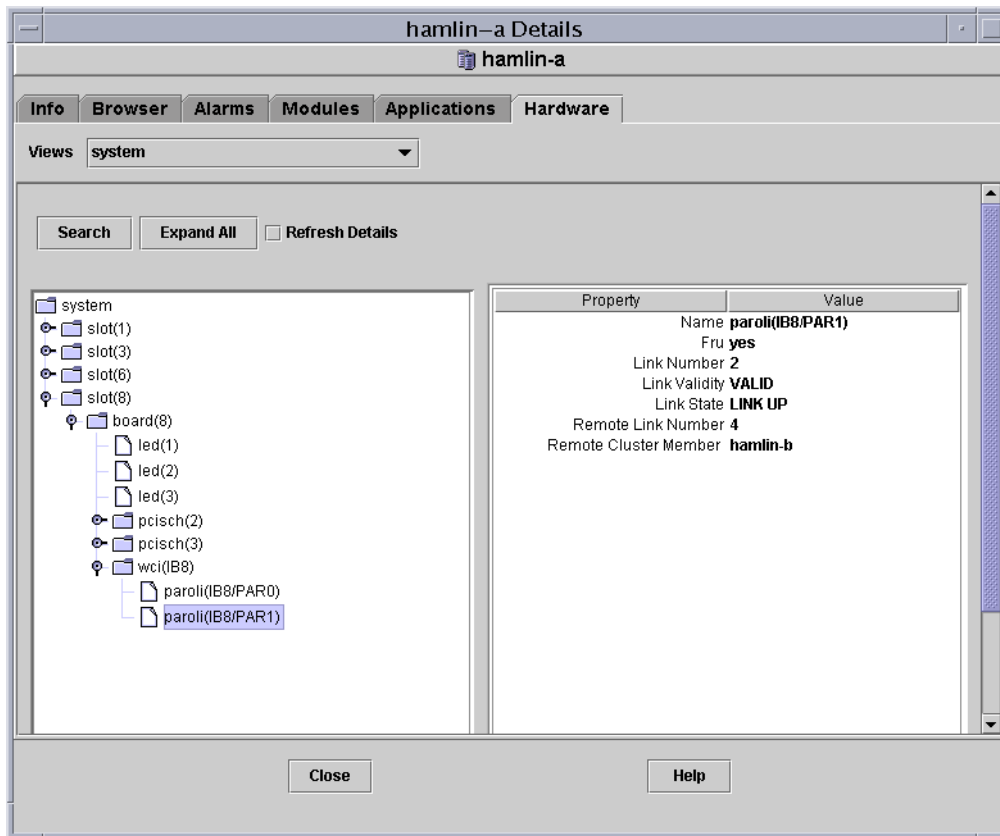
FIGURE 9-17 Logical View Of Paroli IB8 ⁄ PAR1

For further information about the Physical Viewer and Logical Viewer on Sun Fire 6800 systems, see the *Sun™ Management Center 3.0 Supplement for Sun Fire™ 6800/4810/3800 Systems.*

Monitoring Your Fabric Through System Details

You can use the Sun Management Center console to display information about the makeup of your Sun Fire Link partition by opening the Details panel on a Sun Fire System in your cluster. Several Sun Fire Link add-on modules supply information through the Details panel.

The Details panel gives you information about your fabric without requiring that you open the Fabric Console.

You can view:

- FM properties tables
- Fabric information tables
- RSM properties and data tables

FIGURE 9-18 Sun Fire Link FM Properties Table

FM Properties Table

- Fabric Name – The name of the fabric
- Server Name – The name of the Fabric Manager host
- Port – By default, the port number is 1099.
- FM Status – OK or "Could not connect"
- FM Version – The version number of the Fabric Manager software, or "Could not connect"
- FMDAQ Log Name – The log of messages sent between the FM components installed in Sun Management Center server and the Fabric Manager itself
- FMDAQ Log Size – The size (in bytes) of the FMDAQ log
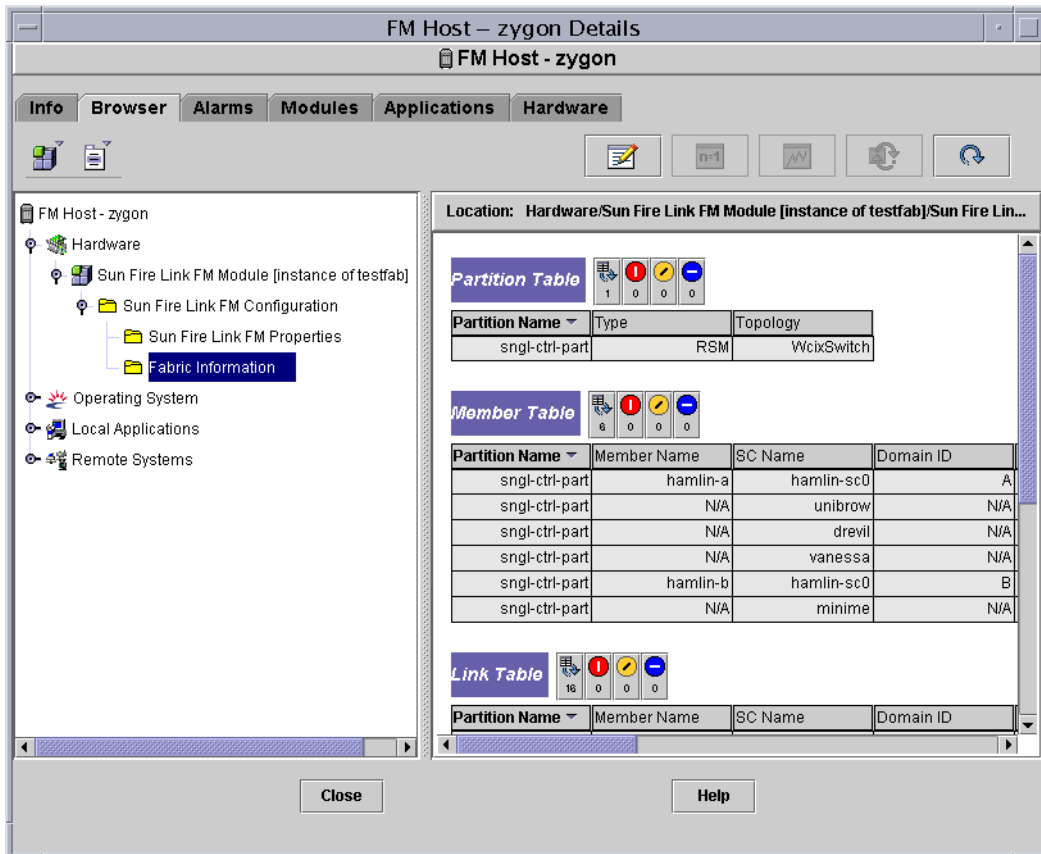
FIGURE 9-19 Sun Fire Link FM Details Tables

Fabric Information Tables

- Partition Table
  - Partition Name – The name of the partition
  - Type – RSM
  - Topology - Direct Connect or WcixSwitch
- Member Table
  - Partition Name – The name of the partition
  - Member Name – The name of the node or switch
  - SC Name – The name of the system controller
  - Domain ID – The ID of the node domain: A, B, C, or D
  - Chassis Type – Types can include: S24 (Sun Fire 6800), S72 (Sun Fire 15K/12K), or WCIX_SWITCH
  - Agent Port – By default, the agent port number is 161
- Link Table
  - Partition Name – The name of the partition
  - Member Name – The name of the node or switch
  - SC Name – The name of the system controller
  - Domain ID – Domain ID – The ID of the node domain: A, B, C, or D
  - Local WCI Slot – The number of the local Sun Fire Link Assembly slot
  - Local WCI ID – The ID of the I/O WIB (always 1)
  - Local Port Number – Either 0 or 2
  - Switch Port Number – A value from 0 to 7
  - Link Status – LINK UP or *unknown*
  - Link State – MPI communications require LINK UP. All other states halt MPI communications.

Link States
LINK UP
LINK DOWN
LINK NOT THERE
WAIT FOR SC TAKEDOWN
WAIT FOR SC LINK UP
SC ERROR WAIT FOR LINK DOWN
UNKNOWN

- Remote Node Name – The name of the remote node
- Remote WCI Slot – The number of the remote Sun Fire Link Assembly slot
- Remote WCI ID – The ID of the I/O WIB
- Remote Port Number – Either 0 or 2

FM Alarms

The Sun Fire Link Fabric Manager propagates alarms when abnormal or informational conditions arise during the monitoring of FM information through the Sun Management Center console. The following conditions will cause an alarm of the noted severity:

Condition	Severity Level (Medallion color)
FM Status not equal to OK	Warning (Yellow)
FM XML File Modification Status equal to Modified	Information (Blue)

Monitoring RSM Information

The WRSM Agent Module monitors information on each node in the fabric. It operates through Solaris libraries and commands, including the WCI RSM Driver, `kstat`, and `syseventd` facilities. The WRSM module gathers detailed RSM link metrics and summarizes those metrics in order to provide that information to other Fabric Manager agent modules. The agent module monitors the metric thresholds and signals failures based on those thresholds.

WNode RSM Properties

The WNode RSM Properties folder displays WNode Data, RSM Controller Data, and RSM Node WCI Data tables. The data displayed in these tables represents the internal configuration and state of the RSM driver.

The term *Wnodes* refers to nodes in partitions.

RSM controllers are involved in all partition configuration operations. One RSM controller is bound to each partition—a node can have multiple controllers, one for each partition in which it participates.
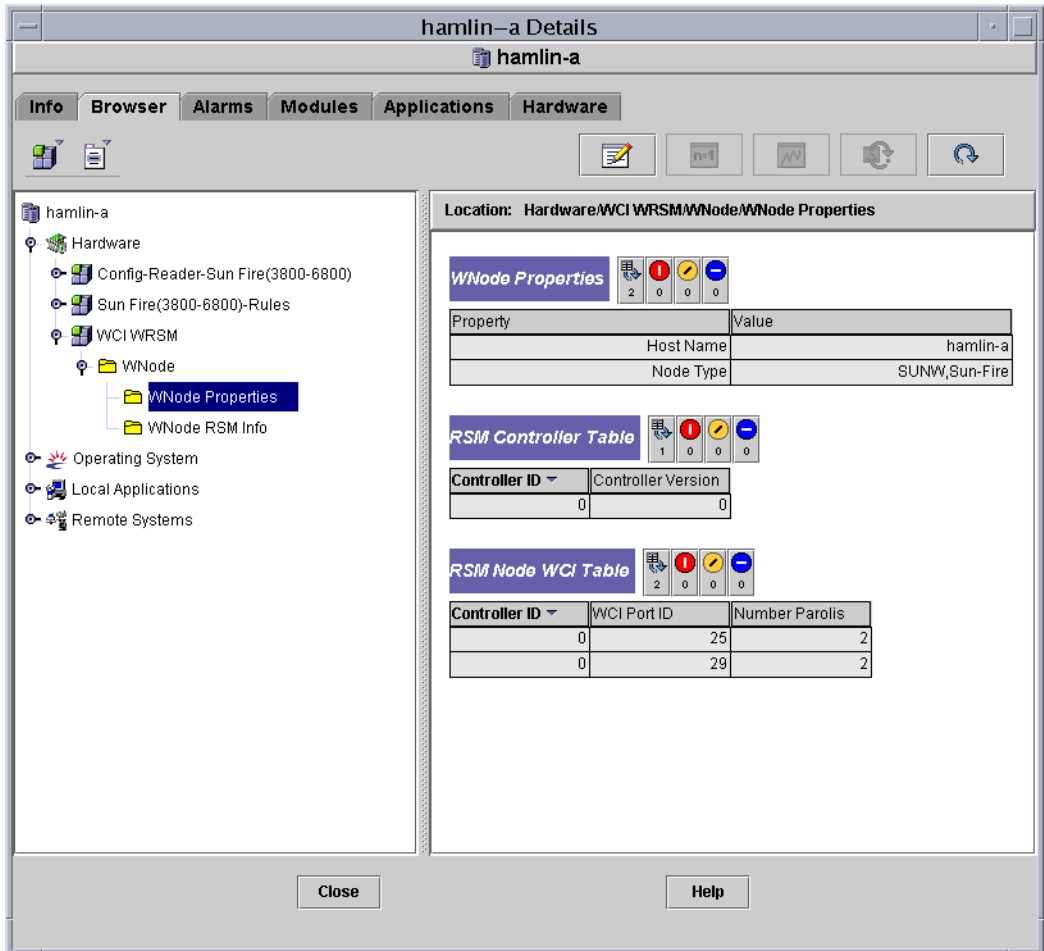
FIGURE 9-20 WNode RSM Properties Tables

- WNode Data
    - Host Name – The name of the node
    - Node Type – SUNW or Sun-Fire
- RSM Controller Data

    - Controller ID – The ID of the RSM controller.
    - Controller Version – The version of the RSM controller. This number is used by Sun Fire Link software to identify the current cluster configuration.

- RSM Node WCI Data

    - Controller ID – The ID of the RSM controller.
    - WCI Port ID – Address of WCI (always 1).

■ Number of Parolis – The default number is 2.

Wnode RSM Info

The Wnode RSM Info folder displays RSM Member, Link Status, and Route data tables.

FIGURE 9-21 Wnode RSM Info Tables

■ RSM Member Data

  ■ Controller ID – The ID of the RSM controller.

- Member Name – The name of the node.
- CNode ID – RSM cluster node ID, a unique value in the range 0:255. This number is used by Sun Fire Link software to route data to members of the partition.

- RSM Link Status Data

  - Controller ID – The ID of the RSM controller.
  - WCI Port ID – Address of WCI (always 1).
  - Link Number – Link port number. Either 0 or 2.
  - Link Status – Up or Down
  - Link State – The seven link states are

Link States
LINK UP
LINK DOWN
LINK NOT THERE
WAIT FOR SC TAKEDOWN
WAIT FOR SC LINK UP
SC ERROR WAIT FOR LINK DOWN
UNKNOWN

  - Link Validity – Valid or Invalid
  - Remote CNode ID – RSM cluster node ID, a unique value in the range 0:255.
  - Remote WNode ID – a value in the range 0:15
  - Remote WCI Port ID – Address of WCI.
  - Remote Link Number – Remote link port number. Either 0 or 2.
  - Link Error Takedowns – Count provided by `kstat`.
  - Link Configuration Takedowns – Count provided by `kstat`.
  - Link Failed Bringups – Count provided by `kstat`.
  - Maximum Link Errors – Count provided by `kstat`.
  - Average Link Errors – Count provided by `kstat`.

- RSM Route Data

  - Controller ID – The ID of the RSM controller.
  - WCI Port ID – Address of WCI (always 1).
  - Link Number – Link port number. Either 0 or 2.
  - Route Type – Passthrough or Multihop.
  - Number of Route Changes – Count provided by `kstat`.
  - Destination Hostname – The hostname of the system at the far end of the link.
  - Destination CNode ID – A number used by the Sun Fire Link ASIC and driver to route data to members of the partition.
  - Destination GNID – A number used by Sun Fire Link software to route data within the partition.

Alarms

The WRSM Agent Module checks `kstat` data for values exceeding predefined thresholds. The WRSM Agent module triggers alarms or warnings when appropriate. The following conditions can cause an alarm of the noted severity for any monitored link:

Condition	Severity Level (Medallion color)
Link Validity is Invalid	Error (Red)
Link State is Link Down	Error (Red)
Link State is SC Error Wait for Link Down	Error (Red)
Link State is an Unknown State	Error (Red)
Link State is Wait for SC Link Takedown	Error (Red)
Link State is Link Not Present	Information (Blue)
Link State is Wait for SC Link Up	Warning (Yellow)
Link Error Takedown Count is greater than Lower Threshold	Warning (Yellow)
Link Error Takedown Count is greater than Upper Threshold	Error (Red)
Link Configuration Takedown Count is greater than Lower Threshold	Warning (Yellow)
Link Configuration Takedown Count is greater than Upper Threshold	Error (Red)
Link Failed Bringup Count is greater than Lower Threshold	Warning (Yellow)
Link Failed Bringup Count is greater than Upper Threshold	Error (Red)
Max Link Error Count is greater than Lower Threshold	Warning (Yellow)
Max Link Error Count is greater than Upper Threshold	Error (Red)
Average Link Error Count is greater than Lower Threshold	Warning (Yellow)
Average Link Error Count is greater than Upper Threshold	Error (Red)

Monitoring Switch Information

The Switch module's data displays the entire Sun Fire Link Switch agent database, excluding traps, as shown in this section.
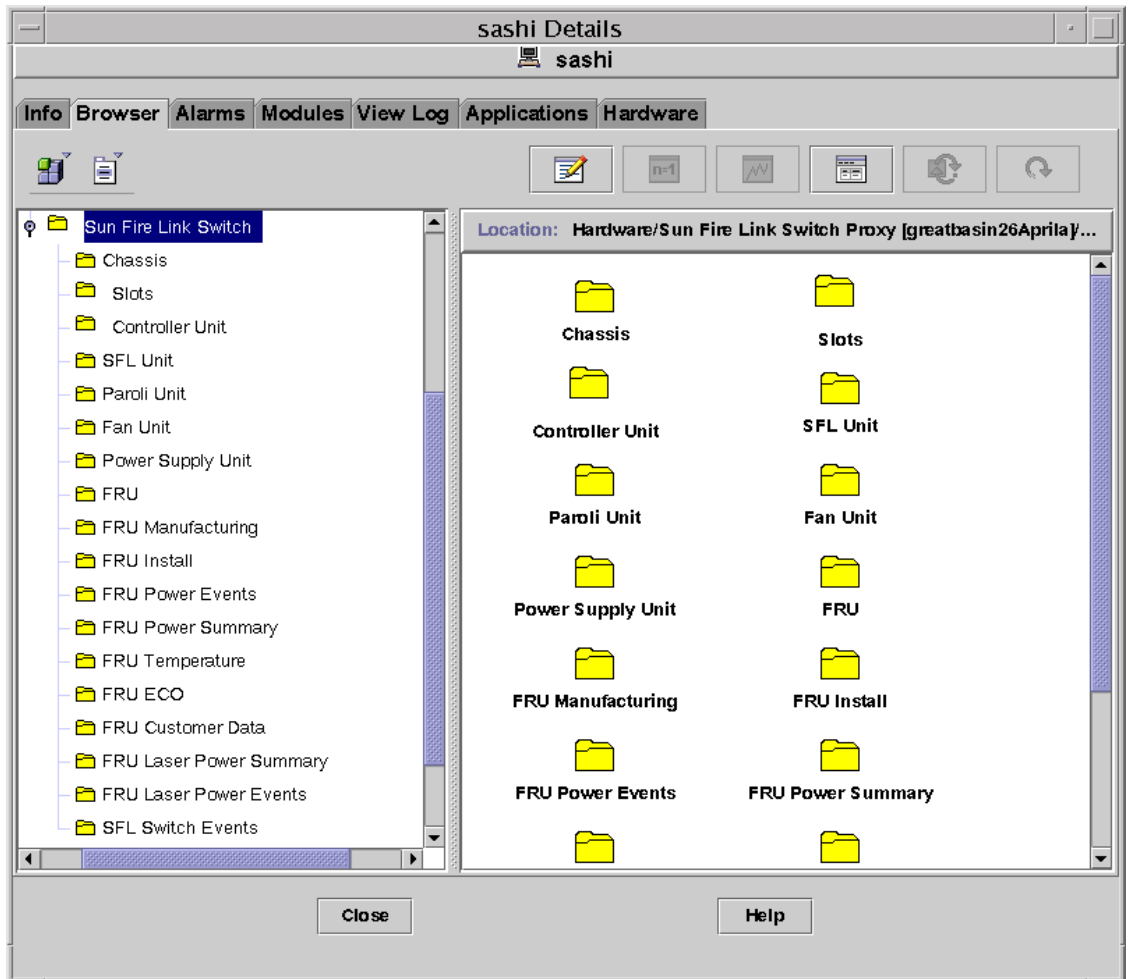
FIGURE 9-22 Switch Proxy Agent Icons on Host Hardware Details Panel

By double-clicking a switch proxy, you open the tree view of the tables of data supplied to the Sun Management Center software by the switch proxy. You can then select a table in the tree view and see its properties and values displayed in the right hand panel.

Chassis Data

The Chassis table stores overview-level data about the switch.

- `Description` – The switch_description of the switch chassis created using Switch CLI set or `setupplatform` commands.
- `FRU Index` – Pointer to the FRU data for the switch backplane.
- `Name` – The hostname of the switch.
- `Location` – User-writable field for storing description of the physical location of the switch.
- `Contact` – User-writable field for storing contact data for the person(s) or groups responsible for this switch's maintenance.
- `Log Host` – User-writable field for storing the name of the log host.
- `Trap Host` – User-writable field for storing the name of the list of trap hosts.
- `Slots` – The number of slots in the switch. This determines the number of entries in the slots table for the switch.
- `Up Time` – The time (in hundredths of a second) since the switch controller was last re-initialized.
- `Ethernet Address` – The MAC address used to access this switch's controller.
- `Name` – Data supporting physical and logical views of the switch.

Slot Data

The slot table stores a list of switch slots, board types occupying those slots, and information such as board status. A slot entry is created for all possible slots in a switch at reset. If a slot is empty at reset, it may be occupied later.

- `SlotIndex` – Integer identifying the specific unit.
- `Index` – Integer identifying the specific slot.
- `Board Type` – The type of board installed in this slot in the chassis. See TABLE 9-1:

TABLE 9-1 `Slot Board Type` values

Value	Meaning
1	empty
2	controller
3	wciX
4	paroli
5	fan
6	powerSupply

- `State` – Current state of the board. If the slot occupant is broken it has the value of *unknown*. If a link is currently operating but has not completely determined its role, it has the value of *assigned*. If the component is not available for use, either because it has failed tests or because it is under test currently, its state is *unavailable*. If the board can be safely removed, it's status is *removable*:

TABLE 9-2   `State` values

Value	Meaning
1	unavailable
2	unknown
3	available
4	assigned
5	active
6	removable

- `Test Status` – Test status of the board occupying this slot.

TABLE 9-3   `Test Status` values

Value	Meaning
1	unknown
2	notTested
3	passed
4	failed
5	underTest

- `Status` – The status of a board, mainly used in traps. If a slot is not accessible the status is *unknown*. If all monitored slot environmental variables are within the warning limits, the status is *ok_green*. If one or more of the environmental variables are higher or lower than the absolute high and low limits, the status is *hiRed* or *loRed*, respectively. If some variables are above or below the warning

limits, the state is *hiYellow* or *loYellow* respectively. If the status is not one of *unknown* or *okGreen*, the field slotSensor contains a string describing the sensor that contributed to the slot status. See TABLE 9-4.

TABLE 9-4 `Status` values

Value	Meaning
1	unknown
2	okGreen
3	loYellow
4	loRed
5	hiYellow
6	hiRed

- `Slot Name` – String describing the slot name.
- `Sensor` – If the status is red or yellow, the name of the sensor that contributed to that status.
- `Name` – Data supporting physical and logical views of the switch.

Controller Unit Data

There is a slot for one Switch System Controller (SSC) unit per switch. If no controller unit has been installed, none of this data is available because the processor and ethernet controller on the SSC card provide the SNMP service supplying the data.

- `Index` – Integer identifying the specific unit.
- `FRU Index` – An index into the FRU table identifying the information associated with this device
- `Sw Version` – An identifying string for the version of software running on this controller card.
- `Time Zone` – A human-readable description of the time zone used for dates on this controller.
- `Date` – The date and time of the controller in format *mmddHHMMSSccyy*.

See TABLE 9-5 for an explanation of these fields:

TABLE 9-5   `Date` values

Field	Meaning
mm	month (1-12)
dd	day (1-31)
HH	hour (0-23)
MM	minute (0-59)
SS	second(0-59)
cc	time zone
yy	least significant two digits of the current year

- `Net Config` – Describes the network configuration and how the controller identifies itself on the network. TABLE 9-6 defines the possible values.

TABLE 9-6   `Net Config` values

Value	Meaning
1	unknown
2	notNetworked
3	fixed
4	dhcp

- `Hostname` – Hostname of this controller.
- `IP` – IPV4 address of this controller. If `controllerNetConfig` was `dhcp` when this controller last booted, this field was implicitly set to the value returned by `dhcp` during initialization.
- `Netmask` – The netmask used for IP network communications by this controller.
- `Default Routers` – A space-separated list of routers for this controller.
- `DNS Domain` – The DNS domain for this controller.
- `DNS Resolvers` – The DNS resolvers used by this controller.

■ `PowerLED` – State of the power LED on the controller. See TABLE 9-7.

TABLE 9-7 `PowerLED` values

Value	Meaning
1	on
2	off
3	unknown

■ `Fault LED` – State of the fault LED on the controller. See TABLE 9-8.

TABLE 9-8 `Fault LED` values

Value	Meaning
1	on
2	off
3	unknown

■ `Hot Plug LED` – State of the HotPlug LED on the controller. See TABLE 9-9.

TABLE 9-9 `Hot Plug LED` values

Value	Meaning
1	on
2	off
3	unknown

■ `State` – Describes the state of the SSC (always `up`)
■ `Volt 33` – Voltage detected on the 3.3 Volt line (in hundredths of Volts).
■ `Volt 50` –Voltage detected on the 5.0 Volt line (in hundredths of Volts).
■ `Volt 120` – Voltage detected on the 12.0 Volt line (in hundredths of Volts).
■ `Temperature` – Board temperature (hundredths of Celsius Degrees).

- **Nvram Backup Battery** – Describes the status of the NVRAM backup battery. See TABLE 9-10.

TABLE 9-10 Nvram Backup Battery values

Value	Meaning
1	unknown
2	ok
3	low

- **Host ID** – Controller Host ID.
- **Name** – Data supporting physical and logical views of the switch.

SFL Unit Data

This table describes SFL switch unit data.

- **Index** – Integer identifying the specific unit.
- **FRU Index** – The index into the FRU Table of the FRU information for this SFL unit.
- **Description** – Sun Fire Link Crossbar Module.
- **State** – An SFL unit is *unknown* if we cannot communicate with it (such as when the card is missing) or we have not processed it yet after power has been applied. Once the device is known to exist, it transitions to either *notTested* if power is just being applied, or *passed* if this is just a controller reboot and the chip can be determined to already have been tested such as when it is currently in use. If the device is tested and has failed said test, the state transitions to *failed*. *inReflectionMode* indicates that one or more of the links have been put into reflection mode for SunVTS testing. See TABLE 9-11.

TABLE 9-11 State values

Value	Meaning
1	unknown
2	notTested
3	passed
4	inReflectionMode
5	failed

- **Temperature** – The internal temperature of the SFL unit (in hundredths of Celsius degrees).

- `ASIC Temp A` – The internal temperature of ASIC A of the SFL unit (in hundredths of Celsius degrees).
- `ASIC Temp B` – The internal temperature of ASIC B of the SFL unit (in hundredths of Celsius degrees).
- `Volt 15` – Voltage detected on the 1.5 Volt line (in hundredths of Volts).
- `Volt 25` – Voltage detected on the 2.5 Volt line (in hundredths of Volts).
- `PS Volt 25 Status` – Status of the 2.5 Volt power supply. See TABLE 9-12.

TABLE 9-12 `PS Volt 25 Status` values

Value	Meanings
1	unknown
2	nominal
3	failed

- `Volt 33` – Voltage detected on the 3.3 Volt line (in hundredths of Volts).
- `Volt 50` – Voltage detected on the 5.0 Volt line (in hundredths of Volts).
- `Name` – Data supporting physical and logical views of the switch.

Paroli Unit Data

This table describes the current state of each Paroli.

- `Number` – Integer (0-7) identifying the specific Paroli.
- `State` – Describes the current state of a specific Paroli. See TABLE 9-13.

TABLE 9-13 `State` values

Value	Meaning
1	notinstalled
2	postfailed
3	disabled
4	waitup
5	up
6	undertest
7	off

- `Link State` – The state of each link. If there does not appear to be a card installed, it is *notinstalled*. If the link is not active, it is *off*. If a signal has been detected from the other side of the link and the link is being brought up, it is *seek*. If the link is up and being used, it is *inuse*. In case of any hardware failure during the link state query, it is *notinstalled*. See TABLE 9-14.

TABLE 9-14 `Link State` values

Value	Meaning
1	notinstalled
2	off
3	seek
4	(not used)
5	inuse

- `Endpoint` – A textual description of the far end of a link, typically including a representation of fabric node identifier, remote link number, and remote Sun Fire system identifier.
- `Error Rate` – A comma-separated list of statistics on the rate at which errors are accumulating. The format of this string is:

  <Short>,<Long>,<MaxShort>,<MaxLong>,<AveShort>,<AveLong>,<Total>.

  Short and Long are the number of errors during the current short and long period (usually 1 hour and 24 hours), Total is the total number of errors since the link has been activated, and the remaining values describe the maximum and average number of errors during short and long periods. If a link is not active or there is an error in accessing this data, the string will have the value *unknown*.
- `Config` – Reserved for internal use.
- `Control High` – Reserved for internal use.
- `Control Low` – Reserved for internal use.
- `Status High` – Reserved for internal use.
- `Status Low` – Reserved for internal use.
- `Num Short Intervals` – The total number of short intervals passed since the link has been activated.
- `Num Minutes Per Short` – The length of a short error monitoring interval in minutes.
- `Num Short Per Long` – The number of short intervals contained in a long interval for error monitoring purposes.
- `Num Rolling Average` – The number of samples used to compute the rolling averages of the link error rate.

- FRU Index – An index into the FRU Data table which gives FRU information for this unit.
- Temperature – Temperature in the Paroli module (in hundredths of Celsius degrees).
- Volt 33 – Voltage detected on the 3.3 Volt line (in hundredths of Volts).
- Volt 15 – Voltage detected on the 1.3 Volt line (in hundredths of Volts).
- Detail – Always *none.*
- LEDs – A report of the state of the LEDs. See TABLE 9-15.

TABLE 9-15   LEDs   values

Value	Meaning
1	allOff
2	greenLinkOn (laser on)
3	yellowLinkOn (error state)
4	allOn (laser on and error state)

- Name – Data supporting physical and logical views of the switch.

Fan Unit Data

Information about the fan units plugged into this Switch.

- Tray Index – Integer identifying the specific fan unit.
- Tray State – If a fan tray is absent, *unknown.* See TABLE 9-16.

TABLE 9-16   Tray State   values

Value	Meaning
1	unknown
2	nominal
3	failed

- Speed 0 – The speed (in RPM) of fan 0 in the tray.
- Speed 1 – The speed (in RPM) of fan 1 in the tray.
- Speed 2 – The speed (in RPM) of fan 2 in the tray.
- Name – Data supporting physical and logical views

Power Supply Unit Data

Information about the power supply units plugged into this Switch.

- `Index` – Integer identifying the specific power supply unit.
- `State` – See TABLE 9-17.

TABLE 9-17 `State` values

Value	Meaning
1	unknown
2	nominal
3	degraded (close to failure)
4	failed

- `Name` – Data supporting physical and logical views of the switch.

FRU Data

Every entity that is a FRU has a pointer to a row in this table. This table includes all generic FRU data fields. Every element of the table is a number of indices into other tables containing FRU data. An index value of -1 denotes that there are no such data available for the given FRU. When multiple FRU records of a type are available for a FRU, they are presented as table entries with contiguous indices. Each `Index` column contains the starting index, and each `Num Records` column contains the number of records. The most recent record is the one with index of minus one, such as Num Records = –1.

- `FRU Index` – A unique index for this FRU within the switch chassis.
- `Manufacturing Table Index` – Index into the table of FRU Manufacturing data.
- `Install Table Index` – Index into the table of FRU Install data.
- `Install Table Num Records` – Number of Install data records for this FRU.
- `Power Summary Table Index` – Index into the table of FRU Power Summary data.
- `Power Events Table Index` – Index into the table of FRU Power Events data.
- `Power Events Table Num Records` – Number of Power Event records for this FRU.
- `Temperature Table Index` – Index into the table of FRU Temperature data.
- `ECO Table Index` – Index into the table of FRU ECO data.
- `Customer Data Table Index` –Index into the table of FRU customer data.

- `Laser Power Summary Table Index` – Index into the table of laser power summary data.
- `Laser Power Events Table Index` – Index into the table of laser power events data.
- `Laser Power Events Table Num Records` – Number of laser power event records for this FRU.

FRU Manufacturing Data

- `Index` – Unique integer identifier.
- `Vendor Name` – Vendor Name: JEDEC identifier.
- `Location` – Manufacturing location.
- `Date` – Date of Manufacture.
- `Part Number` – Sun Part number.
- `Hw Dash Level` – Initial HW Dash Level.
- `Hw Revision Level` – Initial HW Revision Level.
- `Serial Number` – Sun serial number.
- `Short Name` – Short name of this FRU.
- `Description` – Description of this FRU.

FRU Install Data

- `Index` – Unique integer identifier.
- `Time Stamp` – Records the last update to this record.
- `FRU Path` – Hierarchical description of how the FRU is connected to the system, in physical terms.
- `Part Number` – Identifies the parent FRU.
- `Serial Number` – Identifies the parent FRU.
- `Dash Level` – Identifies the parent FRU.
- `System ID` – Unique identifier for the system.
- `Time Zone` – System Timezone.
- `Geo North` – Geographical position (*northing*), where available.
- `Geo East` – Geographical position (*easting*), where available.
- `Geo Alt` – Geographical position (altitude), where available.
- `Geo Location` – System location.

FRU Power Events Data

- `Index` – Unique integer identifier.
- `Time Stamp` – Records the last update to this record.
- `Event` – See TABLE 9-18.

TABLE 9-18 `Event` values

Value	Meaning
1	powerOn
2	stillOn
3	powerOff

FRU Power Summary Data

- `Index` – Unique integer identifier.
- `Time Stamp` – Records the last update to this record.
- `Total Time On` – Total time the FRU has been powered on, measured in minutes.
- `Total Power Ons` – The number of times that the FRU power has been switched on.
- `Total Power Offs` – The number of times that the FRU power has been switched OFF.

FRU Temperature Data

- `Index` – Unique integer identifier.
- `Time Stamp` – Records the last update to this record.
- `Sensor` – Always 0 (zero), indicating (by ordinal value) the sole temperature sensor on the board.
- `Lowest` – The lowest ambient temperature in which the FRU has ever been operated.
- `Highest` – The highest ambient temperature in which the FRU has ever been operated.
- `Latest` – The most recent temperature in which the FRU has been operated.
- `Histogram` – This is a string encoding 10 integer values separated by comma. Each such value records the number of operating hours in a particular temperature range. The first element records hours below 20 C; each of the next eight records a 10 C range; and the final element, records hours over 100 C.

FRU ECO Data

- `Index` – Unique integer identifier.
- `Time Stamp` – Records the last update to this record.
- `Firmware Revision` – Records ECO changes that affect the firmware revision.
- `Hardware Revision` – Records ECO changes that affect the revision number.
- `HW Dash Level` – Records dash level changes from ECOs after the initial build.

FRU Customer Data

- `Index` – Unique integer identifier.
- `Time Stamp` – Records the last update to this record.
- `Data` – Customer-defined data.

FRU Laser Power Summary Data

- `Index` – Unique integer identifier.
- `Time Stamp` – Records the last update to this record.
- `Total Time On` – Total time the laser has been powered on, measured in minutes.
- `Total Power Ons` – The number of times that the laser power has been switched on.
- `Total Power Offs` – The number of times that the laser power has been switched off.

FRU Laser Power Events Data

- `Index` – Unique integer identifier.
- `Time Stamp` – Records the last update to this record.
- `Event` – See TABLE 9-19.

TABLE 9-19   `Event` values

Value	Meaning
1	powerOn
2	stillOn
3	powerOff

SFL Switch Events

The Sun Management Center console displays several additional switch-generated trap events. These events are:

- `SFL Switch On` – The SSC has newly come up.

- `SFL Switch Available` – Sent after POST tests have finished running.

- `SFL Switch Reboot` – Sent before the switch reboots.

- `SFLX Shutdown` – Sent when environmental conditions cause the switch to be powered down. The operator should power off the chassis promptly to prevent damage.

- `Self-test Result Change` – Sent when a POST results indicate a change. The Sun Management Center agent responds by refreshing switch data, checking for alarm conditions, and generating alarms (if necessary).

- `Environmental Status Change` – Sent when the status of an environmental sensor changes (for example, from OK to Warning). The Sun Management Center agent responds by refreshing the switch data, checking for alarm conditions, and generating alarms (if necessary).

- `Paroli Error Rate Status Change` – Sent when the number of errors encountered on a particular link exceeds the warning threshold or the shutdown limit threshold. The Sun Management Center agent responds by refreshing the switch data, checking for alarm conditions, and generating alarms (if necessary).

- `Paroli Environmental Shutdown` – Sent when a link has been shut down due to conditions detected by an environmental sensor.

- `Paroli Error Rate Shutdown` – Sent when a link has been shut down due to a high error rate.

- `NVRAM Backup Battery Low` – Sent when the voltage level is low on the system controller's NVRAM backup battery.

- `SNMP Community String Changed` – Sent when the SNMP community string has been changed on the switch. In order to keep SNMP communication viable, you must run `es-setup` to remove the instance of the switch agent module that is monitoring this switch. Then run `es-setup` again to create a new instance of the module for monitoring this switch and specify the new SNMP community string in response to the prompts.

- `Log Host Change` – Sent when the name of the host that provides a secondary log of switch events and errors has been changed. This does not affect Sync.

- `SNMP Trap Host Change` – Sent when the name of the SNMP trap host has been changed on the switch. This is a critical error, preventing any switch traps from reaching the Sun Management Center agent. The name of the SNMP trap host specified in the switch must match the name of the Sun Management Center server specified for the Sun Management Center agent that is monitoring the switch(es).

Note: To correct this condition, run `es-setup` and change the name of the Sun Management Center server associated with the agent. This needs to be done in the portion of `es-setup` that does setup for the entire agent. There is no need to run the portion of `es-setup` that does setup specifically for instances of the switch agent module.

- `Board Arrival` – Sent when a board has been added to the switch chassis, or enabled via the CLI. The corresponding switch data is refreshed in the Sun Management Center server.
- `Board Departure` – Sent when a board has been removed from the switch chassis, or disabled via the CLI. The corresponding switch data is refreshed in the Sun Management Center server.

Monitoring Switch Alarms

Some switch properties are configured to generate alarms. TABLE 9-20 identifies those properties and their alarm levels. It also identifies the values that trigger the different levels of alarm.

TABLE 9-20 Switch Alarms

Table: Property	String in Log	Info	Warning	Error	Notes
Slot: Test Status	slotTestStatus	Passed		Failed	
Slot: Status	slotStatus		loYellow / hiYellow	loRed / hiRed	Associated with slotSensor
Controller: Sw Version	controllerSwVersion		If not equal to current firmware version (1.12.13)		This is to ensure consistency between the Switch firmware and the wci-switch module.

TABLE 9-20 Switch Alarms *(Continued)*

Table: Property	String in Log	Info	Warning	Error	Notes
Controller: Volt 33	controllerVolt33		<96%, >104%	<90%, >110%	Percentages apply to desired value of 3.3 Volts. Reported in units of hundredths of Volts.
Controller: Volt 50	controllerVolt50		<96%, >104%	<90%, >110%	Percentages apply to desired value of 5.0 Volts. Reported in units of hundredths of Volts.
Controller: Volt 120	controllerVolt120		<96%, >104%	<90%, >110%	Percentages apply to desired value of 12.0 Volts. Reported in units of hundredths of Volts.
Controller: Temp	controllerTemp		>65 C	>75 C	Reported in units of hundredths of degrees Celsius.
Controller: Nvram Backup Battery	controllerNvramBackup Battery		Unknown	Low	Should never have a value of unknown
SFL Unit: State	wciState	(4) Reflection mode	(2) Not Tested	(5) Failed	
SFL Unit: Temperature	wciTemp		>65 C	>75 C	Reported in units of hundredths of degrees Celsius.

TABLE 9-20   Switch Alarms *(Continued)*

Table: Property	String in Log	Info	Warning	Error	Notes
SFL Unit: ASIC TempA	wciASICTempA		>95 C	>105 C	Reported in units of hundredths of degrees Celsius.
SFL Unit: ASIC TempB	wciASICTempB		>95 C	>105 C	Reported in units of hundredths of degrees Celsius.
SFL Unit: Volt15	wciVolt15		<96%, >104%	<90%, >110%	Percentages apply to desired value of 1.5 Volts. Reported in units of hundredths of Volts.
SFL Unit: Volt25	wciVolt25		<96%, >104%	<90%, >110%	Percentages apply to desired value of 2.5 Volts. Reported in units of hundredths of Volts.
SFL Unit: PS Volt 25 Status	wciPSVolt25Status			Failed	
SFL Unit: Volt 33	wciVolt33		<96%, >104%	<90%, >110%	Percentages apply to desired value of 3.3 Volts. Reported in units of hundredths of Volts.

TABLE 9-20 Switch Alarms *(Continued)*

Table: Property	String in Log	Info	Warning	Error	Notes
SFL Unit: Volt 50	wciVolt50		<96%, >104%	<90%, >110%	Percentages apply to desired value of 5.0 Volts. Reported in units of hundredths of Volts.
Paroli: State	paroliState			POST failed	
Paroli: Error Rate	paroliErrorRate				
Paroli: Temp	paroliTemp		>65 C	>75 C	Reported in units of hundredths of degrees Celsius.
Paroli: Volt 33	paroliVolt33		<96%, >104%	<90%, >110%	Percentages apply to desired value of 3.3 Volts. Reported in units of hundredths of Volts.
Paroli: Volt 15	paroliVolt15		<96%, >104%	<90%, >110%	Percentages apply to desired value of 1.5 Volts. Reported in units of hundredths of Volts.
Fan: Tray State	fanTrayState		Unknown	Failed	Should already be associated with below conditions.
Fan: Speed0	fanSpeed0		<75% >250%	<70% >300%	

TABLE 9-20   Switch Alarms *(Continued)*

Table: Property	String in Log	Info	Warning	Error	Notes
Fan: Speed1	fanSpeed1		<75% >250%	<70% >300%	
Fan: Speed2	fanSpeed2		<75% >250%	<70% >300%	
Power Supply: State	powerSupplyState		Degraded	Failed	

Monitoring Switch Traps

Sun Fire Link software generates traps to track system availability. All traps are logged. However, not all traps generate alarms directly. Some traps cause refreshes of data tables. When those tables refresh their data, alarms can be triggered. TABLE 9-21 displays switch traps.

TABLE 9-21   Switch Traps

Table: Property	Trap String in Log	Info	Warning	Error	Notes
SFL Switch Event: On	availabilityOn				
SFL Switch Event: Available	availabilityAvail				
SFL Switch Event: Reboot	availabilityReboot		Received		
SFL Switch Event: Shutdown	availabilityWcxShutdown			Received	
SFL Switch Event: Self Test Result Changed	postResultChanged				
SFL Switch Event: Environmental Status Change	environmentalStatusChanged				
SFL Switch Event: Paroli Error Rate Status Change	paroliErrorRateStatusChanged			Received	
SFL Switch Event: Paroli Environmental Shutdown	paroliEnviromentalShutdown			Received	
SFL Switch Event: Paroli Error Rate Shutdown	paroliErrorRateShutdown			Received	

TABLE 9-21 Switch Traps *(Continued)*

Table: Property	Trap String in Log	Info	Warning	Error	Notes
SFL Switch Event: NVRAM Backup Battery Low	nvramBackupBatteryLow				
SFL Switch Event: Community Change	communityChange			Received	Not necessarily an error but requires immediate corrective action to allow continued monitoring of Switch
SFL Switch Event: Log Host Change	logHostChange	Received			
SFL Switch Event: Trap Host Change	trapHostChange			Received	Not necessarily an error but requires immediate corrective action to allow continued monitoring of Switch
SFL Switch Event: Board Arrival	boardArrival				
SFL Switch Event: Board Departure	boardDeparture				

Corrective Actions

TABLE 9-22 and TABLE 9-23 document the cause for each alarm or trap and the corrective action required. The message generated with each alarm or trap indicates the cause for that alarm or trap.

TABLE 9-22   Alarm Causes and Corrective Actions

Table: Property	Alarm String in Log	Cause	Action / Notes
Slot: Test Status	slotTestStatus	POST failure	Look at diag info. in syslog / replace FRU
Slot: Status	slotStatus	Environmental condition	Observe environmental sensor fields. You may need to power down the switch
Controller: Sw Version	controllerSwVersion	Switch firmware version not match what is expected by wci-switch module	Verify version is acceptable and edit alarm attributes in the Sun Management Center console
Controller: Volt 33	controllerVolt33	Board faults / power supplies	If just one board is affected, replace the board. If many boards are affected, you may need to replace power supplies or backplane.
Controller: Volt 50	controllerVolt50	"	"
Controller: Volt 120	controllerVolt120	"	"
Controller: Temperature	controllerTemp	Overheat condition	Power down switch / replace faulty SSC
Controller: Nvram Backup Battery	controllerNvramBackupBattery	NVRAM battery low	Install replacement battery / swap SSC

TABLE 9-22 Alarm Causes and Corrective Actions *(Continued)*

Table: Property	Alarm String in Log	Cause	Action / Notes
SFL Unit: State	wciState	Overheat condition / voltage mismatch / JPOST failure / sunvts in use	Review environmental data in the Sun Management Center console / review error logs / check JPOST results
SFL Unit: Temperature	wciTemp	Overheat condition	Power down switch / replace faulty WCIX module
SFL Unit: ASIC TempA	wciASICTempA	Overheat condition	Power down switch / replace faulty WCIX module
SFL Unit: ASIC TempB	wciASICTempB	Overheat condition	Power down switch / replace faulty WCIX module
SFL Unit: Volt 15	wciVolt15	Board faults / power supplies	If just one board, replace the board. If many boards, may need to replace power supplies / backplane.
SFL Unit: Volt 25	wciVolt25	Board faults / power supplies	If just one board, replace the board. If many boards, may need to replace power supplies / backplane.
SFL Unit: PS Volt 25 Status	wciPSVolt25Status	wcix module's dc-dc converter is faulty (or power may have been turned off by software due to hazardous environmental conditions)	Power down Switch / replace faulty WCIX module

TABLE 9-22 Alarm Causes and Corrective Actions *(Continued)*

Table: Property	Alarm String in Log	Cause	Action / Notes
SFL Unit: Volt 33	wciVolt33	Board faults / power supplies	If just one board, replace the board. If many boards, may need to replace power supplies / backplane.
SFL Unit: Volt 50	wciVolt50	Board faults / power supplies	If just one board, replace the board. If many boards, may need to replace power supplies / backplane.
Paroli: State	paroliState	POST failure	Look at diagnostic info. in syslog / replace FRU
Paroli: Link State	paroliLinkState		
Paroli: Error Rate	paroliErrorRate		
Paroli: Temperature	paroliTemp	Overheat condition	Power down switch / replace faulty Paroli module
Paroli: Volt33	paroliVolt33	Board faults / power supplies	If just one board, replace the board. If many boards, may need to replace power supplies / backplane.
Paroli: Volt 15	paroliVolt15	Board faults / power supplies	If just one board, replace the board. If many boards, may need to replace power supplies / backplane.
Fan:Tray State	fanTrayState	`failed`: failed fan tray `unknown`: missing or failed fan tray	Replace fan tray

TABLE 9-22   Alarm Causes and Corrective Actions *(Continued)*

Table: Property	Alarm String in Log	Cause	Action / Notes
Fan: Speed0	fanSpeed0	Failed fan tray	Replace fan tray
Fan: Speed1	fanSpeed1	Failed fan tray	Replace fan tray
Fan: Speed2	fanSpeed2	Failed fan tray	Replace fan tray
Power Supply: State	powerSupplyState	Power supply failure	Replace power supply

TABLE 9-23   Trap Causes and Corrective Actions

Trap	Trap String in Log	Cause	Action / Notes
SFL Switch Events: On	availabilityOn	Switch booted & Network adapter initialized	
SFL Switch Events: Available	availabilityAvail	POST complete on all devices installed at boot time.	
SFL Switch Events: Reboot	availabilityReboot	About to reboot	
SFL Switch Event: Shutdown	availabilityWcxShutdown	Environmental conditions lead to shutdown.	Power down switch
SFL Switch Event: Self-test Result Change	postResultChanged	Sent to indicate a a self-test result change. It may indicate POST failure if error-level alarm has been sent.	Look at diagnostic information in syslog. If necessary, replace FRU
SFL Switch Event: Environmental Status Change	environmentalStatusChanged	Sent when the status of an environmental sensor changes (for example, from OK to Warning).	Determine what caused the trap. You may need to power down the switch.
SFL Switch Event: Paroli Error Rate Status Change	paroliErrorRateStatusChanged	Sent when the number of errors encountered on a particular link exceeds the warning threshold or the shutdown limit threshold.	

TABLE 9-23   Trap Causes and Corrective Actions *(Continued)*

Trap	Trap String in Log	Cause	Action / Notes
SFL Switch Event: Paroli Environmental Shutdown	paroliEnviromentalShutdown	Sent when a link has been shut down due to conditions detected by an environmental sensor.	Laser was turned off. If condition persists or worsens, you may need to power down the switch.
SFL Switch Event: Paroli Error Rate Shutdown	paroliErrorRateShutdown	Sent when a link has been shut down due to a high error rate.	
SFL Switch Event (Controller): nvram Backup Battery Low	nvramBackupBatteryLow	System controller NVRAM battery low.	Replace battery on SSC's NVRAM chip. NVRAM contents will be lost if power is removed from SSC. In that case, configuration will have to be restored.
SFL Switch Event: SNMP Community String Changed	communityChange	Sent when the SNMP community string has been changed on the switch.	In order to keep SNMP communication viable, you must run es-setup to remove the instance of the switch agent module that is monitoring this switch. Then run es-setup again to create a new instance of the module for monitoring this switch and specify the new SNMP community string in response to the prompts.

TABLE 9-23 Trap Causes and Corrective Actions *(Continued)*

Trap	Trap String in Log	Cause	Action / Notes
SFL Switch Event: Log Host Change	logHostChange	Sent when the name of the host that provides a secondary log of switch events and errors has been changed. This does not affect the Sun Management Center software.	These are Switch logs that are NOT the event logs generated by the wci-switch module
SFL Switch Event: SNMP Trap Host Change	trapHostChange	Sent when the name of the SNMP trap host has been changed on the switch.	This is a critical error, preventing any switch traps from reaching the Sun Management Center agent. The name of the SNMP trap host specified in the switch must match the name of the Sun Management Center server specified for the Sun Management Center agent that is monitoring the switch(es). Note: To correct this condition, run `es-setup` and change the name of the Sun Management Center server associated with the agent.
SFL Switch Event: Board Arrival	boardArrival	Board arrival or CLI command enabling the board	
SFL Switch Event: Board Departure	boardDeparture	Board departure or CLI command disabling the board	

Techniques for Monitoring Cluster Link Status

This chapter explains how to monitor link status in a Sun Fire Link cluster and how to collect cluster data that will be useful in diagnosing link failures. It is organized into the following topics:

- "Overview" on page 129
- "Identifying Link Failures During Cluster Operation" on page 131
- "Interpreting Link Status in POST Output" on page 152
- "Collecting Troubleshooting Information for Sun Services" on page 157

Overview

A link can be in any one of six states, which are described in TABLE 10-1.

TABLE 10-1   Link State Summary

State	Meaning
Link Up	The link is able to transmit and receive data.
Link Down	The optical assembly is present, but the lasers are not turned on.
Link Not There	The optical assembly is not present for this link location.
Wait Up for SC Takedown	The link is waiting to be taken down by the SC (Sun Fire system) or SSC (switch).
Wait Up for SC Link Up	The link is waiting to be brought up by the SC (Sun Fire system) or SSC (switch).
SC Error Wait Up for Link	The link is in the Link Down state as the result of an error and is waiting for the SC (Sun Fire system) or SSC (switch).

When the FM determines that a Sun Fire Link optical interface is present, but its lasers are not turned on, it places the link in the *Link Down* state. This may result from an intentional action by an operator or from a hardware or software failure. In either case, a Link Down message is sent to the system console and is stored in a log file. If you are using the Sun Management Center graphical user interface (GUI) to monitor the Sun Fire Link cluster, the Link Down status will also cause the link representation on the Fabric Details display to become red.

This chapter explains how to recognize Link Down error messages and determine which physical components are involved. This determination will focus on the basic components that make up a link between two physically connected endpoints. Using FIGURE 10-1 to illustrate this point, you will be able to determine that the following physical components are implicated in the failed link:

- Link 0 in assembly IB8 of system SF6800-1
- Link 0 in assembly IB8 of system SF6800-2
- The optical cable connecting the two links
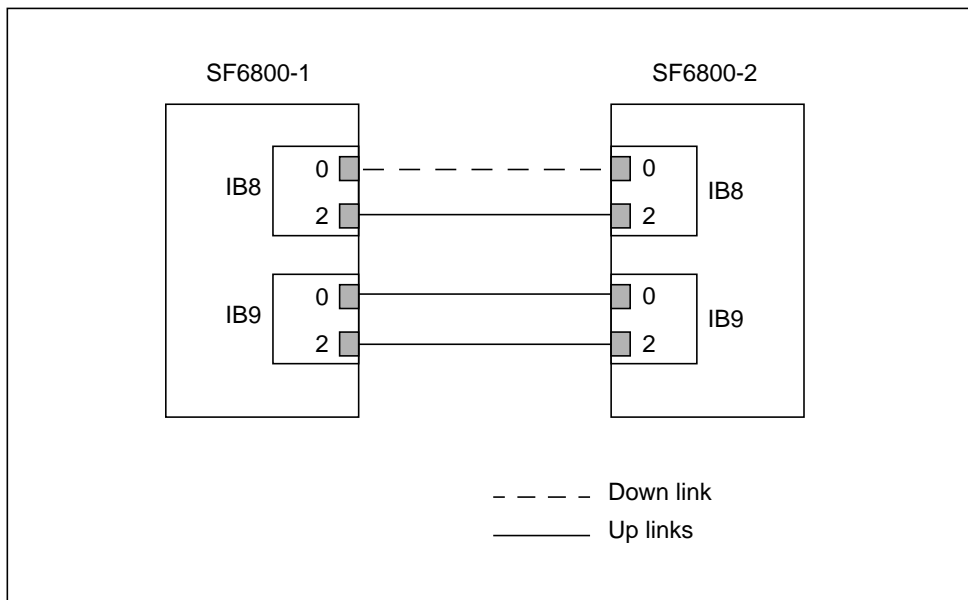
FIGURE 10-1 Two-Node, Direct-Connect Cluster With One Link Down and Three Links Up

This illustration would vary slightly if one or more of the cluster nodes were a Sun Fire 15K/12K system or if the cluster included Sun Fire Link switches. In either case, however, the set of components identified would be the same: local optical link endpoint, cable, and remote optical link endpoint. Only the identities of the hardware assemblies containing the endpoints would be affected by chassis type.

Note – In a direct-connect (switchless) Sun Fire Link cluster, a link failure is detected and reported by both cluster nodes connected by the failed path—in FIGURE 10-1, that would be SF6800-1 and SF6800-2. When a compute node is connected to a Sun Fire Link switch, the Link Down message is detected and reported by the compute node. The switch also generates an error message indicating a Link Down condition.

It will be the task of the Sun field support personnel to isolate the failure below the link-cable-link level. This chapter explains how to collect diagnostic information that will help the field support staff locate the specific failing component.

If a Link Down condition occurs on one or more links in a group of striped links, communication will continue over the remaining links in the stripe group so long as one member of the striped group remains healthy. If a link fails while you are not directly monitoring network status, continued operation of the Sun Fire Link cluster may keep you from noticing that a link is down. This chapter describes two methods you can implement to automatically notify you when a link goes down.

Note – You will find the information in this chapter most useful if you have a chart that lists all the physical connections in the Sun Fire Link network. This chart may already have been created when the Sun Fire Link hardware was installed. The *Sun Fire Link Hardware Installation Guide* includes a recommendation that such a chart be created when the Sun Fire Link network is installed

Identifying Link Failures During Cluster Operation

This section explains how to identify Link Down error conditions that occur while a Sun Fire Link cluster is in operation. This discussion is organized as follows:

- How to use the Sun Management Center/FM console to monitor link status and identify Link Down conditions.
- How to use the WRSM command-line interface tools to monitor link status at the cluster node level.
- How to use the FM command-line interface tools to monitor link status at the cluster-wide level.
- How to interpret Link Down messages generated by Sun Fire 6800 systems, Sun Fire 15K/12K systems, and Sun Fire Link switches.

Using the Sun Management Center/FM Console to Monitor Link Status

Whenever possible, you should use the Sun Management Center/FM console as your primary interface for managing and monitoring Sun Fire Link clusters. Chapter 9 of this manual discusses a broad range of Sun Management Center/FM console features that you can use for this purpose. This section focuses on using the Sun Management Center/FM console to access link status details.

For the following procedure, assume that a Sun Fire Link cluster is in operation, the Sun Management Center console is running, and that one link in a cluster partition goes down.

1. **If it is not already being displayed, bring up the Fabric Details window. If you want instructions for this step, see "Displaying the Fabric Console" on page 71.**

2. **Click on the Fabric Information folder in the left pane.**

   This will cause the following tables to be displayed. See "Monitoring Your Fabric Through System Details" on page 93 for an illustration of these tables.

   - Partition Table
   - Member Table
   - Link Table

3. **Examine the Link Table, which contains a comprehensive description of all the links in the cluster.**

   Each row in the Link Table describes a single link, including descriptions of both the local and remote endpoints for the link. These link attributes are explained in "Fabric Information Tables" on page 95.

Using the WRSM CLI to Monitor Link Status at the Node Level

This section explains how to collect link status information on each node in a cluster. It illustrates the use of `wrsmstat wrsm -v` and `wrsmstat route` on a sample four-node cluster that has a single failed link.

In this example, the cluster nodes are `sys_a`, `sys_b`, `sys_c`, and `sys_d`. The failed link is link 2 in IB9 on `sys_b`. To locate this faulty link, take the following steps:

- Run `wrsmstat wrsm -v` on each node in the cluster. This shows the state of the links on the node. This is first illustrated in CODE EXAMPLE 10-1.

- Run `wrsmstat route` on one node. This displays a global view of each link. See CODE EXAMPLE 10-2 for an illustration.

A brief explanation of the `wrsmstat wrsm -v` output follows CODE EXAMPLE 10-1.

CODE EXAMPLE 10-1   Results From Initial Running of `wrsmstat wrsm -v`

```
sys_a# wrsmstat wrsm -v

WCI instance: 1
-------------
Port ID:                     29
Controller ID:               1
Config Version:              1
Link Error Shutdown Trigger: 1000
Link 0
        Link enabled:        yes
        Link State:          up
        Physical Link State: in use
        Laser Enabled:       yes
        Transmit Enabled:    yes
        Remote RSM HW addr:  0
        Remote wnode ID:     0
        Remote link num:     0
        Remote WCI port ID:  0
        Error takedowns:     0
        Disconnected takedowns: 0
        Bad Config takedowns: 0
        Failed bringups:     0
        Total link errors:   0
        Maximum link errors: 0
        Average link errors: 0
        Auto shutdown enabled: yes
Link 1 is not present.
Link 2
        Link enabled:        yes
        Link State:          up
        Physical Link State: in use
        Laser Enabled:       yes
        Transmit Enabled:    yes
        Remote RSM HW addr:  0
        Remote wnode ID:     0
        Remote link num:     0
        Remote WCI port ID:  0
        Error takedowns:     0
        Disconnected takedowns: 0
        Bad Config takedowns: 0
        Failed bringups:     0
        Total link errors:   0
        Maximum link errors: 0
        Average link errors: 0
```

```
        Auto shutdown enabled:  yes
Cluster Error Count:            0
Uncorrectable SRAM ECC error:   no
Maximum SRAM ECC errors:        0
Average SRAM ECC errors:        0


WCI instance: 2
-------------
Port ID:                        31
Controller ID:                  1
Config Version:                 1
Link Error Shutdown Trigger:    1000
Link 0
        Link enabled:           yes
        Link State:             up
        Physical Link State:    in use
        Laser Enabled:          yes
        Transmit Enabled:       yes
        Remote RSM HW addr:     0
        Remote wnode ID:        0
        Remote link num:        1
        Remote WCI port ID:     0
        Error takedowns:        0
        Disconnected takedowns: 0
        Bad Config takedowns:   0
        Failed bringups:        0
        Total link errors:      0
        Maximum link errors:    0
        Average link errors:    0
        Auto shutdown enabled:  yes
Link 1 is not present.
Link 2
        Link enabled:           yes
        Link State:             up
        Physical Link State:    in use
        Laser Enabled:          yes
        Transmit Enabled:       yes
        Remote RSM HW addr:     0
        Remote wnode ID:        0
        Remote link num:        1
        Remote WCI port ID:     0
        Error takedowns:        0
        Disconnected takedowns: 0
        Bad Config takedowns:   0
        Failed bringups:        0
        Total link errors:      0
```

```
       Maximum link errors:    0
       Average link errors:    0
       Auto shutdown enabled:  yes
Cluster Error Count:           41
Uncorrectable SRAM ECC error:  no
Maximum SRAM ECC errors:       0
Average SRAM ECC errors:       0
```

The following is a summary analysis of the output shown in CODE EXAMPLE 10-1:

■ There are no indications of link errors for links 0 and 2 in either IB8 or IB9 on `sys_a`.

■ The only indication of any errors being detected is found in the `Cluster Error Count` line for IB9, which reports 41 errors. This is the shown on the fouth line from the end of the sample output.

The next step is to run `wrsmstat route` on `sys_a` to get a profile of the routes that connect `sys_a` to other nodes and switches. CODE EXAMPLE 10-2 shows the output generated by this step. An explanation of the output follows the example.

CODE EXAMPLE 10-2 Results From Running `wrsmstat route`

```
sys_a# wrsmstat route

Controller 1 - Route to sys_a
------------------------
Config Version:              1
FM node id:                  0x830740c6
RSM hardware addr:           0
Route Changes:               0
Route Type:                  Multihop
Number of WCIs:              1
Stripes:                     0
WCI #0
       Port ID:              31
       Instance :            2
       Number of hops:       0
       Number of links:      0


Controller 1 - Route to sys_b
------------------------
Config Version:              1
FM node id:                  0x83073cdc
RSM hardware addr:           1
Route Changes:               0
Route Type:                  Multihop
Number of WCIs:              2
```

```
Stripes:                       3
WCI #0
        Port ID:               29
        Instance :             1
        Number of hops:        1
        Number of links:       2
                Link #0, is a switch, leading to RSM HW addr: 0x1
                Link #2, is a switch, leading to RSM HW addr: 0x1
WCI #1
        Port ID:               31
        Instance :             2
        Number of hops:        1
        Number of links:       1
                Link #0, is a switch, leading to RSM HW addr: 0x1

Controller 1 - Route to sys_c
-----------------------
Config Version:                1
FM node id:                    0x8308e1a8
RSM hardware addr:             2
Route Changes:                 0
Route Type:                    Multihop
Number of WCIs:                2
Stripes:                       4
WCI #0
        Port ID:               29
        Instance :             1
        Number of hops:        1
        Number of links:       2
                Link #0, is a switch, leading to RSM HW addr: 0x2
                Link #2, is a switch, leading to RSM HW addr: 0x2
WCI #1
        Port ID:               31
        Instance :             2
        Number of hops:        1
        Number of links:       2
                Link #0, is a switch, leading to RSM HW addr: 0x2
                Link #2, is a switch, leading to RSM HW addr: 0x2

Controller 1 - Route to sys_d
-----------------------
Config Version:                1
FM node id:                    0x8308eea4
RSM hardware addr:             3
Route Changes:                 0
Route Type:                    Multihop
Number of WCIs:                2
```

```
 Stripes:                          4
 WCI #0
       Port ID:                29
       Instance :              1
       Number of hops:         1
       Number of links:        2
               Link #0, is a switch, leading to RSM HW addr: 0x3
               Link #2, is a switch, leading to RSM HW addr: 0x3
 WCI #1
       Port ID:                31
       Instance:               2
       Number of hops:         1
       Number of links:        2
               Link #0, is a switch, leading to RSM HW addr: 0x3
               Link #2, is a switch, leading to RSM HW addr: 0x3
```

This output shows that `sys_a` has a four-link route to `sys_c` and a four-link route to `sys_d`. But the route to `sys_b` consists of only three links, as follows:

■ IB8, link 0 and link 2

■ IB9, link 0 only

The next step is to log into `sys_b` and run `wrsmstat wrsm −v` to get a local view of IB9, link 2. This is shown in CODE EXAMPLE 10-3. The output is explained following the example.

CODE EXAMPLE 10-3   `wrsmstat wrsm −v` Output From Node With Failed Link

```
 sys_b# wrsmstat wrsm −v

 WCI instance: 1
 -------------
 Port ID:                    29
 Controller ID:              1
 Config Version:             1
 Link Error Shutdown Trigger:  1000
 Link 0
       Link enabled:         yes
       Link State:           up
       Physical Link State:  in use
       Laser Enabled:        yes
       Transmit Enabled:     yes
       Remote RSM HW addr:   0
       Remote wnode ID:      0
       Remote link num:      2
       Remote WCI port ID:   0
       Error takedowns:      0
```

```
        Disconnected takedowns: 0
        Bad Config takedowns:   0
        Failed bringups:        0
        Total link errors:      0
        Maximum link errors:    0
        Average link errors:    0
        Auto shutdown enabled:  yes
Link 1 is not present.
Link 2
        Link enabled:           yes
        Link State:             up
        Physical Link State:    in use
        Laser Enabled:          yes
        Transmit Enabled:       yes
        Remote RSM HW addr:     0
        Remote wnode ID:        0
        Remote link num:        2
        Remote WCI port ID:     0
        Error takedowns:        0
        Disconnected takedowns: 0
        Bad Config takedowns:   0
        Failed bringups:        0
        Total link errors:      0
        Maximum link errors:    0
        Average link errors:    0
        Auto shutdown enabled:  yes
Cluster Error Count:            0
Uncorrectable SRAM ECC error:  no
Maximum SRAM ECC errors:        0
Average SRAM ECC errors:        0


WCI instance: 2
-------------
Port ID:                        31
Controller ID:                  1
Config Version:                 1
Link Error Shutdown Trigger:    1000
Link 0
        Link enabled:           yes
        Link State:             up
        Physical Link State:    in use
        Laser Enabled:          yes
        Transmit Enabled:       yes
        Remote RSM HW addr:     0

        Remote wnode ID:        0
```

```
        Remote link num:        3
        Remote WCI port ID:     0
        Error takedowns:        0
        Disconnected takedowns: 0
        Bad Config takedowns:   0
        Failed bringups:        0
        Total link errors:      0
        Maximum link errors:    0
        Average link errors:    0
        Auto shutdown enabled:  yes
Link 1 is not present.
Link 2
        Link enabled:           yes
        Link State:             wait (up)
        Physical Link State:    seek
        Laser Enabled:          yes
        Transmit Enabled:       no
        Remote RSM HW addr:     0
        Remote wnode ID:        0
        Remote link num:        3
        Remote WCI port ID:     0
        Error takedowns:        0
        Disconnected takedowns: 0
        Bad Config takedowns:   0
        Failed bringups:        2128
        Total link errors:      0
        Maximum link errors:    0
        Average link errors:    0
        Auto shutdown enabled:  no
Cluster Error Count:            19
Uncorrectable SRAM ECC error:   no
Maximum SRAM ECC errors:        0
Average SRAM ECC errors:        0
```

The IB9 status is listed under the heading `WCI instance 2`. The following information tells the story for link 2:

■ The link and laser are enabled, which is as it should be for a link that is in use.

■ However, the Link State status is `wait (up)` and the `Transmit Enabled` state is `no`.

This indicates that either the cable connected to link 2 in IB9 or the Sun Fire Link board on the other end of the connection is faulty. In this example, the other end of the cable is connected to a switch.

Using the FM CLI to Monitor Link Status

The following examples show how to use the `wcfmstat` and `wcfmver` commands to investigate link status across the full Sun Fire Link cluster. An explanation of the command output is provided at the end of each example. See Appendix A of the *Sun Fire Link Fabric Administrator's Guide* for more information about these commands.

Execute these commands as root on the FM Management Station (also called the FM host).

============= EXAMPLE 1 =================

The following CODE EXAMPLE 10-4 reports the status of all members of a fabric named fabric1.

CODE EXAMPLE 10-4 Sample Output From `wcfmstat` *fabricname*

```
  fm_host# cd /opt/SUNWwcfm/bin
  fm_host# ./wcfmstat fabric1
Partition                        Stripe Level
Name     Type      Topology       WCI      Link
part1    RSM       WCIX_SWITCH     1        2
part2    RSM       WCIX_SWITCH     1        2
All members of this fabric are assigned to partitions.
```

The following explains how to interpret the output shown in CODE EXAMPLE 10-4:

- `fabric1` contains two partitions, named `part1` and `part2`.
- The partition Type designation is always RSM.
- `WCIX_SWITCH` indicates that each partition uses a switch-based topology.
- Both partitions use the same striping policy:
  - WCI stripe level = 1. No striping across the Sun Fire Link assemblies.
  - Link stripe level = 2. Striping across the two optical ports in each Sun Fire Link assembly.

============= EXAMPLE 2 =================

This example reports detailed status of the partition named `part1`.

CODE EXAMPLE 10-5 Output From `wcfmstat` With Partition Argument

```
  fm_host# cd /opt/SUNWwcfm/bin
  fm_host# ./wcfmstat -p part1 fabric1
Partition Name: part1
Members:
  sys_a_sc1-A
  sys_b_sc1-A
```

```
    link-sw1-null
    link-sw2-null
  ----------------------------------------
Routes:
  sys_a_sc1:A=sys_b_sc1:A
    sys_a_sc1:A:8:1:0=link-sw2::::3=0
    sys_a_sc1:A:8:1:2=link-sw1::::3=0
    sys_b_sc1:A:8:1:0=link-sw2::::1=0
    sys_b_sc1:A:8:1:2=link-sw1::::1=0
  ----------------------------------------
Links:
  sys_a_sc1:A:8:1:0=link-sw2::::3=0
  sys_a_sc1:A:8:1:2=link-sw1::::3=0
  sys_b_sc1:A:8:1:0=link-sw2::::1=0
  sys_b_sc1:A:8:1:2=link-sw1::::1=0
```

The following explains how to interpret the output shown in CODE EXAMPLE 10-5:

■ The `Members` section shows that `part1` contains two nodes, which are named `sys_a_sc1-A` and `sys_b_sc1-A`.

■ The `Members` section also shows that `part1` contains two switches, which are named `link-sw1` and `link-sw2`.

Note: Switch names do not include domain extensions.

■ The `Routes` section identifies the two endpoints for a route by which the nodes in the partition can communicate. In this example, the route endpoints are domain `A` of node `sys_a_sc1` and domain `A` of node `sys_b_sc1`.

■ The `Routes` section also lists the links that make up the route. A Sun Fire Link route is an aggregate path containing all the individual links that connect the route's two endpoints. Because the configuration in this example includes switches, each link has one node endpoint and one switch endpoint.

For example, the first link listed is `sys_a_sc1:A:8:1:0=link-sw2::::3=0`. The meaning of this string is explained below:

■ `sys_a_sc1:A` – This indicates domain `A` of node `sys_a_sc1`.

■ `8:1:0` – In this triplet, the value `8` represents the Sun Fire Link assembly in chassis slot IB8, the value `1` represents the ASIC ID 1, and the value `0` represents the link number (optical port) for that ASIC.

  Note: In a Sun Fire Link assembly, the Sun Fire Link ASIC ID is always 1.

■ `link-sw2:3` – This represents the switch host name and link number (optical port) 3.

■ `=0` – This indicates the status of the link. `0` indicates that the link is up and `-1` indicates that the link is down. In this example, this link is up.

■ The `Links` section lists the links contained in the partition. This is the same list that is provided in the `Routes` section, except the associated route is not shown.

============= EXAMPLE 3=================

This example performs a link discovery operation and reports on all the links that are found. When using this command, you should have a chart that lists all the physical connections in the Sun Fire Link network, including the endpoint details as shown in the CODE EXAMPLE 10-6.

Note – This chart may already exist. The *Sun Fire Link Hardware Installation Guide* includes a recommendation that such a chart be created when the Sun Fire Link network is installed.

If you have the physical link chart, you can compare the set of discovered links generated by the `wcfmver` command with the chart list. This will tell you whether any links are down and, if so, which.

CODE EXAMPLE 10-6   Output From `wcfmver` With Nodes Argument

```
fm_host# cd /opt/SUNWwcfm/bin
fm_host# ./wcfmver -n sys_a_sc1:a sys_b_sc1:a link-sw1 link-sw2 fabric1
Node = sys_a_sc1:a
Node = sys_b_sc1:a
Node = link-sw1
Node = link-sw2
Starting discovery on:
tools.wcfmver.WildcatFMVer@1c88f9e
Estimated wait time = 180 sec.
....................................................................
....................................................................
....................................................................

The following links are in the configuration but were not discovered...

The following links were discovered but are not in the configuration...

sys_b_sc1:A:8:1:2=sys_a_sc1:A:8:1:2=-1
sys_b_sc1:A:8:1:0=sys_a_sc1:A:8:1:0=-1
sys_b_sc1:A:9:1:2=link-sw1::::0=-1
sys_b_sc1:A:9:1:0=link-sw2::::0=-1
sys_a_sc1:A:9:1:2=link-sw1::::1=-1
sys_a_sc1:A:9:1:0=link-sw2::::1=-1
```

The following explains how to interpret the output shown in CODE EXAMPLE 10-6:

- The first four lines list the nodes and switches involved in the link discovery.
- The next three lines indicate that the discovery process is being launched and the expected time to completion.
- The first section in the discovery report lists any links that are in the partition, but were not discovered. This example does not show any links in this category.
- The final report section lists links that were discovered, but are not in the configuration. In this example, six links are reported in this category.

Using the `showlinks` Command to Monitor Link Status on Sun Fire Link Switches

The `showlinks` command reports the status, including errors, of the links in the switch where the command is run. Log into the switch and execute `showlinks` at the SSC prompt. TABLE 10-2 shows the `showlinks` command syntax.

TABLE 10-2   Command Syntax for `showlinks`

`showlinks`	Displays both link status and error statistics.
`showlinks -l`	Displays link status only.
`showlinks -e`	Displays error statistics only.
`showlinks [-e | -l] loop [time]`	Repeats the showlinks operation every *time* seconds until you press Enter. The repetition interval default is 1 second.

CODE EXAMPLE 10-7 illustrates the use of `showlinks` on the switch `wcsw1`. In this example, only link 0 is `up`. All others are in the `waitup` state, which means they are configured in a partition, but are not currently being used to carry data. This may or may not indicate that a problem exists. The next step would be to identify the other endpoints to which these links are connected and investigate the status of the remote link endpoints.

CODE EXAMPLE 10-7   Example of `showlinks` Output

```
wcsw1:SSC> showlinks

======================================
                LINKS
======================================
Link      State
----    -----------
  0     up
```

```
  1     waitup
  2     waitup
  3     waitup
  4     waitup


   =====================================
                 ERRORS
   =====================================
  Link   Current  Average  Total   Num of intervals  Min/interval
  ----   -------  -------  -------  ----------------  ------------
   0         0        0        0                 0        60
   1         0        0        0                 0        60
   2         0        0        0                 0        60
   3         0        0        0                 0        60
   4         0        0        0                 0        60
```

Interpreting Link Down Messages

This section explains how to interpret link status information contained in message logs and/or messages sent to the console.

Link Down Messages on Sun Fire 6800 Systems

When examining `/var/adm/messages` for Link Down messages in a Sun Fire 6800 domain, look for the string "`link` *n* `down`", where *n* is either `0` or `2`.

The text in CODE EXAMPLE 10-8 shows typical Link Down message content for a Sun Fire 6800 domain. The message indicates that link 0 on IB8 (wci 29) of cluster node `sf6800-1-a` went down on September 16, 9:17:44.

CODE EXAMPLE 10-8   Link Down Message for IB8, Port 0 in a Sun Fire 6800 Domain

```
Sep 16 9:17:44 sf6800-1-a wrsm: [ID 220426 kern.notice] NOTICE:
wci 29 link 0 down: hardware-shutdown
```

If the link were in I/O slot IB9, the `wci` ID would be 31. An example of this is shown in CODE EXAMPLE 10-9, which shows port 2 as the failed link.

CODE EXAMPLE 10-9   Link Down Message for IB9, Port 2 in a Sun Fire 6800 Domain

```
Sep 16 9:22:34 sf6800-1-a wrsm: [ID 311464 kern.notice] NOTICE:
wci 31 link 2 down: hardware-shutdown
```

Note – Sun Fire Link I/O assemblies IB8 and IB9 are always assigned wci IDs 29 and 31, respectively.

In a Sun Fire 6800 system, link 0 is on the Paroli board installed in slot 2 of a Sun Fire Link assembly. Link 2 is on the Paroli board installed in slot 1. Refer to FIGURE 10-2 to see the physical locations of the wci *n* and link *n* values. Refer to the *Sun Fire Link System Overview* and the *Sun Fire Link Hardware Installation Guide* for additional details about the physical placement of Sun Fire Link components in Sun Fire system chassis.

When a link comes up, the transition is reported with a message that contains the string "link *n* up". See CODE EXAMPLE 10-10. In this example, the link that had been down in CODE EXAMPLE 10-8 came back up at 10:02:25.

CODE EXAMPLE 10-10 Link Up Message for IB8, Port 0 in a Sun Fire 6800 Domain

```
Sep 16 10:02:25 sf6800-1-a wrsm: [ID 881732 kern.notice] NOTICE:
wci 29 link 0 up
```
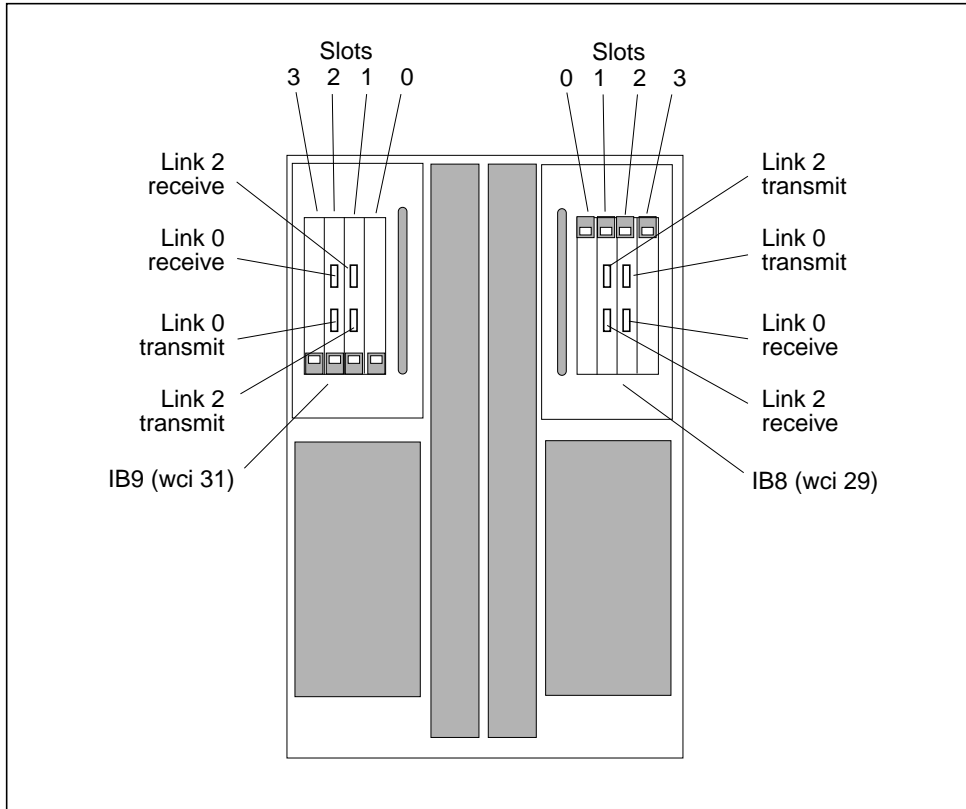
FIGURE 10-2 Sun Fire Link Optical Port Locations in a Sun Fire 6800 Chassis

Link Down Messages on Sun Fire 15K/12K Systems

The Link Down messages reported by a Sun Fire 15K/12K are only slightly different from messages reported on Sun Fire 6800 domains. CODE EXAMPLE 10-11 shows an example of the Link Down message content reported for a Sun Fire 15K domain.

CODE EXAMPLE 10-11 Link Down Message for a Sun Fire 15K Domain

```
Sep 16 8:18:23 sf15k-1-b wrsm: [ID 125775 kern.notice] NOTICE:
wci 61 link 0 down: hardware-shutdown
```

In this example, the failed link is optical port 0 in wci 61. These Sun Fire Link resources are allocated to domain B of SC sf15k-1.

In a Sun Fire 15K/12K system, the wci ID values begin at 29 and increase at intervals of 32 for each of the other I/O slots in the chassis. See FIGURE 10-3 for an illustration of the wci and I/O slot relationships for a Sun Fire 15K/12K chassis.
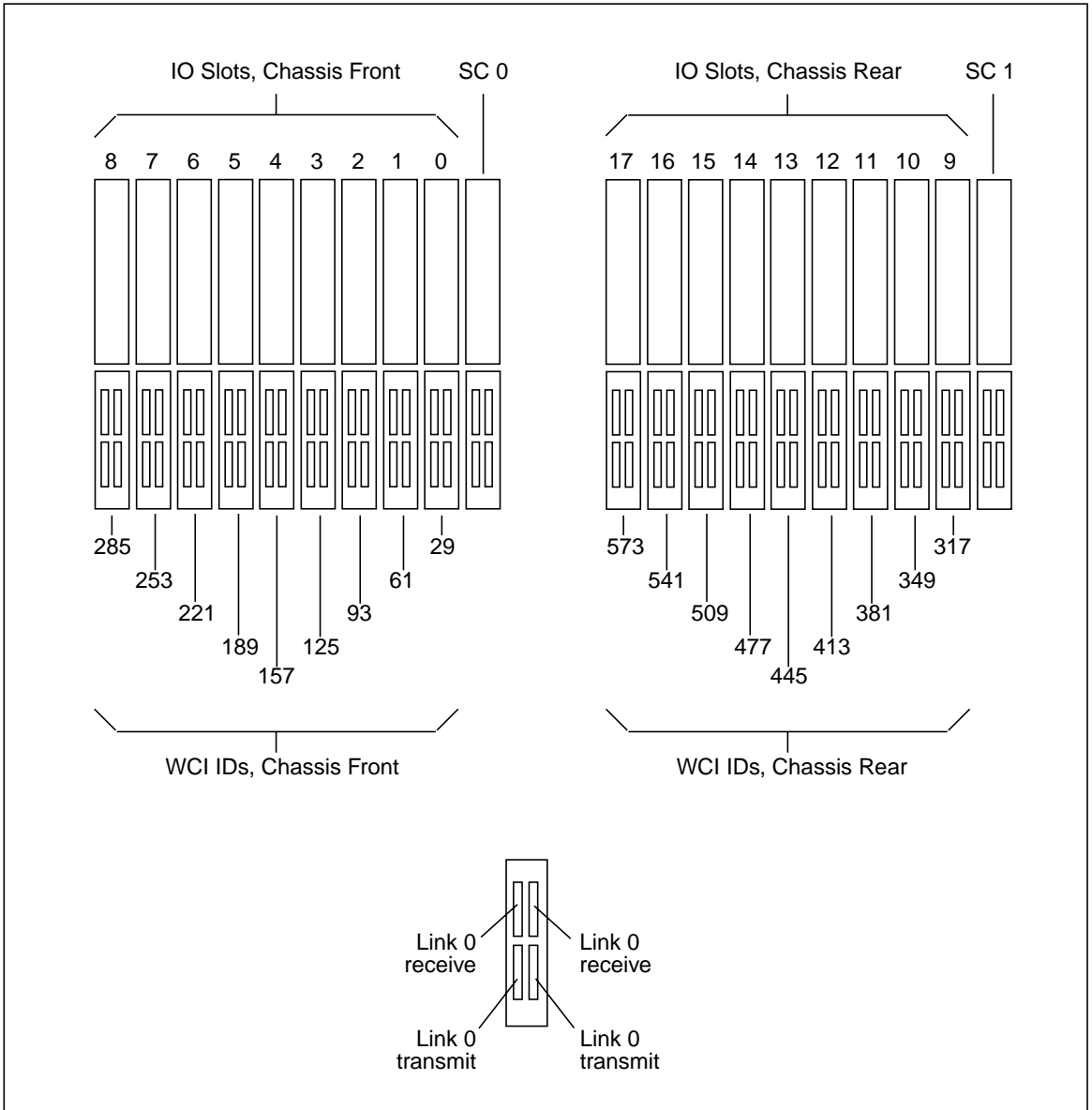
FIGURE 10-3 Sun Fire Link Optical Port Locations in a Sun Fire 15K/12K Chassis

Link Down Messages on Sun Fire Link Switches

When a Sun Fire Link switch experiences a Link Down condition, it generates a message like the sample message shown in CODE EXAMPLE 10-12.

CODE EXAMPLE 10-12 Link Down Message for a Sun Fire Link Switch

```
Sep 16 11:06:13 switch2-1 wrsm: [ID 546774 kern.notice] NOTICE:
wci 1 link 4 down: hardware-shutdown
```

The following summarizes the main characteristics that distinguish a switch Link Down message from those generated by Sun Fire systems:

- No domain is indicated in the host name field because switches are not Solaris hosts and do not have domains.

- The wci ID for a switch will be always be 1.

- A switch can contain up to eight Paroli ports. Consequently, the link value will be an integer in the range 0–7.

In the sample message in CODE EXAMPLE 10-12, the link located in port 4 of switch switch2-1 went down on September 16, 11:06:13. FIGURE 10-4 illustrates the physical layout of the optical ports in a Sun Fire Link switch chassis.
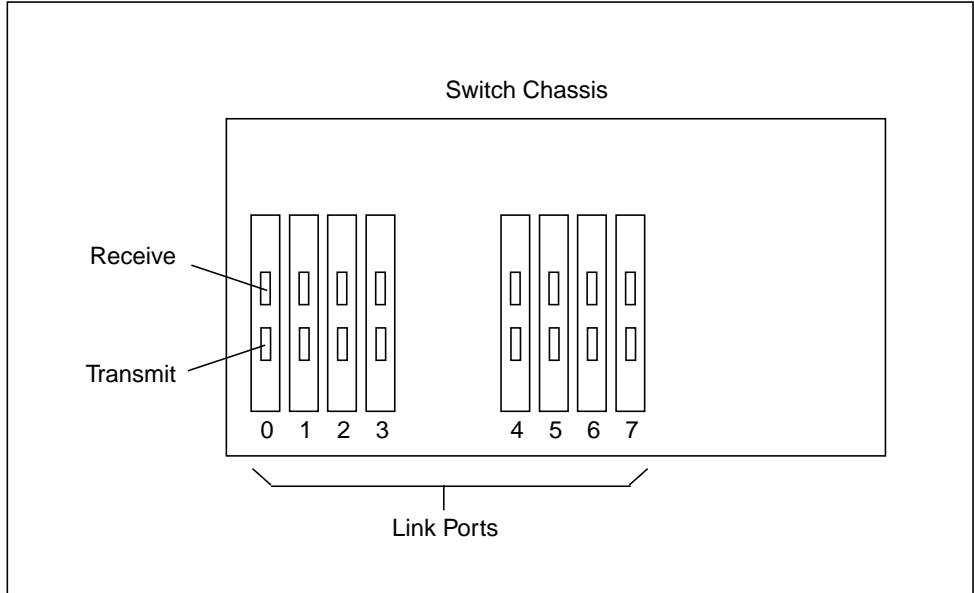
FIGURE 10-4 Optical Port Locations in a Sun Fire Link Switch

Automatic Notification of Link State Changes

This section describes two methods for automatically generating an email message whenever a link changes state on a cluster node.

- One method uses an existing feature that is part of the Sun Management Center alarm handler.
- The second method involves creating a shell script that polls the link states on each node and sends email whenever a link changes state.

Sun Management Center Alarm Handler Method

The WRSM agent running on each cluster node monitors the health of the local cluster components by examining data generated by a local instance of `kstat`. The WRSM agent sends an alarm signal to the Sun Management Center server when a monitored attribute experiences a specified alarm-trigger event.

You can use the Sun Management Center Attribute Editor to determine which cluster attributes will be capable of triggering an alarm and what threshold values will serve as trigger points. See the Attribute Editor discussion in the *Sun Management Center Software User's Guide*.

For most alarm conditions, the Sun Management Center console simply displays a color-coded alarm medallion. The medallion is linked to information about the error, which can be displayed by following the link path down to a table that contains detailed information about the component that was the source of the alarm.

In addition to displaying alarm medallions or other indications of attribute status, Sun Management Center can be set up to send email to one or more recipients when a particular alarm is detected. You can instruct the Sun Management Center software to send an email message whenever a link changes state. In brief, the steps for doing this are:

1. **Create a script that causes email to be sent to you or others when the script runs. Store the script in** `/var/opt/SUNWsymon/bin`**.**

2. **Go to the Fabric Details window and open the Fabric Information folder in the left pane.**

   This displays a window containing the Partition Table, Member Table, and Link Table.

3. **In the Link Table, select the data property cell that you want to be enabled for triggering email.**

   - To send email when any link changes state, right click in the cell at the top of the link state column.

- To send email when particular links change state, right click in the individual link state cell belonging to each link of interest.

Either operation will open the Attribute Editor window.

4. **Click on the Attribute Editor window's Alarms tab.**

5. **Enter the appropriate alarm threshold values and click on the Apply button.**

6. **Click on the Actions tab.**

7. **Click on the Actions... button next to Critical Action. This registers the action.**

8. **Enter the name of the email launching script in the Critical Action box and set the Automatic option to Yes.**

9. **Click on OK to accept the action and then close the Actions dialog.**

Note – As an alternative to writing your own email-launching script, you can use an email facility that is built into the Sun Management Center Kernel Reader module. This method is described in the Attribute Editor discussion in the *Sun Management Center Software User's Guide.*

Alternate Method for Alarm Notification

If the Sun Management Center software is not available, you may want to use your own error monitoring tool to conduct searches in /var/adm/messages for keywords contained in the link state messages described earlier.

Alternatively, you can set up a shell script that will poll the link state on each node and send email when a link changes state. An example of such a script is shown in CODE EXAMPLE 10-13. An example of the email content that this sample script would send is shown in CODE EXAMPLE 10-14. This script also sends messages to the console, which is illustrated in CODE EXAMPLE 10-15.

Note – The script in CODE EXAMPLE 10-13 is not a supported product of Sun Microsystems, Inc. It is presented here simply as a conceptual illustration.

CODE EXAMPLE 10-13 Sample Shell Script for Automatic Notification of Link State Changes

```
#!/bin/sh
# Example of Fire Link Alarm Mailer script for Cluster Nodes.
# Sept-17-2002
# This script polls the Fire Link cluster node link states
# and send alarm (mail)  to registered users (mail list)
# when a link goes down or up.
```

CODE EXAMPLE 10-13 Sample Shell Script for Automatic Notification of Link State Changes *(Continued)*

```
# Polling period in seconds, is set in the sleep instruction.
# The script can be run from any account, but it is recommended
# to launch it as a startup script by including it in
# /etc/init.d/flam

LOCAL_NODE="`hostname"
LINKS_STATE="/var/tmp/flink_node_state.tmp"
CHKS0=0
while true
do
    date
    /usr/platform/sun4u/sbin/wrsmstat wrsm \
    | /bin/nawk -v NODE=$LOCAL_NODE '\
    /^Port ID/      {portid=$3} \
    /^Link [0-2]/   {link=$2}    \
    $2 ~ /State/  {linkstate=$3; \
    if (linkstate != "up") \
    {linkstate = "down"} \
    {printf ("Fire Link Status: %s WCI %s LINK %s : %s\n", \
    NODE, portid, link, linkstate)}}' \
    | tee $LINKS_STATE
    CHKS="`cat $LINKS_STATE | sum"
    echo "\n"
    if [ "$CHKS" != "$CHKS0" ]
    then
        mailx -s "Fire Link Alarm: $LOCAL_NODE  Links Status Change" \
        rudolph@east \
        < $LINKS_STATE
    fi
    CHKS0=$CHKS
    sleep 5
done
```

CODE EXAMPLE 10-14 Mail Generated by Auto-Notification Script

```
From: Super-User <root>
Message-Id: <200209190036.g8J0aEN15656@eis-sg24-1-a>
To: rudolph@east
Subject: Fire Link Alarm: eis-sg24-1-a  Links Status Change

Fire Link Status: eis-sg24-1-a WCI 29 LINK 0 : up
Fire Link Status: eis-sg24-1-a WCI 29 LINK 2 : down
Fire Link Status: eis-sg24-1-a WCI 31 LINK 0 : up
Fire Link Status: eis-sg24-1-a WCI 31 LINK 2 : up
From: Super-User <root>
```

```
Wed Sep 18 13:19:16 EDT 2002
Fire Link Status: eis-sg24-1-a WCI 29 LINK 0 : up
Fire Link Status: eis-sg24-1-a WCI 29 LINK 2 : down
Fire Link Status: eis-sg24-1-a WCI 31 LINK 0 : up
Fire Link Status: eis-sg24-1-a WCI 31 LINK 2 : up

Wed Sep 18 13:19:21 EDT 2002
Fire Link Status: eis-sg24-1-a WCI 29 LINK 0 : up
Fire Link Status: eis-sg24-1-a WCI 29 LINK 2 : down
Fire Link Status: eis-sg24-1-a WCI 31 LINK 0 : up
Fire Link Status: eis-sg24-1-a WCI 31 LINK 2 : up

Wed Sep 18 13:19:26 EDT 2002
Fire Link Status: eis-sg24-1-a WCI 29 LINK 0 : up
Fire Link Status: eis-sg24-1-a WCI 29 LINK 2 : down
Fire Link Status: eis-sg24-1-a WCI 31 LINK 0 : up
Fire Link Status: eis-sg24-1-a WCI 31 LINK 2 : up
```

Interpreting Link Status in POST Output

This section explains how to interpret optical link status that is reported by POST diagnostics.

Note – The Sun Fire Link portion of the POST diagnostics evaluate the integrity of the electrical circuits that directly support the link port. However, they do not turn on or otherwise perform functional tests of the optical links themselves.

Link Status in Sun Fire 6800 System POST Output

This section illustrates the portions of Sun Fire 6800 system POST output that relate directly to Sun Fire Link components. Use `setupdomain` to set the domain diag level to Default (at least). Each Sun Fire Link assembly undergoes two sets of POST tests, Board Interconnect tests and I/O Basic tests. Examples of each type of output are shown below.

CODE EXAMPLE 10-16 shows sample output from the POST Board Interconnect test. Any failure in these tests indicate that the WCI ASIC would fail in use.

CODE EXAMPLE 10-16 Sample Output From the POST Interconnection Tests

```
Powering boards on ...
{/N0/IB8/P1} Subtest: MemBistxt Passed
{/N0/IB8/P1} Subtest: MemBist Passed
{/N0/IB8/P1} Subtest: LogicBist Passed
{/N0/IB8/P1} Running Board Interconnect Test
{/N0/IB8/P1} Subtest: Wci1 to Link0 Test
{/N0/IB8/P1} Subtest: Wci1 to Link2 Test
{/N0/IB8/P1} Subtest: to Safari Test
{/N0/IB8/P1} Board Interconnect Test Passed
```

CODE EXAMPLE 10-17 shows sample output from the POST Basic I/O tests. The last two lines of the sample output represent tests of ports P0 and P1 on IB8. This P0 is the ASIC that controls the I/O interface cards contained in slots 0 and 3 of the Sun Fire Link assembly. In this example, P1 is the ASIC that controls the two optical links, 0 and 2. Since this sample POST output is for IB8, the WCI ID for this P1 is 29.

In this example, the last line shows that the ASIC passed the Basic I/O tests.

CODE EXAMPLE 10-17 Sample Output From the POST Basic I/O Tests

```
Copying IO prom to Cpu dram
..........{/N0/SB0/P0} Running PCI IO Controller Basic Tests ..............
{/N0/SB0/P0} Running Wci Basic Tests
{/N0/SB0/P0} Subtest: Wci Check Reset State for aid 0x1d
{/N0/SB0/P0} Subtest: Wci Register Initialization for aid 0x1d
{/N0/SB0/P0} Subtest: Wci Check SRAM Entries for aid 0x1d
{/N0/SB0/P0} Subtest: Wci Cluster Loopback Initialization for aid 0x1d
{/N0/SB0/P0} Subtest: Wci Cluster Start Performance Registers for aid 0x1d
{/N0/SB0/P0} Running Cluster Data Walk Tests
{/N0/SB0/P0} Subtest: Wci Cluster Data Walk Patterns for aid 0x1d
{/N0/SB0/P0} Subtest: Wci Cluster Data Half Patterns for aid 0x1d
{/N0/SB0/P0} Running Cluster Address Walk Tests
{/N0/SB0/P0} Subtest: Wci Cluster LoopBack Address Bits 12 to 6 for aid 0x1d
{/N0/SB0/P0} Subtest: Wci Cluster LoopBack Address Bits 21 to 13 for aid 0x1d
{/N0/SB0/P0} Subtest: Wci Cluster LoopBack Address Bits 33 to 22 for aid 0x1d
{/N0/SB0/P0} Subtest: Wci Cluster LoopBack Address Bits 41 to 34 for aid 0x1d
{/N0/SB0/P0} Running Wci Cluster Restore Test
{/N0/SB0/P0} Subtest: Wci  Restore Register State for aid 0x1d
{/N0/SB0/P0} Running Optical Link LoopBack Tests
{/N0/SB0/P0} Subtest: Wci Link LoopBack for aid 0x1d
{/N0/SB0/P0} After 5 Attempt(s), Node=0 Slot=8 Port=1 WCI=1 Link=0 link was
not in loopback
{/N0/SB0/P0} Node=0 Board=8 Port=1 WCI=1 Link=1 Link unpopulated
```

CODE EXAMPLE 10-17 Sample Output From the POST Basic I/O Tests *(Continued)*

```
{/N0/SB0/P0} After 5 Attempt(s), Node=0 Slot=8 Port=1 WCI=1 Link=2 link was
not in loopback
{/N0/SB0/P0} @(#) lpost        5.13.0  2002/05/28 20:31
{/N0/SB0/P0} Copyright 2001 Sun Microsystems, Inc. All rights reserved.
{/N0/IB8/P0} Passed
{/N0/IB8/P1} Passed
```

> **Note –** The path {N0/SB0/P0} points to the CPU. This simply means that the tests
> are being run from that processor.

To get a summary of the POST test results, enter the show-post-results
command at the OBP prompt. CODE EXAMPLE 10-18 contains an example of this. The
link-related POST output appears toward the end, in the Board 8 and Board 9
sections.

CODE EXAMPLE 10-18 Sample show-post-results Output

```
{2} ok
{2} ok show-post-results
Board 0, Type: CPU/Memory
  port  0 - Status = Pass, Type: CPU/Memory
           E-cache: 8 MB R-freq: 750 MHz A-freq: 750 MHz
  port  1 - Status = Pass, Type: CPU/Memory
           E-cache: 8 MB R-freq: 750 MHz A-freq: 750 MHz
  port  2 - Status = Pass, Type: CPU/Memory
           E-cache: 8 MB R-freq: 750 MHz A-freq: 750 MHz
  port  3 - Status = Pass, Type: CPU/Memory
           E-cache: 8 MB R-freq: 750 MHz A-freq: 750 MHz
Board 6, Type: PCI IO
  port 24 - Status = Pass, Type: PCI IO
  port 25 - Status = Pass, Type: PCI IO
Board 8, Type: CPCI WCI board with 2 pci slots & 2 PAROLI slots
  port 28 - Status = Pass, Type: CPCI WCI board with 2 pci slots & 2 PAROLI
  port 29 - Status = Pass, Type: CPCI WCI board with 2 pci slots & 2 PAROLI
Board 9, Type: CPCI WCI board with 2 pci slots & 2 PAROLI slots
  port 30 - Status = Pass, Type: CPCI WCI board with 2 pci slots & 2 PAROLI
  port 31 - Status = Pass, Type: CPCI WCI board with 2 pci slots & 2 PAROLI
{2} ok
```

The lines for port 29 and port 31 represent the status of the WCI ASICs in IB8 and
IB9, respectively. These ASICs are assigned WCI IDs 29 and 31. This output shows
that both ASICs passed the POST tests.

Link Status in Sun Fire 15K/12K System POST Output

On Sun F15K/12K systems, POST diagnostics run on the wpci board. The POST output includes a summary report that indicates the number of links available to the domain. If any links fail POST diagnostics, the summary identifies them as well.

In CODE EXAMPLE 10-19, the POST summary report indicates that one of four links in a Sun Fire 15K/12K domain is faulty.

CODE EXAMPLE 10-19 POST Summary With Link Failure Indication

```
CPU_Brds:   Proc  Mem P/B: 3/1 3/0  2/1 2/0  1/1 1/0  0/1 0/0
Slot Gen  3210       /L: 10  10   10  10   10  10   10  10      CDC
SB06:  P   PPPP          PP  PP   PP  PP   PP  PP   PP  PP        P

I/O_Brds:         IOC  P1/Bus/Adapt   IOC  P0/Bus/Adapt
Slot  Gen  Type   P1   B1/10 B0/10    P0   B1/eb10 B0/10   (e=ENet, b=BBC)
IO06:  P   wPCI                        P    p PP_p  p _m
IO07:  P   wPCI                        P    p fP_m  p _m

WCI_Brds:    WCI/Link    WCI/Link    WCI/Link
Slot  Gen    3  210     2  210      1   210
IO06:  P                            P   pmp
IO07:  P                            P   pmf

Configured in 333 with 4  procs, 8.000 GBytes, 1 IO adapter, 3  wlinks.
Interconnect frequency is 149.975 MHz, Measured.
Golden sram is on Slot IO6.
POST (level=16, verbose=40) execution time 4:52
```

The following explains how to interpret this report:

■ This domain was configured with two WPCI assemblies in I/O slots 6 and 7.

■ The first line in the text portion of the report:

```
Configured in 333 with 4 procs, 8.000 GBytes, 1 IO adapter, 3 wlinks.
```

shows that only three links are found to be available for use. There should be four links.

■ To determine which link is bad, look at the status code for each WPCI. This information is provided in the last column in the `WCI_Brds` section of the report.

CODE EXAMPLE 10-20

```
WCI_Brds    WCI/Link    WCI/Link    WCI/Link
Slot  Gen    3  210      2  210      1   210
IO06:  P                              P  pmp
IO07:  P                              P  pmf
```

■ For IO slot 6, links 0 and 2 both have a status of `p`, which means pass. Link 1 has a status of `m`, which means missing. Link 1 will always have a missing status because there are only two links per slot, which are assigned 0 and 2.

■ For IO slot 7, link 2 has a status of `p`, but the status of link 0 is `f`, meaning it has failed a POST test.

Link Status in Sun Fire Link Switch POST Output

POST diagnostics for Sun Fire Link switches include tests that evaluate each of the switch's links. CODE EXAMPLE 10-21 shows the results from tests that were run on link 1 in the switch `wcsw1`. As with other POST output, any test failure is indicated by the keyword `FAILED`.

CODE EXAMPLE 10-21 Portion of Link Test Output From Switch POST Session

```
Sep 19 17:58:29 wcsw1 Switch.POST: {/LINK1/}  Testing Link 1...
Sep 19 17:58:29 wcsw1 Switch.POST: {/LINK1/}  Running test "i2cprobe"
Sep 19 17:58:29 wcsw1 Switch.POST: {/LINK1/}   Test "i2cprobe" PASSED.
Sep 19 17:58:29 wcsw1 Switch.POST: {/LINK1/}  Running test "volt"
Sep 19 17:58:30 wcsw1 Switch.POST: {/LINK1/}   Test "volt" PASSED.
Sep 19 17:58:30 wcsw1 Switch.POST: {/LINK1/}  Running test "temp"
Sep 19 17:58:30 wcsw1 Switch.POST: {/LINK1/}   Test "temp" PASSED.
Sep 19 17:58:30 wcsw1 Switch.POST: {/LINK1/}  Running test "frubasic"
Sep 19 17:58:32 wcsw1 Switch.POST: {/LINK1/}   Test "frubasic" PASSED.
Sep 19 17:58:32 wcsw1 Switch.POST: {/LINK1/}  Running test "jtagid"
Sep 19 17:58:32 wcsw1 Switch.POST: {/LINK1/}   Test "jtagid" PASSED.
Sep 19 17:58:34 wcsw1 Switch.POST: {/LINK1/}  Running test "interconnect"
```

```
Sep 19 17:58:35 wcsw1 Switch.POST: {/LINK1/}   WCIX is in use.  Running
hot-plug interconnect test.
Sep 19 17:58:44 wcsw1 Switch.POST: {/LINK1/}  Test "interconnect" PASSED.
Sep 19 17:58:44 wcsw1 Switch.POST: {/LINK1/}  JPOST version 1.12.28 PASSED
on Link 1.
```

Collecting Troubleshooting Information for Sun Services

This section explains how to collect diagnostic information that can be helpful in troubleshooting a Sun Fire Link cluster. If you need to request technical support from Sun Service for help with your Sun Fire Link cluster, you should have this information available at the time you make the request.

The most efficient method for collecting this information is to run the Sun Explorer data collector tool. If you do not already have this tool installed, you can download a copy from the Explorer page on the GeSE Web site. At the time of publication, the URL for the GeSE Web site is:

`http://ginko.central/kds/explorer/index.html`

This site includes instructions for downloading and installing the SUNWexplo package that contains the Explorer data collection scripts.

Run Sun Explorer on your cluster's FM Management Station and on each cluster node that you suspect has a link failure. To do this, log onto the FM Management Station and cluster nodes as root and enter:

```
# /opt/SUNWexplo/bin/explorer
```

Sun Explorer gathers system information, bundles it into a compressed tar file, which can be sent to Sun Services. On the FM Management Station, the Sun Explorer scripts collect detailed fabric information, including topology and link states for the partitions in each fabric. On each cluster node, it collects:

■ WCI interface statistics
■ Routes information
■ RSM controller statistics
■ RSM controller configuration data

If Sun Explorer is not available, you can collect sufficient information using various Solaris and FM administration commands.

On the FM Management Station, run the `wcfmstat` command, as described in "Using the FM CLI to Monitor Link Status" on page 140.

On each cluster node, run the Solaris commands `wrsmsat` as described in "Using the WRSM CLI to Monitor Link Status at the Node Level" on page 132.

In addition, copy the relevant section of `/var/adm/messages`.

Command Line Interface

This appendix explains how to use the Sun Fire Link FM command-line interface (CLI) to configure and manage Sun Fire Link clusters. The tasks described here include:

- Creating a fabric
- Starting a fabric
- Configuring a fabric
- Monitoring configuration status
- Discovering links
- Removing a fabric

These tasks make use of the following CLI commands:

- `listfabrics`(1m)
- `createfabric`(1m)
- `startfabric`(1m)
- `stopfabric`(1m)
- `killfabrics`(1m)
- `deletefabric`(1m)
- `wcfmconf`(1m)
- `wcfmstat`(1m)
- `wcfmver`(1m)

The six *fabric* commands are all described in a single man page, `fabric`. Individual man pages are provided for `wcfmconf`, `wcfmstat`, and `wcfmver`.

Configuring Sun Fire Link clusters using `wcfmconf` requires the creation of an XML-based configuration file for each fabric. See Appendix B for instructions.

Note – *All* text entries, including but not limited to fabric, domain, and partition names, are restricted to ASCII characters. This applies to any text you supply to the Sun Management Center GUI as well as text you enter when editing an XML-based configuration file.

These commands described in this appendix all include a –h command-line option, which displays usage information for the command. They also have options that were developed for internal use. These options are not documented here because they are of no benefit in the administration of Sun Fire Link clusters.

Creating a New Fabric

Use the createfabric command to initiate a new fabric. This establishes a name for the fabric and creates directories to hold the fabric's configuration files and message logs.

When a fabric is created, it is initially in an unconfigured state. That is, it is not associated with a configuration file that identifies a set of nodes and, optionally, switches as members of that fabric. See Appendix B for a description of fabric configuration files and instructions on how to create and edit them.

The syntax for this command is:

```
createfabric [ –h ] fabric_name
```

- –h – Displays the usage for createfabric (optional)
- *fabric_name* – Specify the name of the fabric (required)

▼ To Create a New Fabric

● **Type:**

```
# cd /opt/SUNWwcfm/bin
# ./createfabric fabric_name
```

Starting, Initializing, Resetting, and Verifying a Fabric

Use the `startfabric` command to invoke a fabric manager for the specified fabric. The fabric must already exist. The syntax for this command is:

```
startfabric [ -h ][ -p rmi_port ] [ [ -i [ config_file ] ] | [ -r
[ [ config_file ] | [ scname:domain , switch_name, scname:domain ... ]
] ] | [ -v ] ]   fabric_name
```

- `-h` – Displays the usage for `startfabric` (optional)
- `-p` – Specify the RMI port that the fabric will use. The default is 1099 (optional)
- `-i` – Initialize a fabric's configuration data on the FM server and start the fabric's instance daemon. (optional)
- `-r` – Reset a fabric's configuration data to a prescribed state on all nodes and switches in a cluster or on particular nodes and/or switches and start the fabric's instance daemon. (optional)
- `-v` – Verify the state of the fabric on all nodes and switches. (optional)
- *fabric_name* – Specify the name of the fabric (required)

These options allow you to use the `startfabric` command to initialize, reset, and verify a fabric.

The initialization option allows you to remove all data previously associated with the named fabric and to start the fabric in either of the following states:

- Initialize to an unconfigured state – If no configuration file is specified, the fabric will be unaffiliated with any nodes or switches.
- Initialize to a new configured state – If a configuration file is specified, the fabric's configuration will be defined by that file.

The reset option allows you to reset a fabric's configuration data on cluster nodes and switches, with the following choices:

- Reset all nodes and switches to current configuration – If no nodes or switches are specified and no configuration file is specified, all the nodes and switches will have their local configuration data updated to match the fabric's current configuration file.
- Reset specific nodes and/or switches to current configuration – If one or more nodes and/or switches are specified, the specified nodes and/or switches will have their local configuration data updated to match the fabric's current configuration file.

- Reset to a new configuration – If a configuration file is specified, all the nodes and/or switches will have their local configuration data updated to match the specified configuration file.

The primary purpose of the −r option is to provide a way to restore configuration data directly on nodes and switches when something happens to cause their local configuration state to become inconsistent with the established state. In such cases, the −r option would be used without the *config_file* argument.

Note – Using −r *config_file* has the same effect as −i *config_file*.

With the verify option, you can verify the state of an existing fabric on all fabric member node(s) and switch(es). The information provided by the verify option allows you to identify specific fabric members that have a configuration state that is inconsistent with the current fabric state. Based on this information, you can selectively reset those nodes and/or switches rather than resetting all the members of the fabric.

▼ To Start a Fabric

- **Type:**

```
# cd /opt/SUNWwcfm/bin
# ./startfabric fabric_name
```

Verifying and Resetting a Fabric

CODE EXAMPLE A-1 shows the verify option being used, with its output indicating that the switch minime and the compute node hamlin:A both have invalid FM node IDs.

CODE EXAMPLE A-1

```
node1# ./startfabric -v testfab
Verifying Fabric...
FM Node ID invalid for Fabric Member minime.        *** invalid switch
FM Node ID verified for Fabric Member vanessa.
FM Node ID invalid for Fabric Member hamlin:A.      *** invalid node
FM Node ID verified for Fabric Member hamlin:B.
```

CODE EXAMPLE A-1

```
Validating partition part1:
    Verified node: hamlin-b
    Invalid FM Node ID on node: hamlin-a
    Invalid FM Node ID on switch: minime
    Verified switch: vanessa
Concluded partition verification.
Fabric Verification Completed.
```

CODE EXAMPLE A-2 shows the reset option being used to reset the switch `minime` and the node `hamlin:A`. This step resets all configuration information on `minime` and `hamlin:A`, leaving the other nodes and switches untouched. Since no configuration file is specified, the current configuration is used.

CODE EXAMPLE A-2

```
node1# ./startfabric -r minime hamlin:A testfab
Reset Nodes set to: minime hamlin:A
Stopping Fabric testfab for Reset/Verify
Found FM at [rmi://localhost:1099/testfab]
Fabric "testfab" stopped, the FM process may not exit for several seconds
Resetting Fabric
testfab Bound To Registry
```

Configuring the Fabric

Use the `wcfmconf` command to configure a fabric and create one or more partitions. You need a completed XML configuration file for this process. See Appendix B for an explanation of this process.

Note – When an XML configuration file is invoked by the `wcfmconf` command, the fabric name specified in that file must match exactly the name of a fabric that has already been created and started.

The syntax of this command is

```
wcfmconf [ –h ] [ –p rmi_port ] configuration_file_path
```

- `–h` – Displays the usage for `wcfmconf` (optional)
- `–p` – Specify the RMI port that the fabric will use. The default is 1099 (optional)

- *configuration_file_path* – The absolute or relative path to the XML configuration file

---

Note – Before starting the configuration process, verify that the configuration file satisfies your site-specific configuration requirements.

Set the configuration file's read/write/execute permissions to 600.This is not a requirement, but the contents of the file should be protected from accidental modification.

▼ To Configure a Sun Fire Link Fabric

● **Type:**

```
# cd /opt/SUNWwcfm/bin
# ./wcfmconf configuration_file_path
```

If the configuration process is successful, a message will be displayed indicating that the Sun Fire Link Manager (FM) was found. CODE EXAMPLE A-3 illustrates this is with sample output for a fabric named `fmdc`. The output concludes with a message indicating that the FM was found at `localhost:1099/fmdc`. The output does not include any error messages.

CODE EXAMPLE A-3   Output From a Successful `wcfmconf` Run

```
# ./wcfmconf fmdc.xml
Config file set to fmdc.xml
Found FM at [//localhost:1099/fmdc]
Configuration file processed successfully.
```

CODE EXAMPLE A-4 shows output from an unsuccessful execution of `wcfmconf`. This output includes a message indicating that an error occurred when attempting to read the configuration file.

CODE EXAMPLE A-4   Output Example From an Unsuccessful `wcfmconf` Run

```
# ./wcfmconf fmdc.xml
Config file set to fmdc.xml
Found FM at [//localhost:1099/fmdc.xml]
Configuration failed
Missing Links are:
moab::0=fmdc:a:29
```

Monitoring Configuration Status

Use the `wcfmstat` command on the FM host to gather member and status information about the fabric. The `wcfmstat` command also provides link and route information for a specified partition or node.

The syntax of this command is

```
wcfmstat [ –h ] [ –r rmi_port ] [ –p partition_name |
–n node_name ] fabric_name
```

- `–h` – Displays the usage for `wcfmstat`.
- `–r` – Specify the RMI port that the fabric will use. The default is 1099 (optional).
- `–p` – Specify the name of an existing partition (optional).
- `–n` – Specify the System Controller name, followed by the domain ID, in the format, *sc_name:domain* (optional).
- *fabric_name* – Specify the name of the fabric (required).

▼ To Show Fabric Information at the Highest Level

- **Type:**

```
# cd /opt/SUNWwcfm/bin
# ./wcfmstat fabric_name
```

CODE EXAMPLE A-5 shows `wcfmstat` output when the fabric name is the argument. In this example, there are no unassigned nodes or switches.

CODE EXAMPLE A-5 Output From `wcfmstat` at the Fabric Level

```
# ./wcfmstat fmdc
print_fabric_status
Found FM at [rmi://localhost:1099/fmdc]
Fabric name is "fmdc"
Partition                                        Stripe Level
Name             Type     Topology               WCI      Link
fmdc             RSM      WCIX_SWITCH            2        2
All members of this fabric are assigned to partitions.
```

▼ To Show Fabric Members, Routes, and Links for a Specific Partition

- **Type:**

```
# cd /opt/SUNWwcfm/bin
# ./wcfmstat -p partition_name  fabric_name
```

CODE EXAMPLE A-6 shows `wcfmstat` output when the partition option is specified as well as the fabric name.

CODE EXAMPLE A-6   Output From `wcfmstat` Showing Fabric Members, Routes, and Links for a Specific Partition

```
# ./wcfmstat -p part1 fmdc
print_partition_status
Found FM at [rmi://localhost:1099/fmdc]
Partition Name: part1
Members:
  hamlin-a
  hamlin-b
  minime-null
  vanessa-null
-----------------------------------------
Routes:
  hamlin:a=hamlin:b
    hamlin:a:8:1:0=minime::::0=0
    hamlin:a:8:1:2=vanessa::::0=0
    hamlin:a:9:1:0=vanessa::::2=0
    hamlin:a:9:1:2=minime::::2=0
    hamlin:b:8:1:0=minime::::1=0
    hamlin:b:8:1:2=vanessa::::1=0
    hamlin:b:9:1:0=minime::::3=0
    hamlin:b:9:1:2=vanessa::::3=0
-----------------------------------------
Links:
  hamlin:a:8:1:0=minime::::0=0
  hamlin:a:8:1:2=minime::::0=0
  hamlin:a:9:1:0=minime::::2=0
  hamlin:a:9:1:2=minime::::2=0
  hamlin:b:8:1:0=minime::::1=0
  hamlin:b:8:1:2=minime::::1=0
  hamlin:b:9:1:0=minime::::3=0
  hamlin:b:9:1:2=minime::::3=0
```

The Routes section of the output lists every route and, under each route description, the individual links that make up that route. A route description is simply the two domains that are connected by the route, joined by an equal sign. The single route listed in CODE EXAMPLE A-6, connects `hamlin:a` and `hamlin:b`.

Each link description consists of two endpoint descriptions joined by an equal sign. Each endpoint description consists of five fields, which are described below:

■ SC or SSC name – The name of the system (node or switch).

■ Domain ID – For Sun Fire 6800 systems, this will be a letter in the range `a` – `d`. For Sun Fire 15K/12K systems, this will be a letter in the range `a` – `r`. For switches, this field will be empty.

■ Chassis I/O slot – For Sun Fire 6800 systems, this will be either `8` or `9`. For Sun Fire 15K/12K systems, the I/O slot number will be in the range `0` – `17`. For switches, this field will be empty.

■ WCI ID – For Sun Fire 6800 systems, the WCI ID value is always `1`. For Sun Fire 15K/12K systems, the WCI ID value equals (*slot number* x 32) + 29. See FIGURE 10-3 for a complete list of the WCI ID values used in Sun Fire 15K/12K systems. For switches, this field will be empty.

■ Link (port) ID – For both Sun Fire 6800 and Sun Fire 15K/12K systems, this will be either `0` or `2`. For switches, the link number will be in the range `0` – `7`.

The link description is completed with a status value, which can be one of the following:

■ `0` – Link is up.
■ `–1` – Link is down.

The Links section of the output lists all links found in the partition without associating them with particular routes.

Verifying Fabric Connectivity

Use the `wcfmver` command to verify that the connections specified in an XML configuration file are matched by actual links in the hardware. You can supply `wcfmver` with data in two ways.

■ In the first method you supply a fabric configuration file using the `–c` option. The `wcfmver` program reports only links specified in the file that have not been revealed by the discovery process and any links discovered that are not listed in the configuration file.

■ In the second method, you supply a list of fabric nodes using the `–n` option. The `wcfmver` program then reports all discovered link information for each of the specified nodes. All specified nodes must exist within the fabric.

> **Note** – The −n option is particularly helpful when used in conjunction with a chart that describes the physical connections between nodes or between nodes and switches.

The syntax of this command is

```
wcfmver [ −h ] [ −p rmi_port ] {−c config_file_path | −n SC_name:Domain
switch_name ... } fabric_name
```

- −h – Displays usage for `wcfmver`
- −p – Specify the RMI port that the fabric will use. The default is 1099 (optional)
- −c – Specify the file containing all the links in the fabric (either this option or −n is required). Note that this refers to the file created by the FM, not the XML configuration file described in Appendix B.
- −n – List of nodes that link verification should be performed on (either this option or −c is required)
- *fabric_name* – The name of the fabric (required)

▼ To Verify Fabric Connectivity for Specified Nodes

- **Type:**

  ```
  # cd /opt/SUNWwcfm/bin
  # ./wcfmver −n SC_name:Domain, switch_name fabric_name
  ```

  CODE EXAMPLE A-7 shows `wcfmver` output when two nodes are specified.

  > **Note** – See "To Show Fabric Members, Routes, and Links for a Specific Partition" on page 166 for an explanation of link descriptions.

CODE EXAMPLE A-7   Output From `wcfmver` Showing the Link Connections Found for the Nodes `hamlin-a` and `hamlin-b`

```
# ./wcfmver −n hamlin-sc0:A, hamlin-sc0:b fmdc
Found FM at [//localhost:1099/testfab]

Starting discovery.
Estimated wait time = 180 sec.
...............
```

CODE EXAMPLE A-7 Output From `wcfmver` Showing the Link Connections Found for the Nodes `hamlin-a` and `hamlin-b` *(Continued)*

```
The following links are in the configuration but were not discovered...


The following links were discovered but are not in the configuration...

hamlin-sc0:A:6:1:2=unibrow::::2
hamlin-sc0:A:6:1:0=minime::::0
hamlin-sc0:A:8:1:2=drevil::::4
hamlin-sc0:A:8:1:0=vanessa::::0
hamlin-sc0:B:7:1:2=minime::::1
hamlin-sc0:B:7:1:0=unibrow::::0
hamlin-sc0:B:9:1:2=vanessa::::1
hamlin-sc0:B:9:1:0=drevil::::6
```

Listing Fabrics

Use the `listfabrics` command to display a list of all fabrics currently running. It lists all fabrics bound to the specified RMI registry.

The syntax of this command is

```
listfabrics [ –h ] [ –p rmi_port ]
```

- ■ –h – Displays the usage for `listfabrics` (optional)
- ■ –p – Specify the RMI port that the fabric will use. The default is 1099 (optional)

▼ To List Fabrics

● **Type:**

```
# cd /opt/SUNWwcfm/bin
# ./listfabrics
```

Stopping a Fabric

Use the `stopfabric` command to stop the named fabric. This unbinds the FM associated with the fabric from the RMI registry.

The syntax of this command is

```
stopfabric [ –h ] [ –p rmi_port ] [ –f ] fabric_name
```

- ■ `–h` – Displays the usage for `listfabrics` (optional)
- ■ `–p` – Specify the RMI port that the fabric will use. The default is 1099 (optional)
- ■ `–f` – Force the specified fabric to be stopped regardless of its state.
- ■ *fabric_name* – The name of the fabric (required)

▼ To Stop a Fabric

- ● **Type:**
  ```
  # cd /opt/SUNWwcfm/bin
  # ./stopfabric fabric_name
  ```

Stopping All Fabrics

Use the `killfabrics` command to stop all fabrics from running. It also removes the RMI registries to which the associated FM daemons were bound.

The syntax of this command is

```
killfabrics [ –h ]
```

where `–h` displays the usage for `killfabrics`.

▼ To Stop All Fabrics

- **Type:**

  # **cd /opt/SUNWwcfm/bin**
  # **./killfabrics**

Deleting Fabrics

The deletefabric command removes all fabric information associated with a named fabric.

The syntax of this command is

```
deletefabric [–h] fabric_name
```

- –h – Displays the usage for listfabrics (optional)
- *fabric_name* – The name of the fabric (required)

▼ To Delete a Fabric

- **Type:**

  # **cd /opt/SUNWwcfm/bin**
  # **./deletefabric** *fabric_name*

Sun Fire Link XML Configuration Files

Sun Fire Link clusters are defined in XML-formatted configuration files, with a separate configuration file used for each Sun Fire Link fabric. When you configure a cluster using the Sun Management Center/FM interface, the Sun Fire Link Manager software creates the configuration file automatically. When you use the FM command-line interface (CLI), you need to supply the configuration file.

This appendix explains how to create an XML-based Sun Fire Link configuration file by hand. It provides a set of configuration templates to simplify the creation process.

These templates contain the essential features needed to implement a representative variety of basic configurations. If you plan to configure a cluster that has a different number of nodes and/or switches, you can simply modify the template that most closely resembles the configuration you want to achieve.

This appendix also illustrates the Data Type Definition (DTD) on which the configuration files are based.

General Guidelines

The following list presents a set of recommendations that can make the creation of your Sun Fire Link configuration files go smoothly.

- All text in the XML configuration file must consist of ASCII characters exclusively.
- If available, use an existing configuration file as a starting point to minimize the amount of new text you enter.
- If you expect to create many Sun Fire Link configuration files by hand, use an XML editing tool. This will help you avoid syntactical errors, which can be hard to debug without the aid of such a tool.

- Except for server passwords, use *only lower-case characters* in the XML file.

- Be certain that all SC names, domain names, hostnames, and passwords are spelled correctly in every instance where they appear.

- Be consistent with SC names and domain names. For example, if an SC name or domain name is listed in the fabric members, the same name must be listed as a partition member.

- Avoid specifying the following optional elements. The FM will generate suitable values automatically.
  - `config_file`
  - `properties`
  - `last_date_time`
  - `fm_nodeid`
  - `partition_id`

- The following fields must have unique values:
  - `fname` – Fabric name.
  - `fm_nodeid` – FM node ID.
  - `pname` – Partition name.
  - `partition_id` – Partition ID.

  If you do not specify the `fm_nodeid` and `partition_id`, the FM (as the previous bullet recommends), the FM will ensure that these values are unique.

- The `chassis_type` field must have one of the following values:
  - `S24` – Use this value for Sun Fire 6800 systems.
  - `S72` – Use this value for Sun Fire 15K/12K systems.
  - `wcix_switch` – Use this value for Sun Fire Link switches.

  If you specify an invalid `chassis_type`, an invalid configuration will be created, which the RSM driver will reject.

- Specify both `wci_striping_level` and `link_striping_level` explicitly instead of just a single `stripe_level` value. This will ensure that the stripe configuration will be as you prefer. Otherwise, the FM will make the choice for you.

- If you specify reserve wcis or reserve links that do not exist as hardware, a missing hardware error will be generated. This error will also be generated if you specify too few links or wcis to support the striping level you specify.

- If you specify a partition that already exists, the new configuration will be treated as an upgrade operation.

Creating an XML Configuration File

This section describes two procedures for editing an XML template to configure a Sun Fire Link cluster. One procedure is for configuring *single-controller* clusters; the other is for configuring *dual-controlle*r clusters. The key features of these two configuration types are explained below.

Single-Controller Configuration Summary

Single-controller clusters are used for Sun HPC ClusterTools applications—that is, clusters that have Sun HPC ClusterTools software running on the nodes.

In this type of configuration, the RSM software in each domain maintains a single logical controller to manage both Sun Fire Link ASICs used by that domain. This means each domain has all four optical ports available for sending and receiving messages across the Sun Fire Link network.

See "Configuring a Single-Controller Cluster" on page 176 for a instructions on creating this type of configuration.

Dual-Controller Configuration Summary

Dual-controller clusters are used for Sun Cluster applications in which each domain must have Sun Fire Link ASIC failover support.

To achieve this, each domain's two Sun Fire Link ASICs are mapped into separate partitions, and separate RSM controllers are allocated to manage the two partitions. One ASIC and the two optical ports it controls function as the primary network interface. The other ASIC is brought into use if the primary network interface should fail.

See "Configuring a Dual-Controller Cluster" on page 184 for an example of a dual-controller configuration.

Note – The Sun Fire Link ASIC is the circuit chip that is at the heart of the Sun Fire Link assembly. The term *WCI*, which you will see in some system messages and some Sun Management Center/FM console icon labels, refers to the Sun Fire Link ASIC.

Summary of Striping Options

The striping levels you are able to choose for partitions are affected by the following configuration variables:

- Single controller versus dual controller
- Three-node, direct connect

TABLE B-1 provides a summary of how these variables limit the wci and link striping levels that are available to partitions.

TABLE B-1   WCI and Link Striping Levels for Different Configuration

	WCI Striping per Partition	Link Striping per Partition
Single controller		
three-node direct connnect	can be 1 or 2	must be 1
all other cluster configurations	can be 1 or 2	can be 1 or 2
Dual controller		
three-node direct connnect	must be 1	must be 1
all other cluster configurations	must be 1	can be 1 or 2

Configuring a Single-Controller Cluster

The template shown in CODE EXAMPLE B-1 represents a Sun Fire Link fabric that contains four Sun Fire 6800 chassis (one domain per chassis) and two Sun Fire Link switches. These are configured into a single partition.

Note – Because both Sun Fire Link ASICs on a domain are configured into the same partition, that domain has a single-controller configuration.

Instructions for creating a Sun Fire Link cluster that matches this template are provided after the example. The instructions also include guidelines for creating a single-controller configuration with different characteristics.

CODE EXAMPLE B-1   Four-Node, Two-Switch, Single-Controller Configuration

```
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE fabric SYSTEM "fabric.dtd">
<fabric>
      <fname>name _of_fabric</fname>
      <config_file>XML_config_filename.xml</config_file>
      <members>
             <switch_node>
              <node>
                     <sc_name>switch1_sc_hostname</sc_name>
                     <sc_user_name>switch1_sc_username</sc_user_name>
                     <sc_password>switch1_sc_password</sc_password>
                     <chassis_type>wcix_switch</chassis_type>
                     <info>description of switch1 location</info>
              </node>
              </switch_node>

              <switch_node>
              <node>
                     <sc_name>switch2_sc_hostname</sc_name>
                     <sc_user_name>switch2_sc_username</sc_user_name>
                     <sc_password>switch2_sc_password</sc_password>
                     <chassis_type>wcix_switch</chassis_type>
                     <info>description of switch2 location</info>
              </node>
              </switch_node>
              <rsm_node>
              <node>
                     <sc_name>node1_sc_hostname</sc_name>
                     <sc_user_name>node1_sc_username</sc_user_name>
                     <sc_password>node1_sc_password</sc_password>
                     <chassis_type>S24</chassis_type>
                     <info>description of node1 location</info>
              </node>
                    <domain_name>node1_domain_name</domain_name>
                    <hostname>node1_hostname–domain</hostname>
                    <host_user>node1_username</host_user>
                    <host_password>node1_password</host_password>
              </rsm_node>
              <rsm_node>
              <node>
                     <sc_name>node2_sc_hostname</sc_name>
                     <sc_user_name>node2_sc_username</sc_user_name>
                     <sc_password>node2_sc_password</sc_password>
                     <chassis_type>S24</chassis_type>
```

```
                    <info>description of node2 location</info>
            </node>
                    <domain_name>node2_domain_name</domain_name>
                    <hostname>node2_hostname-domain</hostname>
                    <host_user>node2_username</host_user>
                    <host_password>node2_password</host_password>
            </rsm_node>

            <rsm_node>
            <node>
                    <sc_name>node3_sc_hostname</sc_name>
                    <sc_user_name>node3_sc_username</sc_user_name>
                    <sc_password>node3_sc_password</sc_password>
                    <chassis_type>S24</chassis_type>
                    <info>description of node3 location</info>
            </node>
                    <domain_name>node3_domain_name</domain_name>
                    <hostname>node3_hostname-domain</hostname>
                    <host_user>node3_username</host_user>
                    <host_password>node3_password</host_password>
            </rsm_node>

            <rsm_node>
            <node>
                    <sc_name>node4_sc_hostname</sc_name>
                    <sc_user_name>node4_sc_username</sc_user_name>
                    <sc_password>node4_sc_password</sc_password>
                    <chassis_type>S24</chassis_type>
                    <info>description of node4 location</info>
            </node>
                    <domain_name>node4_domain_name</domain_name>
                    <hostname>node4_hostname-domain</hostname>
                    <host_user>node4_username</host_user>
                    <host_password>node4_password</host_password>
            </rsm_node>
    </members>
    <partitions>
            <partition type="RSM" topology="WcixSwitch">
                    <pname>partition1_name</pname>
                    <wci_striping_level>wci_stripe_level</wci_striping_level>
                    <link_striping_level>link_stripe_level</link_striping_level>
                    <partition_members>
                        <switch_partition_member>
                                <sc_name>switch1_sc_hostname</sc_name>
                        </switch_partition_member>
                        <switch_partition_member>
                                <sc_name>switch2_sc_hostname</sc_name>
```

```
                      </switch_partition_member>
                      <node_partition_member>
                           <sc_name>node1_sc_hostname</sc_name>
                           <domain_name>node1_domain</domain_name>
                      </node_partition_member>
                      <node_partition_member>
                           <sc_name>node2_sc_hostname</sc_name>
                           <domain_name>node2_domain</domain_name>
                      </node_partition_member>
                      <node_partition_member>
                           <sc_name>node3_sc_hostname</sc_name>
                           <domain_name>node3_domain</domain_name>
                      </node_partition_member>
                      <node_partition_member>
                           <sc_name>node4_sc_hostname</sc_name>
                           <domain_name>node4_domain</domain_name>
                      </node_partition_member>
                 </partition_members>

             </partition>
      </partitions>
</fabric>
```

Editing the Single-Controller Template

The following procedure is based on the assumption that you have a copy of the template shown in CODE EXAMPLE B-1 and have use of an editor—an XML editor, if possible.

To make the steps easier to follow, the procedure is presented in four stages:

- Entering the file header information.
- Naming the fabric.
- Adding switch members to the fabric.
- Adding node members to the fabric.
- Defining a partition.

▼ To Create the File Header

A Sun Fire Link XML configuration file begins with the following lines:

- `<?xml version="1.0" standalone="no" ?>`

- `<!DOCTYPE fabric SYSTEM "fabric.dtd">`
- `<fabric>`

If you are using a template, do not change the contents of these lines.

▼ To Name a Fabric

1. Enter the name of the fabric between `<fname>` **and** `</fname>`**.**

This name match exactly the spelling of the name you give to the fabric when you create it with `createfabric`, as described in Appendix A. This name must also be unique.

Note – You can create the XML configuration file for a fabric before you create or start the fabric, but the fabric must be created before the configuration file is used by `wcfmconf`.

---

2. Enter the file name that this file will be known by between `<config_file>` **and** `</config_file>`**.**

This entry is optional and can be left blank, but the bracketing elements `<config_file></config_file>` must be present.

3. Enter `<members>` **to begin the section of the file that will specify the fabric's switch (if any) and node membership.**

▼ To Add Switches to the Fabric

1. If you want to include switches in this fabric, describe one of the switches in a section of the file that is bracketed by `<switch_node>` **and** `</switch_node>`**. If you don't want to include an switches, omit this section altogether.**

The information you need to supply for switches is described below:

a. Enter the host name of the Switch System Controller (SSC) between `<sc_name>` **and** `</sc_name>`**.**

b. Enter a user name between `<sc_user_name>` **and** `</sc_user_name>`**.**

c. Enter the SSC's community password between `<sc_password>` **and** `</sc_password>`**.**

This is the RMI password that provides security to communication between the FM and switches. See "To Provide an RMI Password for the FM/SSC Interface" on page 14 for more information.

d. **Enter** `wcix_switch` **between** `<chassis_type>` **and** `</chassis_type>`.

e. **The** `<info>` **field is optional. If you want to describe the physical location of this switch, enter a brief text description between** `<info>` **and** `</info>`.

2. **Repeat Step 1 for each switch that you want to include in this fabric.**

   Each individual switch description must be bracketed as follows:

   - `<switch_node>`
   - `<node>`
   - Switch description
   - `</node>`
   - `</switch_node>`

▼ To Add Nodes to the Fabric

1. **Describe one of the compute nodes in a section of the file that is bracketed by** `<rsm_node>` **and** `</rsm_node>`.

   The information you need to supply is described below:

   a. **Enter the SC host name of the domain console between** `<sc_name>` **and** `</sc_name>`.

   b. **Enter a user name between** `<sc_user_name>` **and** `</sc_user_name>`. **This can be any name.**

   c. **Enter the password for this domain console between** `<sc_password>` **and** `</sc_password>`.

   Refer to "To Create a Password for Domain Consoles" on page 13 for more information.

   d. **For Sun Fire 6800 systems, enter** `S24` **between** `<chassis_type>` **and** `</chassis_type>`.

   For Sun Fire 15K/12K systems, enter `S72` between `<chassis_type>` `</chassis_type>`.

   e. **The** `<info>` **field is optional. If you want to describe the physical location of this node, enter a brief text description between** `<info>` **and** `</info>`.

2. **In the section following the** `</node>` **delimiter and before the** `</rsm_node>` **delimiter, enter domain information associated with this host.**

   a. **Enter the domain name between** `<domain_name>` **and** `</domain_name>`.

   For Sun Fire 6800 systems, this value will be in the range `a` through `d`. For Sun Fire 15K/12K systems, this value will be in the range `a` through `r`.

b. Enter the host name for the domain between `<hostname>` **and** `</hostname>`**.**

For example, if the host name for domain a is `node1-a`, enter `node1-a`.

c. Enter a user name between `<host_user>` **and** `</host_user>`**.**

The user must already have been created and be a member of the group `sfladmin`. See "To Set Up WRSM Proxy Security on Cluster Nodes" on page 12 for details.

d. Enter the password for the user specified in Step c between `<node_password>` **and** `</node_password>`**.**

Refer to "To Set Up WRSM Proxy Security on Cluster Nodes" on page 12 for more information.

3. Repeat Step 1 and Step 2 for each node that you want to include in this fabric.

Each individual switch description must be bracketed as follows:

- `<rsm_node>`
- `<node>`
- SC description
- `</node>`
- Domain description
- `</rsm_node>`

4. When you have described all the switches and nodes that you want to include in this fabric, terminate the section with `</members>`**.**

▼ To Define a Partition in a Fabric

1. Begin the partitions description section with `<partitions>`**.**

2. Enter the type and topology of the partition.

Specify the type and topology attributes of the partition as follows:

- `<type>` – Always enter `"RSM"`. Include the quote marks.
- `<topology>` – Enter `"DirectConnect"` for a partition with *no* switches. Specify `"WciSwitch"` for a partition that includes switches.

3. Enter a name for the partition between `<pname>` **and** `</pname>`**.**

This name must be unique within the fabric.

4. Enter a wci striping level of `1` **or** `2` **between** `<wci_striping_level>` **and** `</wci_striping_level>`**.**

Since this example is for a single-controller configuration, you should specify 2 so that maximum bandwidth is enabled.

5. **Enter a link striping level of** `1` **or** `2` **between** `<link_striping_level>` **and** `</link_striping_level>`**.**

   Ordinarily, you should specify `2` so that the partition has the benefit of maximum bandwidth.

6. **Enter** `<partition_members>` **to begin the partition membership section.**

7. **If the partition will contain switches, enter the SSC host name of one of the switches between** `<sc_name>` **and** `</sc_name>`**.**

   This name must match exactly one of the SSC host names in the fabric member section. This entry is bracketed in the following manner:

   - `<switch_partition_member>`
   - `<sc_name>switch_sc_hostname</sc_name>`
   - `</switch_partition_member>`

8. **Repeat Step 7 for each switch to be included in the partition.**

9. **Enter the following information about one of the nodes in a section of the file that is bracketed by** `<node_partition_member>` **and** `</node_partition_member>`**.**

   a. **Enter the host name of the node's SC between** `<sc_name>` **and** `</sc_name>`**.**

   b. **Enter the domain name between** `<domain_name>` **and** `</domain_name>`**.**

      For Sun Fire 6800 systems, this value will be in the range `a` through `d`. For Sun Fire 15K/12K systems, the value will be in the range `a` through `r`.

10. **Repeat Step 9 for each node you want to include in the partition.**

11. **When you have added all the switches and nodes to the partition, conclude nested sections and then the fabric description. The terminating sequence is as follows:**

    - `</partition_members>`
    - `</partitions>`
    - `</fabric>`

Configuring a Dual-Controller Cluster

This section will describe two dual-controller configurations, one with four nodes and two switches and the other with eight nodes and four switches. Separate templates are used for two-switch and four-switch configurations because the rules for allocating switches to partitions is not the same for two-switch configurations as it is for four-switch configurations. In brief, the differences are:

■ If your cluster contains two switches, both switches must be shared by the two partitions. That is, each switch must be assigned to both partitions. This is shown in FIGURE 6-3.

■ If your cluster contains four switches, two of the switches must be added to one partition and the other two must be in the other partition. Both links for a given WCI module must be allocated to the same partition. For example, both links in chassis slot IB8 must be in one partition and both links in slot IB9 must be in the other partition. This is shown in FIGURE 6-4.

Configuring a Four-Node, Two-Switch Dual-Controller Cluster

CODE EXAMPLE B-2 contains a template for the same four-node, two-switch cluster as described in "Configuring a Single-Controller Cluster" on page 176, but configured with dual controllers for use by Sun Cluster applications.

CODE EXAMPLE B-2   Four-Node, Two-Switch, Dual-Controller Configuration

```
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE fabric SYSTEM "fabric.dtd">
<fabric>
      <fname>name_of_fabric</fname>
      <config_file>XML_config_name.xml</config_file>
      <members>
            <switch_node>
             <node>
                     <sc_name>switch1_sc_hostname</sc_name>
                     <sc_user_name>switch1_sc_username</sc_user_name>
                     <sc_password>switch1_sc_password</sc_password>
                     <chassis_type>wcix_switch</chassis_type>
                     <info>description of switch1 location</info>
             </node>
             </switch_node>
```

```
<switch_node>
<node>
        <sc_name>switch2_sc_hostname</sc_name>
        <sc_user_name>switch2_sc_username</sc_user_name>
        <sc_password>switch2_sc_password</sc_password>
        <chassis_type>wcix_switch</chassis_type>
        <info>description of switch2 location</info>
</node>
</switch_node>
<rsm_node>
<node>
        <sc_name>node1_sc_hostname</sc_name>
        <sc_user_name>node1_sc_username</sc_user_name>
        <sc_password>node1_sc_password</sc_password>
        <chassis_type>S24</chassis_type>
        <info>description of node1 location</info>
</node>
    <domain_name>node1_domain_name</domain_name>
    <hostname>node1_hostname-domain</hostname>
    <host_user>node1_username</host_user>
    <host_password>node1_password</host_password>
</rsm_node>
<rsm_node>
<node>
        <sc_name>node2_sc_hostname</sc_name>
        <sc_user_name>node2_sc_username</sc_user_name>
        <sc_password>node2_sc_password</sc_password>
        <chassis_type>S24</chassis_type>
        <info>description of node2 location</info>
</node>
    <domain_name>node2_domain_name</domain_name>
    <hostname>node2_hostname-domain</hostname>
    <host_user>node2_username</host_user>
    <host_password>node2_password</host_password>
</rsm_node>

<rsm_node>
<node>
        <sc_name>node3_sc_hostname</sc_name>
        <sc_user_name>node3_sc_username</sc_user_name>
        <sc_password>node3_sc_password</sc_password>
        <chassis_type>S24</chassis_type>
        <info>description of node3 location</info>
</node>
    <domain_name>node3_domain_name</domain_name>
    <hostname>node3_hostname-domain</hostname>
    <host_user>node3_username</host_user>
```

```
                      <host_password>node3_password</host_password>
            </rsm_node>

            <rsm_node>
            <node>
                      <sc_name>node4_sc_hostname</sc_name>
                      <sc_user_name>node4_sc_username</sc_user_name>
                      <sc_password>node4_sc_password</sc_password>
                      <chassis_type>S24</chassis_type>
                      <info>description of node4 location</info>
            </node>
                      <domain_name>node4_domain_name</domain_name>
                      <hostname>node4_hostname-domain</hostname>
                      <host_user>node4_username</host_user>
                      <host_password>node4_password</host_password>
            </rsm_node>
    <partitions>
            <partition type="RSM" topology="WcixSwitch">
                      <pname>partition1_name</pname>
                      <wci_striping_level>wci_stripe_level</wci_striping_level>
                      <link_striping_level>link_stripe_level</link_striping_level>
                      <partition_members>
                          <switch_partition_member>
                                <sc_name>switch1_sc_hostname</sc_name>
                          </switch_partition_member>
                          <switch_partition_member>
                                <sc_name>switch2_sc_hostname</sc_name>
                          </switch_partition_member>
                          <node_partition_member>
                                <sc_name>node1_sc_hostname</sc_name>
                                <domain_name>node1_domain_name</domain_name>
                          </node_partition_member>
                          <node_partition_member>
                                <sc_name>node2_sc_hostname</sc_name>
                                <domain_name>node2_domain_name</domain_name>
                          </node_partition_member>
                          <node_partition_member>
                                <sc_name>node3_sc_hostname</sc_name>
                                <domain_name>node3_domain_name</domain_name>
                          </node_partition_member>
                          <node_partition_member>
                                <sc_name>node4_sc_hostname</sc_name>
                                <domain_name>node4_domain_name</domain_name>
                          </node_partition_member>
                      </partition_members>

            <partition type="RSM" topology="WcixSwitch">
```

```
                    <pname>partition2_name</pname>
                    <wci_striping_level>wci_stripe_level</wci_striping_level>
                    <link_striping_level>link_stripe_level</link_striping_level>
                    <partition_members>
                        <switch_partition_member>
                            <sc_name>switch1_sc_hostname</sc_name>
                        </switch_partition_member>
                        <switch_partition_member>
                            <sc_name>switch2_sc_hostname</sc_name>
                        </switch_partition_member>
                        <node_partition_member>
                            <sc_name>node1_sc_hostname</sc_name>
                            <domain_name>node1_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                            <sc_name>node2_sc_hostname</sc_name>
                            <domain_name>node2_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                            <sc_name>node3_sc_hostname</sc_name>
                            <domain_name>node3_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                            <sc_name>node4_sc_hostname</sc_name>
                            <domain_name>node4_domain_name</domain_name>
                        </node_partition_member>
                    </partition_members>
                </partition>
            </partitions>
</fabric>
```

Editing a Four-Node, Two-Switch Dual-Controller Template

The following example uses the same combination of nodes and switches as in the single-controller example.

Note – For three-node direct-connect configurations with dual controllers, both `<wci_striping_level>` and `<link_striping_level>` must be set to 1.

▼ To Create a Two-Switch Dual Controller Configuration

1. **Peform the same steps as described in the single-controller example.**

   > **Note –** When creating the first partition in a dual-controller configuration, set the `<wci_striping_level>` value to 1. Both partitions in a dual-controller configuration must have WCI striping set to 1.

2. **Create a second partition, giving it a unique name.**

3. **Set the** `<wci_striping_level>` **to** 1 **and the** `<link_striping_level>` **to** 2 **in the second partition.**

4. **Include the same nodes and switches in the second partition as are in the first partition.**

5. **When you have added all the switches and nodes to the partition, conclude the nested sections and then the fabric description. The terminating sequence is just as in the single-controller example:**

   - `</partition_members>`
   - `</partitions>`
   - `</fabric>`

Configuring an Eight-Node, Four-Switch Dual-Controller Cluster

CODE EXAMPLE B-3 contains a template for an eight-node, four-switch cluster configured with dual controllers for use by Sun Cluster applications.

CODE EXAMPLE B-3   Eight-Node, Four-Switch Dual-Controller Configuration

```
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE fabric SYSTEM "fabric.dtd">
<fabric>
      <fname>name_of_fabric</fname>
      <config_file>XML_config_name.xml</config_file>
      <members>
            <switch_node>
             <node>
                     <sc_name>switch1_sc_hostname</sc_name>
                     <sc_user_name>switch1_sc_username</sc_user_name>
                     <sc_password>switch1_sc_password</sc_password>
```

```
                <chassis_type>wcix_switch</chassis_type>
                <info>description of switch1 location</info>
        </node>
        </switch_node>

        <switch_node>
        <node>
                <sc_name>switch2_sc_hostname</sc_name>
                <sc_user_name>switch2_sc_username</sc_user_name>
                <sc_password>switch2_sc_password</sc_password>
                <chassis_type>wcix_switch</chassis_type>
                <info>description of switch2 location</info>
        </node>
        </switch_node>
        <switch_node>
        <node>
                <sc_name>switch3_sc_hostname</sc_name>
                <sc_user_name>switch3_sc_username</sc_user_name>
                <sc_password>switch3_sc_password</sc_password>
                <chassis_type>wcix_switch</chassis_type>
                <info>description of switch3 location</info>
        </node>
        </switch_node>
        <switch_node>
        <node>
                <sc_name>switch4_sc_hostname</sc_name>
                <sc_user_name>switch4_sc_username</sc_user_name>
                <sc_password>switch4_sc_password</sc_password>
                <chassis_type>wcix_switch</chassis_type>
                <info>description of switch4 location</info>
        </node>
        </switch_node>
        <rsm_node>
        <node>
                <sc_name>node1_sc_hostname</sc_name>
                <sc_user_name>node1_sc_username</sc_user_name>
                <sc_password>node1_sc_password</sc_password>
                <chassis_type>S24</chassis_type>
                <info>description of node1 location</info>
        </node>
            <domain_name>node1_domain_name</domain_name>
            <hostname>node1_hostname-domain</hostname>
            <host_user>node1_username</host_user>
            <host_password>node1_password</host_password>
        </rsm_node>
        <rsm_node>
        <node>
```

CODE EXAMPLE B-3 Eight-Node, Four-Switch Dual-Controller Configuration *(Continued)*

```
            <sc_name>node2_sc_hostname</sc_name>
            <sc_user_name>node2_sc_username</sc_user_name>
            <sc_password>node2_sc_password</sc_password>
            <chassis_type>S24</chassis_type>
            <info>description of node2 location</info>
    </node>
        <domain_name>node2_domain_name</domain_name>
        <hostname>node2_hostname-domain</hostname>
        <host_user>node2_username</host_user>
        <host_password>node2_password</host_password>
    </rsm_node>

    <rsm_node>
    <node>
            <sc_name>node3_sc_hostname</sc_name>
            <sc_user_name>node3_sc_username</sc_user_name>
            <sc_password>node3_sc_password</sc_password>
            <chassis_type>S24</chassis_type>
            <info>description of node3 location</info>
    </node>
        <domain_name>node3_domain_name</domain_name>
        <hostname>node3_hostname-domain</hostname>
        <host_user>node3_username</host_user>
        <host_password>node3_password</host_password>
    </rsm_node>

    <rsm_node>
    <node>
            <sc_name>node4_sc_hostname</sc_name>
            <sc_user_name>node4_sc_username</sc_user_name>
            <sc_password>node4_sc_password</sc_password>
            <chassis_type>S24</chassis_type>
            <info>description of node4 location</info>
    </node>
        <domain_name>node4_domain_name</domain_name>
        <hostname>node4_hostname-domain</hostname>
        <host_user>node4_username</host_user>
        <host_password>node4_password</host_password>
    </rsm_node>
    <rsm_node>
    <node>
            <sc_name>node5_sc_hostname</sc_name>
            <sc_user_name>node5_sc_username</sc_user_name>
            <sc_password>node5_sc_password</sc_password>
            <chassis_type>S24</chassis_type>
            <info>description of node5 location</info>
    </node>
```

```
                    <domain_name>node4_domain_name</domain_name>
                    <hostname>node5_hostname–domain</hostname>
                    <host_user>node5_username</host_user>
                    <host_password>node5_password</host_password>
            </rsm_node>
            <rsm_node>
            <node>
                    <sc_name>node6_sc_hostname</sc_name>
                    <sc_user_name>node6_sc_username</sc_user_name>
                    <sc_password>node6_sc_password</sc_password>
                    <chassis_type>S24</chassis_type>
                    <info>description of node6 location</info>
            </node>
                    <domain_name>node6_domain_name</domain_name>
                    <hostname>node6_hostname–domain</hostname>
                    <host_user>node6_username</host_user>
                    <host_password>node6_password</host_password>
            </rsm_node>
            <rsm_node>
            <node>
                    <sc_name>node7_sc_hostname</sc_name>
                    <sc_user_name>node7_sc_username</sc_user_name>
                    <sc_password>node7_sc_password</sc_password>
                    <chassis_type>S24</chassis_type>
                    <info>description of node7 location</info>
            </node>
                    <domain_name>node7_domain_name</domain_name>
                    <hostname>node7_hostname–domain</hostname>
                    <host_user>node7_username</host_user>
                    <host_password>node7_password</host_password>
            </rsm_node>
            <rsm_node>
            <node>
                    <sc_name>node8_sc_hostname</sc_name>
                    <sc_user_name>node8_sc_username</sc_user_name>
                    <sc_password>node8_sc_password</sc_password>
                    <chassis_type>S24</chassis_type>
                    <info>description of node8 location</info>
            </node>
                    <domain_name>node8_domain_name</domain_name>
                    <hostname>node8_hostname–domain</hostname>
                    <host_user>node8_username</host_user>
                    <host_password>node8_password</host_password>
            </rsm_node>
    <partitions>
            <partition type="RSM" topology="WcixSwitch">
                    <pname>partition1_name</pname>
```

```
                    <wci_striping_level>wci_stripe_level</wci_striping_level>
                    <link_striping_level>link_stripe_level</link_striping_level>
                    <partition_members>
                        <switch_partition_member>
                            <sc_name>switch1_sc_hostname</sc_name>
                        </switch_partition_member>
                        <switch_partition_member>
                            <sc_name>switch2_sc_hostname</sc_name>
                        </switch_partition_member>
                        <node_partition_member>
                            <sc_name>node1_sc_hostname</sc_name>
                            <domain_name>node1_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                            <sc_name>node2_sc_hostname</sc_name>
                            <domain_name>node2_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                            <sc_name>node3_sc_hostname</sc_name>
                            <domain_name>node3_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                            <sc_name>node4_sc_hostname</sc_name>
                            <domain_name>node4_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                            <sc_name>node5_sc_hostname</sc_name>
                            <domain_name>node5_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                            <sc_name>node6_sc_hostname</sc_name>
                            <domain_name>node6_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                            <sc_name>node7_sc_hostname</sc_name>
                            <domain_name>node7_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                            <sc_name>node8_sc_hostname</sc_name>
                            <domain_name>node8_domain_name</domain_name>
                        </node_partition_member>
                    </partition_members>

            <partition type="RSM" topology="WcixSwitch">
                    <pname>partition2_name</pname>
                    <wci_striping_level>wci_stripe_level</wci_striping_level>
                    <link_striping_level>link_stripe_level</link_striping_level>
```

```
                        <partition_members>
                            <switch_partition_member>
                                <sc_name>switch3_sc_hostname</sc_name>
                            </switch_partition_member>
                            <switch_partition_member>
                                <sc_name>switch4_sc_hostname</sc_name>
                            </switch_partition_member>
                            <node_partition_member>
                                <sc_name>node1_sc_hostname</sc_name>
                                <domain_name>node1_domain_name</domain_name>
                            </node_partition_member>
                            <node_partition_member>
                                <sc_name>node2_sc_hostname</sc_name>
                                <domain_name>node2_domain_name</domain_name>
                            </node_partition_member>
                            <node_partition_member>
                                <sc_name>node3_sc_hostname</sc_name>
                                <domain_name>node3_domain_name</domain_name>
                            </node_partition_member>
                            <node_partition_member>
                                <sc_name>node4_sc_hostname</sc_name>
                                <domain_name>node4_domain_name</domain_name>
                            </node_partition_member>
                            <node_partition_member>
                                <sc_name>node5_sc_hostname</sc_name>
                                <domain_name>node5_domain_name</domain_name>
                            </node_partition_member>
                            <node_partition_member>
                                <sc_name>node6_sc_hostname</sc_name>
                                <domain_name>node6_domain_name</domain_name>
                            </node_partition_member>
                            <node_partition_member>
                                <sc_name>node7_sc_hostname</sc_name>
                                <domain_name>node7_domain_name</domain_name>
                            </node_partition_member>
                            <node_partition_member>
                                <sc_name>node8_sc_hostname</sc_name>
                                <domain_name>node8_domain_name</domain_name>
                            </node_partition_member>
                        </partition_members>
                </partition>
        </partitions>
</fabric>
```

Editing an Eight-Node, Four-Switch Dual-Controller Template

The following procedure is for editing the template shown in CODE EXAMPLE B-3.

▼ To Create a Four-Switch, Dual-Controller Configuration

● **Perform the same steps as described in the two-switch, dual-controller example, with one exception—put two switches in the first partition and the other two switches in the second partition.**

Note – Remember that both links for a given WCI module must be in the same partition. For example, both links in chassis slot IB8 must be in one partition and both links in slot IB9 must be in the other partition.

Using XML Files to Remove Nodes and Partitions

The `wcfmconf` command can be used to selectively remove nodes from partitions, to remove nodes from partitions and fabrics, and to remove partitions from fabrics. The command takes as input an XML file that specifies what is to be removed. The syntax for using `wcfmconf` for this purpose is:

```
# cd /opt/SUNWwcfm/bin
# ./wcfmconf filename
```

Sample XML files for each of the three removal operations are provided below.

Removing a Node From a Partition

CODE EXAMPLE B-4 shows an example of an XML file that specifies removal of a node (testnode1) and a switch (testswitch1) from the partition testpart1. Although they are removed from the partition, they are still members of the fabric testfab.

CODE EXAMPLE B-4   Sample XML File for Removing a Node From a Partition

```
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE fabric SYSTEM "fabric.dtd">
<fabric>
    <fname>testfab</fname>
    <config_file>testfab.xmlrmnp</config_file>
    <members>
    </members>
    <partitions>
        <partition type="RSM" topology="WcixSwitch">
            <pname>testpart1</pname>
            <remove_partition_members>
                <node_partition_member>
                    <sc_name>testnode1</sc_name>
                    <domain_name>a</domain_name>
                </node_partition_member>>
                <switch_partition_member>
                    <sc_name>testswitch1</sc_name>
                    <domain_name></domain_name>
                </switch_partition_member>>
        </partition>
    </partitions>
</fabric>
```

The XML file has the same basic construction as a fabric configuration file, except the <members> section is empty and the <partition> section contains the removal instructions instead of member descriptions. The chief features of the remove-node-from-partition file are summarized below:

1. **Use the** <pname> </pname> **line to name the partition that is to be downgraded by one or more nodes.**

2. **Below this line, construct a section that identifies the node(s) to be removed. This section begins and ends with the** <remove_partition_members> </remove_partition_members> **tags.**

3. **For each node to be removed, create a** <node_partition_member> **subsection that specifies the node's SC name and its domain name.**

4. **For each switch to be removed, create a** <switch_partition_member> **subsection that specifies the switch's SC name.**

Removing a Node From All Partitions and From a Fabric

CODE EXAMPLE B-5 shows an example of an XML file that specifies removal of a node (testnode1) and a switch (testswitch1) from the fabric testfab. This automatically causes them to be removed from all partitions of which they are members.

CODE EXAMPLE B-5   Sample XML File for Removing a Node and a Switch From a Fabric

```
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE fabric SYSTEM "fabric.dtd">
<fabric>
    <fname>testfab</fname>
    <config_file>testfab.xmlrmnpf</config_file>
    <members>
    </members>
    <remove_members>
        <remove_node_member>
            <sc_name>testnode1</sc_name>
            <domain_name>a</domain_name>
        </remove_node_member>
        <remove_switch_member>
            <sc_name>testswitch1</sc_name>
        </remove_switch_member>
    </remove_members>
</fabric>
```

This XML file includes an empty `<members>` section, but does not include a `<partition>` section. Instead, the `<members>` section is immediately followed by the `<remove_members>` section, which lists the nodes and/or switches to be removed.

Removing a Partition From a Fabric

CODE EXAMPLE B-6 shows an example of an XML file that specifies removal of a partition (testpart1) from the fabric testfab. This automatically removes all members of the partition (nodes and switches) from the partition, but not from the fabric.

CODE EXAMPLE B-6   Sample XML File for Removing a Partition From a Fabric

```
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE fabric SYSTEM "fabric.dtd">
<fabric>
    <fname>testfab</fname>
```

```
     <config_file>testfab.xmlrmpf</config_file>
     <members>
     </members>
     <partitions>
         <remove_partition>
             <pname>testpart1</pname>
         </remove_partition>
     </partitions>
</fabric>
```

This XML file includes an empty `<members>` section plus a `<partitions>` section. The `<partitions>` section contains one or more `<remove_partition>` subsections, each of which names a partition to be removed. The `<partitions>` section does not contain a `<partition>` section

Allocating Particular WCI Links or Link Pairs to a Partition

When you create a partition as described in earlier examples, the FM automatically configures an optimal set of link paths according to the available hardware and physical connections it discovers. You have the option, however, of explicitly controlling the selection of WCI ASICS or link pairs by using the following XML tags when creating a fabric configuration file:

- `<reserve_wcis>` – Use this tag to select specific WCI ASICS to be included in a partition.
- `<reserve_links>` – Use this tag to select specific links to be included in a partition.

Adding Specific WCI Components to a Partition

CODE EXAMPLE B-7 shows a fabric configuration template that includes the `<reserve_wcis>` tag. This template represents two Sun Fire 6800 nodes that are in a direct-connect configuration, with level 2 striping. An explanation of the `<reserve_wcis>` portion is provided following the template.

CODE EXAMPLE B-7   Sample Fabric Configuration File With Explicit WCI Selection

```
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE fabric SYSTEM "fabric.dtd">
<fabric>
        <fname>testfab2</fname>
        <config_file>testfab2.xml</config_file>
        <members>
                <rsm_node>
                    <node>
                        <sc_name>node1_sc_hostname</sc_name>
                        <sc_user_name>node1_sc_username</sc_user_name>
                        <sc_password>node1_sc_password</sc_password>
                        <chassis_type>S24</chassis_type>
                        <info>description of node1 location</info>
                    </node>
                    <domain_name>node1_domain_name</domain_name>
                    <hostname>node1_hostname_domain</hostname>
                    <host_user>node1_username</host_user>
                    <host_password>node1_password</host_password>
                </rsm_node>
                <rsm_node>
                    <node>
                        <sc_name>node2_sc_hostname</sc_name>
                        <sc_user_name>node2_sc_username</sc_user_name>
                        <sc_password>node2_sc_password</sc_password>
                        <chassis_type>S24</chassis_type>
                        <info>description of node2 location</info>
                    </node>
                    <domain_name>node2_domain_name</domain_name>
                    <hostname>node2_hostname_domain</hostname>
                    <host_user>node2_username</host_user>
                    <host_password>node2_password</host_password>
                </rsm_node>
        </members>
        <partitions>
                <partition type="RSM" topology="DirectConnect">
                    <pname>partition1_name</pname>
                    <stripe_level>2</stripe_level>
                    <partition_members>
                        <node_partition_member>
```

```
                                <sc_name>node1_sc_hostname</sc_name>
                                <domain_name>node1_domain_name</domain_name>
                        </node_partition_member>
                        <node_partition_member>
                                <sc_name>node2_sc_hostname</sc_name>
                                <domain_name>node2_domain_name</domain_name>
                        </node_partition_member>
                </partition_members>
                <reserve_wcis>
                        <wci_end_point>
                                <sc_name>node1_sc_hostname</sc_name>
                                <domain_name>node1_domain_name</domain_name>
                                        <wci>
                                                <slot>8</slot>
                                                <wci_id>1</wci_id>
                                        </wci>
                        </wci_end_point>
                        <wci_end_point>
                                <sc_name>node2_sc_hostname</sc_name>
                                <domain_name>node2_domain_name</domain_name>
                                        <wci>
                                                <slot>9</slot>
                                                <wci_id>1</wci_id>
                                        </wci>
                        </wci_end_point>
                </reserve_wcis>
            </partition>
        </partitions>
</fabric>
```

▼ To Add Specific WCIs to a Partition

1. **Go to the end of the partition members portion of the template—that is, following the line containing** `</partition_members>`**.**

2. **Begin the WCI reservation section with the** `<reserve_wcis>` **tag.**

3. **Follow this with a sequence of lines that describe a WCI endpoint for a particular domain.**

   CODE EXAMPLE B-8 shows the selection of the WCI in slot IB8 for the domain `testnode1-a`.

   CODE EXAMPLE B-8   Sample WCI Specification for a Sun Fire 6800 Node

   ```
   <reserve_wcis>
       <wci_end_point>
           <sc_name>testnode1</sc_name>
           <domain_name>a</domain_name>
               <wci>
                   <slot>8</slot>
                   <wci_id>1</wci_id>
               </wci>
       </wci_end_point>
   ```

4. **Repeat Step 3 for all other WCI components that you want to select for use by the partition members.**

5. **Conclude the WCI reservation section with the `</reserve_wcis>` tag.**

   The `<slot>` and `<wci_id>` content will be different for Sun Fire 15K/12K systems and for switches. CODE EXAMPLE B-9 and CODE EXAMPLE B-10 illustrate these differences.

   For Sun Fire 15K/12K nodes, the `<slot>` value can be in the range 0 – 17.

   CODE EXAMPLE B-9   Sample WCI Specification for a Sun Fire 15K/12K Node

   ```
   <reserve_wcis>
       <wci_end_point>
           <sc_name>testnode1</sc_name>
           <domain_name>m</domain_name>
               <wci>
                   <slot>10</slot>
                   <wci_id>1</wci_id>
               </wci>
       </wci_end_point>
   ```

   For Sun Fire Link switches, simply specify the switch SC name.

   CODE EXAMPLE B-10 Sample WCI Specification for a Sun Fire Switch

   ```
   <reserve_wcis>
       <wci_end_point>
           <sc_name>testnode1</sc_name>
       </wci_end_point>
   ```

Adding Link Pairs to a Partition in a Direct-Connect Configuration

CODE EXAMPLE B-11 shows a fabric configuration template that includes the
`<reserve_links>` tag. This template represents two Sun Fire 6800 nodes that are in
a direct-connect configuration, with level 2 striping. An explanation of the
`<reserve_links>` portion is provided following the template.

CODE EXAMPLE B-11   Sample Fabric Configuration File With Explicit Link Pair Selection

```
<?xml version="1.0" standalone="no" ?>
<!DOCTYPE fabric SYSTEM "fabric.dtd">
<fabric>
        <fname>testfab2</fname>
        <config_file>testfab2.xml</config_file>
        <members>
                <rsm_node>
                    <node>
                        <sc_name>node1_sc_hostname</sc_name>
                        <sc_user_name>node1_sc_username</sc_user_name>
                        <sc_password>node1_sc_password</sc_password>
                        <chassis_type>S24</chassis_type>
                        <info>description of node1 location</info>
                    </node>
                    <domain_name>node1_domain_name</domain_name>
                    <hostname>node1_hostname_domain</hostname>
                    <host_user>node1_username</host_user>
                    <host_password>node1_password</host_password>
                </rsm_node>
                <rsm_node>
                    <node>
                        <sc_name>node2_sc_hostname</sc_name>
                        <sc_user_name>node2_sc_username</sc_user_name>
                        <sc_password>node2_sc_password</sc_password>
                        <chassis_type>S24</chassis_type>
                        <info>description of node2 location</info>
                    </node>
                    <domain_name>node2_domain_name</domain_name>
                    <hostname>node2_hostname_domain</hostname>
                    <host_user>node2_username</host_user>
                    <host_password>node2_password</host_password>
                </rsm_node>
        </members>
        <partitions>
                <partition type="RSM" topology="DirectConnect">
                    <pname>partition1_name</pname>
                    <stripe_level>2</stripe_level>
```

```
                  <partition_members>
                      <node_partition_member>
                          <sc_name>node1_sc_hostname</sc_name>
                          <domain_name>node1_domain_name</domain_name>
                      </node_partition_member>
                      <node_partition_member>
                          <sc_name>node2_sc_hostname</sc_name>
                          <domain_name>node2_domain_name</domain_name>
                      </node_partition_member>
                  </partition_members>
                  <reserve_links>
                      <reserve_link_pair>
                          <wci_end_point>
                              <sc_name>node1_sc_hostname</sc_name>
                              <domain_name>node1_domain_name</domain_name>
                                      <wci>
                                          <slot>8</slot>
                                          <wci_id>1</wci_id>
                                      </wci>
                          </wci_end_point>
                          <link_num>0</link_num>
                          <wci_end_point>
                              <sc_name>node2_sc_hostname</sc_name>
                              <domain_name>node2_domain_name</domain_name>
                                      <wci>
                                          <slot>8</slot>
                                          <wci_id>1</wci_id>
                                      </wci>
                          </wci_end_point>
                      </reserve_link_pair>
                      <reserve_link_pair>
                          <wci_end_point>
                              <sc_name>node1_sc_hostname</sc_name>
                              <domain_name>node1_domain_name</domain_name>
                                      <wci>
                                          <slot>9</slot>
                                          <wci_id>1</wci_id>
                                      </wci>
                          </wci_end_point>
                          <link_num>2</link_num>
                          <wci_end_point>
                              <sc_name>node2_sc_hostname</sc_name>
                              <domain_name>node2_domain_name</domain_name>
                                      <wci>
                                          <slot>9</slot>
                                          <wci_id>1</wci_id>
                                      </wci>
```

```
                           </wci_end_point>
                           <link_num>2</link_num>
                        </reserve_link_pair>
                  </reserve_links>
               </partition>
         </partitions>
</fabric>
```

▼ To Add Specific Pairs of Links to a Partition

1. **Go to the end of the partition members portion of the template—that is, following the line containing** `</partition_members>`**.**

2. **Begin the link pair reservation section with the** `<reserve_links>` **tag.**

   This section contains two subsections, each beginning with a `<reserve_link_pair>` tag.

3. **In the first** `<reserve_link_pair>` **subsection, specify two WCI end points and assign a link number to them.** CODE EXAMPLE B-12 **shows one of the two link pairs shown in** CODE EXAMPLE B-11**.**

   Note that the WCI end points are specified in the same way as shown in"Adding Specific WCI Components to a Partition" on page 198. Each WCI end point specification is followd by the associated link number.

   CODE EXAMPLE B-12 Specifying End Points for a Link Pair in a Direct-Connect Configuration

```
<reserve_links>
    <reserve_link_pair>
        <wci_end_point>
            <sc_name>testnode1</sc_name>
            <domain_name>a</domain_name>
            <wci>
                <slot>8</slot>
                <wci_id>1</wci_id>
            </wci>
        </wci_end_point>
        <link_num>0</link_num>
        <wci_end_point>
            <sc_name>testnode2</sc_name>
            <domain_name>a</domain_name>
            <wci>
                <slot>8</slot>
                <wci_id>1</wci_id>
            </wci>
```

```
        </wci_end_point>
        <link_num>2</link_num>
    </reserve_link_pair>
```

4. **Repeat this pattern for each link pair and conclude the link reservation section with the** `</reserve_links>` **tag.**

Adding Link Pairs to a Partition in a Node-Switch Configuration

CODE EXAMPLE B-13 shows a link pair specification when the end points connect a node to a switch. This example includes only the `<reserve_links>` section of the specification.

CODE EXAMPLE B-13 Specifying End Points for a Link Pair That Includes a Node and a Switch

```
<reserve_links>
    <reserve_link_pair>
        <wci_end_point>
            <sc_name>testnode2</sc_name>
            <domain_name>a</domain_name>
            <wci>
                <slot>8</slot>
                <wci_id>1</wci_id>
            </wci>
        </wci_end_point>
        <link_num>2</link_num>
        <wci_end_point>
            <sc_name>testswitch2</sc_name>
        </wci_end_point>
        <link_num>2</link_num>
    </reserve_link_pair>
```

For the switch specification, simply provide the switch SC name.

The Sun Fire Link Fabric DTD

The XML specification that the configuration files use is defined in a document named `fabric.dtd`. The contents of that file are shown in CODE EXAMPLE B-14.

> **Note –** The following elements in the DTD are reserved for use by Sun
> Microsystems technical support personnel: discovery_links,
> discovery_link_pair, and link_num. These elements are not supported for use
> by any one other than authorized Sun Microsystems personnel.

CODE EXAMPLE B-14   DTD for Sun Fire Link Fabric Configuration Elements

```
<!--                                                              -->
<!-- Copyright 2000-2002 by Sun Microsystems, Inc. All rights reserved.   -->
<!-- Use is subject to license terms.                            -->
<!--                                                              -->
<!-- @version      "@(#)fabric.dtd 1.11    01/01/04 SMI"          -->
<!--                                                              -->
<!-- This DTD defines a fabric, at the fabric level. Meaning the  -->
<!-- partition's private data string should be treated as character data  -->
<!-- and not parsed as XML.                                       -->

<!-- Entities used within this DTD                                -->
<!ENTITY % partition_type      "RSM | SSM">
<!ENTITY % partition_topology  "DirectConnect | WcixSwitch">
<!ELEMENT fabric               (last_date_time?, fname, config_file, members,
                                remove_members?, partitions?, discovery_data?)>
<!ELEMENT last_date_time      (#PCDATA)>
<!ELEMENT fname               (#PCDATA)>
<!ELEMENT config_file         (#PCDATA)>
<!ELEMENT members             ( (switch_node | ssm_node | rsm_node)* )>
<!ELEMENT node                (sc_name, sc_user_name, sc_password,
                                chassis_type?, fm_nodeid?, info?, properties?)>
<!ELEMENT switch_node         (node)>
<!ELEMENT ssm_node            (node, domain_name)>
<!ELEMENT rsm_node            (node, domain_name, hostname,
                                host_user, host_password)>

<!ELEMENT sc_name             (#PCDATA)>
<!ELEMENT domain_name         (#PCDATA)>
<!ELEMENT sc_user_name        (#PCDATA)>
<!ELEMENT sc_password         (#PCDATA)>

<!-- The chassis_type element should have one of the following values.    -->
<!-- S72, S72_CENTRAL_SWITCH, S24, S12, S12_DESK_SIDE, S8, WCIX_SWITCH    -->
<!ELEMENT chassis_type        (#PCDATA)>
<!ELEMENT hostname            (#PCDATA)>
<!ELEMENT host_user           (#PCDATA)>
<!ELEMENT host_password       (#PCDATA)>
<!ELEMENT fm_nodeid           (#PCDATA)>
```

CODE EXAMPLE B-14   DTD for Sun Fire Link Fabric Configuration Elements

```
<!ELEMENT info                  (#PCDATA)>
<!ELEMENT properties            (name_value_pair*)>
<!ELEMENT name_value_pair       (name, value)>
<!ELEMENT name                  (#PCDATA)>
<!ELEMENT value                 (#PCDATA)>
<!ELEMENT partitions            ( (partition | remove_partition)* )>
<!ELEMENT partition             (pname, partition_id?,
                                ((wci_striping_level, link_striping_level) |
                                  stripe_level)?, partition_members?,
                                  remove_partition_members?,
                                  (reserve_wcis | reserve_links)?,
                                  discover_links?, private_data? )>
<!ELEMENT remove_partition      (pname)>
<!ELEMENT remove_members        ( (remove_node_member |
                                   remove_switch_member)* )>
<!ELEMENT remove_node_member    (sc_name, domain_name)>
<!ELEMENT remove_switch_member  (sc_name)>
<!ATTLIST partition
         type                   (%partition_type;)    #REQUIRED
         topology               (%partition_topology;) #REQUIRED>
<!ELEMENT pname                 (#PCDATA)>
<!ELEMENT partition_id          (#PCDATA)>
<!ELEMENT wci_striping_level    (#PCDATA)>
<!ELEMENT link_striping_level   (#PCDATA)>
<!ELEMENT stripe_level          (#PCDATA)>
<!ELEMENT partition_members           ((node_partition_member |
                                        switch_partition_member)*)>
<!ELEMENT remove_partition_members   ((node_partition_member |
                                        switch_partition_member)*)>
<!ELEMENT node_partition_member      (sc_name, domain_name)>
<!ELEMENT switch_partition_member    (sc_name)>
<!ELEMENT private_data               (#PCDATA)>

<!-- The reserve_wcis element allows users to specify a set of WCIs to     -->
<!-- be used within a node for a single partition definition.              -->
<!ELEMENT reserve_wcis               (wci_end_point*)>
<!ELEMENT wci                        (slot, wci_id)>
<!ELEMENT slot                       (#PCDATA)>
<!ELEMENT wci_id                     (#PCDATA)>

<!-- The reserve_links element is similar to the reserve_wcis element.     -->
<!-- It allows users to specify a set of links to use within a node for    -->
<!-- a single partition definition.                                        -->
<!ELEMENT reserve_links              (reserve_link_pair*)>
<!ELEMENT reserve_link_pair          (wci_end_point, link_num, wci_end_point,
                                      link_num)>
```

CODE EXAMPLE B-14   DTD for Sun Fire Link Fabric Configuration Elements

```
<!-- The discovery_links element allows users to specify a set -->
<!-- of links to use for a single partition definition. All links to be -->
<!-- used within the partition definition must be present. -->
<!ELEMENT discovery_links          (discovery_link_pair*)>
<!ELEMENT discovery_link_pair          (wci_end_point, link_num, wci_end_point,
                                   link_num)>
<!ELEMENT link_num                 (#PCDATA)>

<!-- The optional fields are to allow users to specify WCIX switch ports  -->
<!-- as remote endpoints.                                                 -->
<!ELEMENT wci_end_point            (sc_name, domain_name?, wci?)>
```

Index

A

alarm
    causes, 122 to 125
    colors, 75
    corrective actions, 122 to 125
    FM, 96
    icons, 73
    levels, 75
    switch, 116 to 120
architecture overview, 3 to 7
ASIC striping, 42

C

causes for alarms, 122 to 125
causes for traps, 125 to 127
chassis, 73
    data for switch chassis, 102
    viewing properties, 81
Chassis and Links view, 77
checking status of a fabric, 140
checking that a fabric is running, 19
cluster
    discovering nodes and switches, 28
    illustration, typical, 2
    password, 11
    security, 11
color of alarms, 75
command line interface, 159 to 171
commands
    `createfabric`, 18, 160

    `deletefabric`, 23, 24, 171
    `killfabrics`, 22, 23, 171
    `listfabrics`, 20, 169
    `startfabric`, 19, 161
    `stopfabric`, 22, 170
    `wcfmconf`, 159, 163
    `wcfmstat`, 140
    `wcfmver`, 142, 168
configuration
    direct-connect, 41
    dual controller, 45 to 47
    reviewing details, 70
    single controller, 43 to 45
controller
    definition, 42
    dual, 45 to 47, 175
    RSM, 96
    single, 43 to 45, 175
controller unit data, 104
corrective actions
    for alarms, 122 to 125
    for traps, 125 to 127
`createfabric`, 18, 160
customer data, 114
customer data for switch FRUs, 114

D

`deletefabric`, 23, 24, 171
deleting
    fabric, 61
    node, 59, 60