Sun Fire™ B1600 Blade System Chassis Software Setup Guide

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054 U.S.A.
650-960-1300

Part No. 817-4603-11
February 2004, Revision A

Send comments about this document to: docfeedback@sun.com

Contents

Preface

This manual tells you how to configure the software on the components of the Sun Fire B1600 blade system chassis to enable you to integrate the system chassis in to your network.

The manual is intended for experienced Solaris and Linux system administrators.

Before You Read This Book

Before performing the instructions in this manual, make sure you have installed the blade system chassis into a rack and connected all of the cables required. For information on how to install the system hardware, read the *Sun Fire B1600 Blade System Chassis Hardware Installation Guide.*

How This Book Is Organized

Chapter 1 provides an overview of the software for the Sun Fire B1600 blade system chassis and tells you what you need to do before following the instructions in the rest of the manual.

Chapter 2 tells you how to perform the preliminary system chassis setup tasks.

Chapter 3 tells you how to perform the quickest configuration of the system chassis if you want to install it into a simple network without enforcing any separation between your data and management networks.

Chapter 4 tells you how to power on a server blade, access its console, and run preliminary diagnostics.

Chapter 5 tells you how to introduce the system chassis into a network environment that separates data and management traffic.

Chapter 6 tells you how to refine the configuration that you performed in Chapter 5 by configuring the system chassis to permit secure management of the blades directly from the management network.

Chapter 7 is intended for ISPs (Internet Service Providers). It tells you how to assign server blades to different customers (referred to as server blade tenants) and enable them to manage their own blades without having access to other customers' blades.

Chapter 8 is provided for customers whose network infrastructure does not fully support VLANs or for customers who want to minimize the wider network administration tasks associated with integrating a blade system chassis into their existing environment.

Appendix A tells you how to perform certain tasks on the switch that you will need to perform when you follow the instructions in the subsequent chapters.

Appendix B supplements the instructions in the *Solaris Advanced Installation Guide* and the *DHCP Administration Guide*. It enables you to complete the configuration of the DHCP server on your data network so that the server blades on the system chassis can receive their IP addresses dynamically.

Appendix C provides information on using Web Start Flash Archives to replicate a server blade's Operating Environment and application software on other blades.

Appendix D lists the commands available from the System Controller's `sc>` prompt.

Appendix E provides detailed information on the relationship between the active and standby System Controllers.

After You Read This Book

After you read this book you may need to consult two other manuals:

- For information about configuring Linux and Solaris x86 server blades, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.
- For further information about using the command-line interface to the System Controller on the chassis, refer to the *Sun Fire B1600 Blade System Chassis Administration Guide*.
- For further information about managing the integrated switches on the chassis, refer to the *Sun Fire B1600 Blade System Chassis Switch Administration Guide.* This manual describes the hardware and architecture of the integrated switch (Chapter 1). It also tells you how to perform the initial configuration the switch (Chapter 2), how to manage the switch using the web Graphical User Interface and/or SNMP (Chapter 3), and how to use all the commands available for managing the switch from the command-line interface (Chapter 4).

Using UNIX Commands

This document does not contain information on basic UNIX® commands and procedures.

See either of the following for this information:

- *Solaris Handbook for Sun Peripherals*
- AnswerBook2™ online documentation for the Solaris™ operating environment

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	*machine-name*`%`
C shell superuser	*machine-name*`#`
Bourne shell and Korn shell	`$`
Bourne shell and Korn shell superuser	`#`
System Controller shell	sc>
Integrated switch shell	Console#

Related Documentation

Application	Title
Compliance and safety	*Sun Fire B1600 Blade System Chassis Compliance and Safety Manual*
Hardware installation overview (foldout poster)	*Sun Fire B1600 Blade System Chassis Quick Start*
Hardware installation	*Sun Fire B1600 Blade System Chassis Hardware Installation Guide*
Software installation (foldout poster)	*Sun Fire B1600 Blade System Chassis Software Setup Quick Start*
Chassis software and Solaris blade setup	*Sun Fire B1600 Blade System Chassis Software Setup Guide (this manual)*
B100x and B200x server blade installation and setup	*Sun Fire B100x and B200x Server Blade Installation and Setup Guide*
B10n content load balancing blade installation and setup	*Sun Fire B10n Content Load Balancing Blade Administration Guide*
Chassis administration and component replacement	*Sun Fire B1600 Blade System Chassis Administration Guide*
Switch administration	*Sun Fire B1600 Blade System Chassis Switch Administration Guide*
Late-breaking information	*Sun Fire B1600 Blade System Chassis Product Notes*

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 (817-4603-11) of your document in the subject line of your email.

CHAPTER 1

Preparing to Configure the System Chassis

This chapter provides an overview of the procedures for setting up the system chassis. It then introduces the system chassis and explains the roles of the System Controller and the switches. The remainder of the chapter (except the last section) tells you what you need to do before you configure the system chassis.

The last section tells you how to transition between the different user interfaces by using the #. escape sequence.

This chapter contains the following sections:

1.1 Software Setup Overview

This section summarizes the procedures for setting up the system chassis.

Note – To configure the system chassis, you need to use the command-line interface to the System Controller. From this interface, you will need to access the consoles to the two switches and the consoles to the server blades. Whenever you are at a switch or blade console, type **#.** to return to the active System Controller's `sc>` prompt.

---

1. **Create a Network Install Server (for Solaris) or a PXE boot environment (for Linux or Solaris x86) for loading the operating system onto the server blades.**

   To install the operating system onto a server blade you must boot the blade from the network. Therefore, before you proceed to set up the software on the chassis you must do the following:

   - For Sun Fire B100s server blades, follow the instructions on creating a Network Install Server in the *Solaris 8 Advanced Installation Guide* (supplied with the Solaris 8 12/02 media kit). Alternatively, if a Network Install Server already exists on your network, add the Solaris image for the server blades to the existing Network Install Server.
   - For Sun Fire B100x and B200x server blades, build a PXE boot install environment for Linux or Solaris x86 by following the instructions in the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide.* This is contained on the CD supplied with the individual blades, and it is also supplied as hard copy with any chassis that you purchase containing B100x and/or B200x server blades.

Note – If you are using N1 Provisioning software, you do not need to set up a Network Install Server or PXE boot server. Before you do the chassis software setup, read the N1 Provisioning Server 3.0, Blades Edition Implementation Guide.

If you want to use dynamically assigned IP addresses for the components of the blade system chassis, see:

- Section 1.7, "IP Information Required for the Chassis" on page 1-15 and
- Section 1.8, "Using a DHCP Server to Provide the SSC IP Addresses Automatically" on page 1-16,

and refer to the supplementary information in Appendix B to complete the configuration of both your Network Install Server and the DHCP server on the data network.

2. **Make sure you have a serial connection set up to one of the System Controllers in the chassis.**

   Alternatively, configure a DHCP server to supply IP configuration information to the System Controller. You can then access the System Controller by using telnet.

   To set up a serial connection to one of the System Controllers on the chassis, refer to the *Sun Fire B1600 Blade System Chassis Hardware Installation Guide*.

3. **Log into the System Controller and set a password and the date and time**

   You need to set a password and the date and time (see Chapter 2).

4. **Log into and set at least one password for each switch**

   For information on how to do this, see Chapter 2.

5. **Prepare the IP environment**

6. **Perform a simple setup**

   To give yourself a working setup that can be refined afterwards, follow the instructions in Chapter 3 and, for B100x and B200x server blades, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

7. **Power on the blades in the chassis and, if required (and if diagnostics are available), perform initial diagnostics on the blades.**

   To do this for the B100s blade, follow the instructions in Chapter 4. For B100x and B200x server blades, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

8. **If you require, set up the system chassis for use in an environment that separates the data and management networks.**

   For B100s Solaris blades, follow the instructions in Chapter 5. These instructions tell you how use IPMP to take advantage of the presence of two switches in the chassis (if you have two SSCs installed).

   For B100x and B200x blades, to take advantage of the presence of two switches in the chassis, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

9. **If you require, set up the system chassis so that each blade has both a redundant connection to the data network (as described in Chapter 5) and a redundant connection to the management network.**

   For B100s blades, follow the instructions in Chapter 6.

   For B100x and B200x server blades running Linux or Solaris x86, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

10. **If you require, set up the system chassis so that different server blades are assigned to different owners, each of whom can manage his or her own blades without being able to access the System Controller, switches, or any other owners' blades.**

    To do this, follow the instructions in Chapter 7.

1.2 The Sun Fire B1600 Blade System Chassis

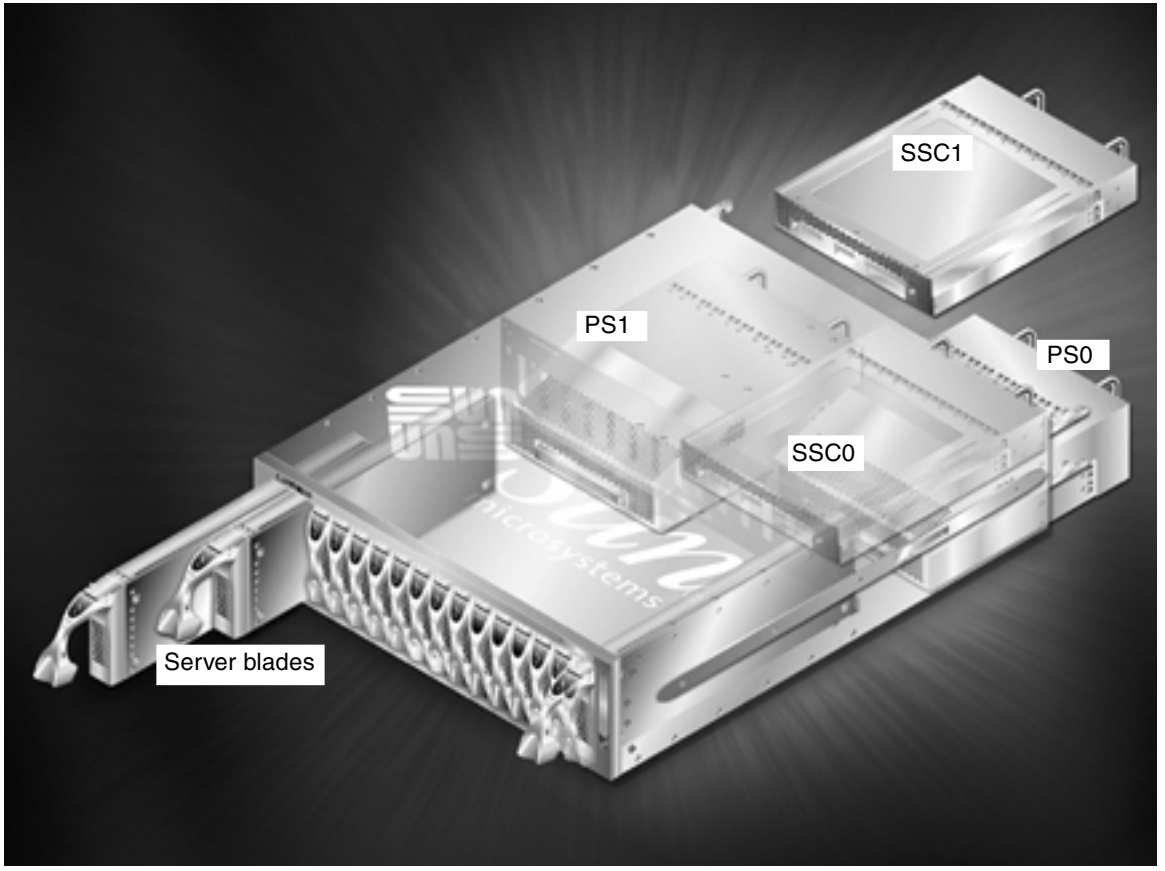

FIGURE 1-1 The Sun Fire B1600 Blade System Chassis

The Sun Fire B1600 blade system chassis is a 3-U high, 16-server chassis designed primarily for use by internet service providers. It is also suitable for use within corporate customer networks, wherever there is a need to maximize the density of high-performance servers.

As well as having the capacity for up to 16 server blades, the chassis contains two Power Supply Units (PSUs) and either one or two Switch and System Controller (SSC) units.

1.3 This Manual

This manual, plus the *Sun Fire B1600 System Chassis Software Setup Quick Start* poster, and the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide* (supplied as hard copy with chassis containing B100x and B200x blades) are designed to enable you - in the first instance - to set up a blade system chassis without needing to refer to the online manuals on the documentation and drivers CD supplied in the ship kit.

However, as well as online versions of all the hardcopy documents you received in the chassis ship kit, this CD includes:

- an online (PDF) guide to the Advanced Lights Out Management Software
  This is called the *Sun Fire B1600 Blade System Chassis Administration Guide*
- an online (PDF) command-reference manual for the integrated switch
  This is called the *Sun Fire B1600 Blade System Chassis Switch Administration Guide*

1.4 Software for the Blade System Chassis

The three main software components of the blade system chassis are the software for:

- the one or two System Controllers (SSC0 and SSC1)
- the one or two switches (one in SSC0 and one in SSC1)
- the server blades

1.4.1 Active and Standby System Controllers

As you can see from FIGURE 1-1, a chassis can contain up to two SSC units. If you have two SSCs installed, then they conduct themselves in an active-standby relationhsip.

In the factory configuration of a chassis with dual SSCs, the System Controller in SSC0 is the "active" one, and the System Controller in SSC1 is the "standby" one.

If the SSC containing the active System Controller is physically removed or if its main software application undergoes a major failure, the standby System Controller (in the remaining SSC) automatically becomes active.

It is also possible from the active System Controller's command line to cause the standby System Controller to become the active one. For information about how to do this, refer to the *Sun Fire B1600 Blade System Chassis Administration Guide*.

In a future release of the System Controller software, the standby System Controller will automatically take over from the active one if it judges (as a result of its constant monitoring of the standby System Controller) that it is less fit than the other to perform the active role.

The System Controllers share an alias IP address and they can also each have a private IP address. The alias IP address is the address of the active System Controller, whichever that System Controller is. This address needs to be specified in the Name Service. When a System Controller takes the role of active System Controller, it assigns to itself the alias address and advertises itself (in a broadcast containing both its MAC address and the alias IP address) to the wider network as the device that owns the alias IP address.

For information about the relationship between the active and standby System Controllers, see Section 1.5.1, "The Role of the System Controllers" on page 1-9, and Appendix E.

If you assign a private IP address to each System Controller, you can use the private IP address (as an alternative to the alias IP address) if you want to telnet into the active System Controller.

You cannot telnet into the standby System Controller, even if it has a private IP address. However, it is useful to assign private IP addresses to the System Controllers (see Chapter 3), because it enables you (as network administrator) to perform a quick check of their presence on the network by pinging them individually.

1.4.2 Dual Redundant Switches

The chassis can be run with one or two Switch and System Controller units installed. Although only one System Controller can be active at a time, the switches inside both SSCs are active all the time. This is an important feature of the blade system chassis's design. Each server blade has either two Gigabit network interfaces (one for each switch) or four Gigabit network interfaces (two for each switch). Therefore, in the event of a failure of network connectivity on one interface of a two-port blade, the other interface continues to provide service. Similarly, if for example a switch failed and a blade with four interfaces were in use in the chassis, the two interfaces connected to the second switch would continue to provide service.

For information about how to take advantage of:

- the dual connections from each B100s blade to your wider network, see Chapter 3 and Chapter 5.
- the dual connections from each B100x blade or the quadruple connections from each B200x blade to your wider network, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

Note – Note that in a chassis containing two SSCs, both switches are active all the time, even though only one System Controller is active at any one time.

1.4.3 Sun Fire B100s Server Blades (SPARC Solaris)

The B100s server blades are logically equivalent to standard Sun entry-level SPARC Solaris servers. All standard methods for network and `sysid` configuration (for example, TFTP and DHCP) are available for them, and so are the following methods of network installation for the Solaris Operating Environment:

- Web Start Install
- Interactive Install
- Custom Jumpstart Install
- Web Start Flash Install

For information about these methods of installing Solaris, refer to Chapter 3 of the *Solaris 8 Advanced Installation Guide* (supplied with the Solaris 8 12/02 media kit).

1.4.4 Sun Fire B100x and B200x Server Blades (Linux and Solaris x86)

The B100x and B200x server blades are logically equivalent to standard Linux or Solaris x86 servers. They are designed to receive either operating system from a PXE boot environment and, once having done so, to boot from their own hard disk.

For information about providing a PXE boot environment and configuring the blade to boot from it, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

To use the B100x or B200x server blades, you need to use System Controller firmware version 1.2 (or later).

Download the latest version of the firmware from the following website:

`http://wwws.sun.com/software/download/network.html`

To perform the upgrade of the System Controller firmware, follow the instructions in the *Sun Fire B1600 Blade System Chassis Administration Guide*.

1.4.5 Content Load Balancing Blade

The Sun Fire B10n Content Load Balancing Blade is now available to provide load balancing across server blades in the Sun Fire B1600 Blade System Chassis and other horizontally scaled Sun platforms.

To use the B10n Content Load Balancing Blade, you need to use System Controller firmware version 1.1 (or later).

Download the latest version of the firmware from the following website:

`http://wwws.sun.com/software/download/network.html`

To perform the upgrade of the System Controller firmware, follow the instructions in the *Sun Fire B1600 Blade System Chassis Administration Guide*.

To configure and use the B10n Content Load Balancing Blade, refer to the *Sun Fire B10n Content Load Balancing Blade Administration Guide*.

1.4.6 SSL Proxy Blades

The Sun Fire B10p and B15p SSL proxy blades are now available. These are hardware acceleration network systems that offload the SSL handshake and encryption/decryption functions from servers. The SSL proxy blade receives SSL

encrypted traffic (typically HTTPS) from the client, decrypts the data, and sends it to the server (or to a virtual server port of a load balancer or switch). The server responses are sent to the SSL proxy blade, where they are encrypted and returned to the client.

To use the B10p or B15p proxy blade, you need to be running System Controller firmware version 1.2 (or later).

Download the latest version of the System Controller firmware from the following website:

`http://wwws.sun.com/software/download/network.html`

To perform the upgrade of the System Controller firmware, follow the instructions in the *Sun Fire B1600 Blade System Chassis Administration Guide*.

To configure and use the B10n Content Load Balancing Blade, refer to the *Sun Fire B10p and B15p SSL Proxy Blade Administration Guide*.

1.5 The Roles of the System Controllers, Switches, and Server Blades

1.5.1 The Role of the System Controllers

The active System Controller does two things: it communicates with the sub-components of the system chassis to monitor their operational status; and it provides a command-line interface (available over a serial connection or via telnet) to the main chassis configuration software called the "Advanced Lights Out Management Software". This software is an application that runs on the active System Controller in the chassis.

Chapter 2 of this manual tells you how to log into the Advanced Lights Out Management Software on the System Controller.

When you are logged in, you have access to:

- The set of commands specific to the active System Controller for monitoring and managing the system chassis and its components. For more information about these commands, see Appendix D, and refer to the *Sun Fire B1600 Blade System Chassis Administration Guide*.

- The consoles on the two integrated switches in the system chassis. For information about the commands specific to the switch's command-line interface, see Appendix A and refer to the *Sun Fire B1600 Blade System Chassis Switch Administration Guide*.
- The consoles on the server blades that you have installed in the system chassis.

For a detailed discussion of the relationship between the active and standby System Controllers, and of the limitations of this relationship, see Appendix E.

1.5.2 The Role of the Switch

FIGURE 1-2 shows all the Ethernet Ports on each switch plus the Ethernet interfaces on each server blade. Each server blade has one interface to the switch in SSC0 and one interface to the switch in SSC1. The individual switches have a single port for each server blade. These are labeled SNP0 through SNP15. The data network uplink ports are labeled NETP0 through NETP7.

Note – There is no direct relationship between particular data network uplink ports and particular server blade ports. Instead a high-speed midplane between these two groups of ports switches all traffic between them. This is indicated in the diagram by the heavy black line from the SNP ports and the NETP ports to the switch fabric.

1.5.2.1 The NETMGT Port

The diagram also shows the external RJ-45 port that is labeled NETMGT on the back panel of the SSC. This port provides an Ethernet connection, via a mini-switch (see FIGURE 1-2), to both the System Controller (indicated in the diagram by SC) and the switch. The external label NETMGT therefore describes both the switch and the System Controller management ports.

However, the functionality of the switch's NETMGT port is largely internal to the switch component of the SSC; and furthermore this functionality is distributed among several hardware and software entities inside the switch, including:

- an AN 983 NIC (network interface card) with associated driver software running on the switch controller (which is an MPC 8245 CPU chip);
- a PCI port on a BCM 5632 switch chip, with associated driver and also port emulation software running on the switch controller, providing a pseudo-port within the switching fabric;

- and packet filter software (also running on the switch controller) to control traffic between the pseudo-port and the NIC. The packet filter prevents any traffic from the server blades passing through the NETMGT port unless it been permitted by user-specified filter rules. (By default no traffic is permitted from the blades through the NETMGT port.) However, traffic can pass from the external NETMGT to the server blades. The packet filter does not block traffic in this direction. (The packet filter is described further in the next sub-section.)

This distributed NETMGT port functionality provides user access to the switch's management applications. These applications (which provide a command-line interface, a web GUI, and an SNMP interface) also run on the switch controller. They enable you to configure the switch parameters. They also control the working of the switch.

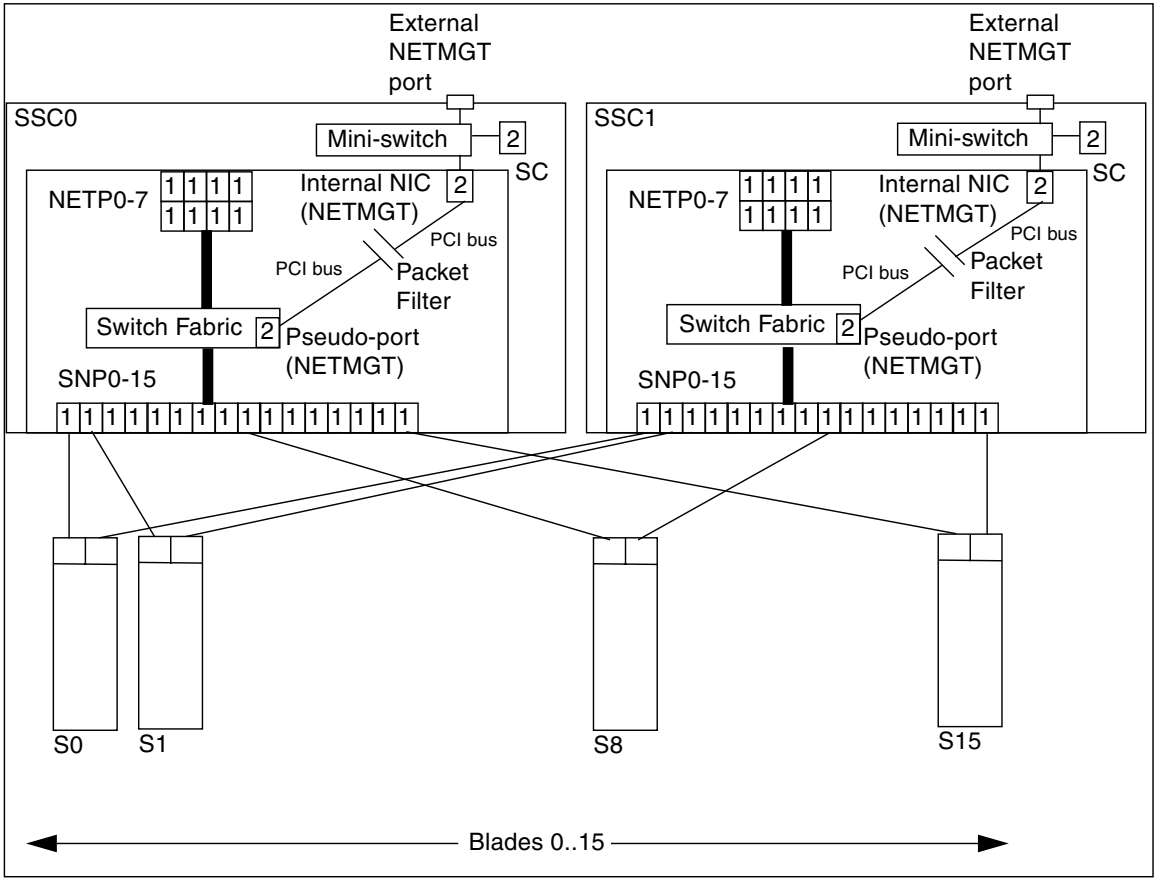

FIGURE 1-2 The Ethernet Ports and Interfaces on the System Chassis and Their Default VLAN Numbers

The switch controller communicates with the switching fabric via an internal PCI bus, and it communicates with the outside world (in other words, with your management network) via a 100BaseT ethernet card (this is the AN 983 NIC). The 100BaseT ethernet card is connected to an internal five-port mini-switch (a Marvell 6051). Only three of the five ports on the mini-switch are used (see FIGURE 1-2):

- one links the external RJ-45 port to the mini-switch;
- one links the switch's network interface card to the mini-switch;
- and one links the System Controller's network interface card to the mini-switch.

This three-way connectivity provided by the mini-switch, therefore, enables users to access both the System Controller and the switch from the management network.

The internal mini-switch auto-negotiates the speed and duplex mode for its three ports at boot time. You cannot view information about the state of these ports from any of the switch's management application interfaces, and you cannot configure the ports manually. However, it is possible to connect a 10BaseT link to the RJ45 port (the external NETMGT port), although, if you do this, the management applications will still configure (and describe) the connection to the management network as a 100BaseT full-duplex link.

The switch controller is effectively acting as a bridge between the managment network (attached to the RJ-45 external NETMGT port) and the main switching fabric. However, for security reasons, the pseudo-port within the switching fabric will not bridge traffic to or from any of the uplink data ports.

1.5.2.2 VLANs

The numbers 1 and 2 inside the Ethernet ports and SC interface in FIGURE 1-2 indicate the factory default VLAN[1] configuration for the switch ports. The default VLAN for the data network is VLAN 1. The default VLAN for the management network is VLAN 2.

The SC's VLAN Id is not configurable from the switch; it is configured as part of the interactive process of setting up the System Controller using the `setupsc` command (see Chapter 3). When you run this command you are asked a series of questions, including whether you want to enable a VLAN for the SC. If you answer yes, you are then prompted to specify a VLAN Id for the SC interface; the default is VLAN 2 in accordance with the default management VLAN on the switch. The SC interface is not a switch port. The effect of enabling VLAN support on this interface is to cause it to accept and transmit only frames tagged for the VLAN you specify for it.

1. A VLAN is a Virtual Local Area Network; that is, a self-contained logical network and broadcast domain defined by the software configuration of a set of ports on one or more network infrastructure devices.

As stated above, for security reasons, the pseudo-port within the switching fabric will not bridge traffic to or from any of the uplink data ports. This switching policy is enforced by the fact that it is not possible for any uplink data port to be a member of the same VLAN as the NETMGT port. The switch controller will not permit you to add the NETMGT port to any VLAN containing an uplink data port, or to add any uplink data port to a VLAN containing the NETMGT port.

1.5.2.3 The Packet Filter

The packet filter in FIGURE 1-2 is in the first instance a barrier between the internal NETMGT port and all of the server blade ports. It protects your management network from attack by external users accessing the blades through the data network.

By default, no network traffic is allowed to pass between the server blades and the NETMGT port on the switch. However, you can permit certain traffic to pass through the packet filter by specifying rules concerning particular protocols. For information about how to do this, see Appendix A.

1.5.3 The Role of the Server Blades

The server blades provide the computing power to run software applications. Their primary means of I/O (input/output) is the network, although you can console into them from the System Controller's command-line interface by means of an internal serial connection between the System Controller and each blade.

The single processor blades have one Gigabit Ethernet interface to each of the chassis's two internal switches; the dual-processor blades have two Gigabit Ethernet interfaces to each internal switch. The switches also provide Gigabit Ethernet interfaces to the external network.

Blades typically have local disk storage to hold Operating System software and configuration information. It is not expected that customers will store user data on the blades' local storage devices but will use remote storage facilities instead.

In their factory default state, the B100s (SPARC) Solaris blades boot using an Operating System stub stored on the local hard disk. When this has booted, the blade searches for a Network Install Server on your network from which to install the Operating System. When you have booted the first B100s server blade from the Network Install Server, you can add the application software you intend to run on the blade, then follow the instructions in the *Solaris Advanced Installation Guide* to create a Web Start Flash Archive. Using Web Start Flash Archives on the Sun Fire B100s Solaris server blades (in the Sun Fire B1600 Blade System Chassis) enables you

to replicate one blade's operating system and application software on other blades. It therefore speeds up the process of configuring a whole chassis of server blades. For more information about using Web Start Flash Archives, see Appendix C.

In their factory default state, the B100x and B200x (Linux and Solaris x86) blades need to be configured to boot temporarily from the network (see Chapter 4 of this manual) so that they can install their operating system from a PXE boot environment. For instructions about installing Linux or Solaris x86 from a PXE boot environment, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

1.6 Before You Configure the Software

To perform the initial configuration when you have installed and applied power to the blade system chassis, you must either set up a serial connection to SSC0 (by default, the active System Controller) or you must set up a DHCP server to perform the IP configuration of the chassis's active System Controller automatically. If you set up a DHCP server to do this, you can then telnet into the active System Controller to set up the chassis for the first time.

For information about the cabling for setting up a serial connection, refer to the *Sun Fire B1600 Blade System Chassis Hardware Installation Guide*.

For information about using a DHCP server, see Section 1.8, "Using a DHCP Server to Provide the SSC IP Addresses Automatically" on page 1-16.

Note – If you have two SSCs installed (if both SSCs are powered and working normally and neither is damaged), then by default SSC0 contains the active System Controller and SSC1 contains the standby System Controller. This means that, to set up the chassis for the first time using a serial connection, you need a serial connection to (at least) SSC0.

However, for day-to-day operation of the blade system chassis, we recommend you set up serial connections to both SSCs. This ensures that, if the active SSC fails for any reason, you do not lose serial connectivity to the chassis.

Chapter 2 and Chapter 3 of this manual tell you how to configure the chassis when you have set up a serial or telnet connection to SSC0 (assuming that SSC0 contains the active System Controller).

1.7 IP Information Required for the Chassis

To enable your network environment to receive a Sun Fire B1600 blade system chassis, you need to configure it to provide an alias IP address for the active System Controller, plus an IP address, netmask, and default gateway for each of the Ethernet interfaces on the chassis.

The alias IP address for the System Controller is specified in the Name Service, but the System Controllers can also have a private IP address each (you do not need to specify this in the Name Service). When a System Controller takes the role of active System Controller, it assigns to itself the alias IP address and advertises itself (in a broadcast containing both its MAC address and the alias IP address) to the wider network as the device that owns the alias IP address.

A chassis that is fully populated with server blades and two SSCs uses at least 37 IP addresses (including the two private SC addresses):

1. An alias IP address for the active System Controller (this is the address used by the active System Controller, whether it is the System Controller in SSC0 or SSC1).
2. A private IP address for the System Controller in SSC0.
3. A private IP address for the System Controller in SSC1.
4. An IP address for the switch in SSC0.
5. An IP address for the switch in SSC1.
6. 16 IP addresses for the `ce0` or `bge0` primary (Gigabit Ethernet) interfaces on B100s or B100x blades (assuming 16 blades in the chassis), or eight IP addresses for the primary blade interfaces in a chassis full of B200x blades.
7. 16 IP addresses for the `ce1` secondary (Gigabit Ethernet) interface on B100s blades (assuming 16 blades in the chassis), none for the `bge1` secondary interface on B100x blades running Linux (the Linux “master driver” software that handles all interface management requires only a primary IP address), or eight IP addresses (assuming eight B200x blades in the chassis) for the second, third, and fourth interfaces on B200x blades running Solaris x86.

If you intend to perform any of the chassis configurations (using B100s blades) described in Chapter 5, Chapter 6, or Chapter 7, each of which involves the use of the Internet Multipathing (IPMP) facility, you will require more than 64 IP addresses for the blades in a fully populated chassis.

For information about configuring redundant network connections on Linux and Solaris x86 blades, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

1.8 Using a DHCP Server to Provide the SSC IP Addresses Automatically

By default, the System Controller inside the active SSC will attempt to discover from a DHCP server the IP configuration information both for itself and for the standby System Controller.

The switches inside each SSC will also by default attempt to discover their IP configuration information from a DHCP server.

The System Controllers use a maximum of three IP addresses:

- One alias IP address (this is the address used by the active System Controller, whether it is the System Controller in SSC0 or SSC1)
- One (optional) private IP address for the System Controller in SSC0
- One (optional) private IP address for the System Controller in SSC1

Each switch requires one IP address.

Note – If you intend to enforce the separation of your data and management networks, the DHCP server you use to configure the SSCs needs to be on the management network, and the DHCP server you use to configure the server blades needs to be on the data network. For information about configuring a DHCP server to provide the IP addresses for the server blades, see Appendix B.

1.8.1 Configuring the SSCs with "Consistent" IP Addresses

The active System Controller sends a DHCP request for three IP addresses (SSC0, SSC1, and an alias IP address).

Each switch sends a DHCP request for one IP address.

If you want to use five "consistent" IP addresses (that is, five IP addresses that will not change), you must associate five specific addresses on the DHCP server with the client identifiers for the System Controllers and switches.

Client identifiers exist for the System Controllers individually (as well as for the active System Controller, whichever that one is) because the System Controllers can each have an optional private IP address. (It is useful to allocate private IP addresses, because it enables network administrators to ping them to check their presence on the network.)

The client identifiers for the System Controllers and switches in the chassis are listed in TABLE 1-1.

TABLE 1-1 Client Identifiers for the System Controllers and Integrated Switches

Device	Client Identifier
Active System Controller	SUNW,SHELF_ID=*serial number of chassis*
System Controller in SSC0 (private IP)	SUNW,SC_ID=*serial number of chassis*,0
System Controller in SSC1 (private IP)	SUNW,SC_ID=*serial number of chassis*,1
Switch in SSC0	SUNW,SWITCH_ID=*serial number of chassis*,0
Switch in SSC1	SUNW,SWITCH_ID=*serial number of chassis*,1

Note – The serial number of the chassis is printed on a label on the rear of the chassis (on the righthand side). For the client identifiers, you need to use only the last 6 digits of the number printed on the chassis label.

(An alternative way to find the chassis serial number is to run the `showfru ch` command on the System Controller's command line and inspect the field for `/ManR/Sun_serial_No`.)

Using the instructions in the *Solaris DHCP Administration Guide* for creating "consistent" IP addresses, configure a DHCP server on the same network as the SSCs to provide a block of five IP addresses mapped to the client identifiers listed above. Make a note of the IP address you map to each client identifier. You will need to know this in order to telnet into the active System Controller or into either of the switches. You will also need to know it if you want to access the web-based Graphical User Interface to the switches.

1.8.2 Configuring the SSCs with Dynamic IP Addresses

If you do not want to provide "consistent" IP addresses for the System Controllers and switches in the chassis, you can configure the DHCP server to provide a block of dynamic IP addresses. These will be bound to the client identifier after the devices have made their DHCP requests. For instructions on how to do this, refer to the *Solaris DHCP Administration Guide* (806-5529).

If you configure the DHCP server to provide a block of dynamic IP addresses, you will have to find out which IP address it has allocated to the System Controller and to the two switches, before you can telnet into the System Controller or either of the switches and before you can access the web-based Graphical User Interface to the switches (see Section 1.8.3, "Finding out the Chassis's IP Addresses to Enable You to Use Telnet" on page 1-18).

1.8.3 Finding out the Chassis's IP Addresses to Enable You to Use Telnet

If you want to log into the active System Controller for the first time over telnet (instead of by using a serial connection), and you have allocated dynamic (instead of "consistent") IP addresses to the chassis components, you will need to find out the IP address that your DHCP server has allocated to the System Controller.

If the DHCP server you are using is a Solaris system, you can use the `pntadm` command to list all the devices, with their corresponding IP addresses, on the network containing the chassis.

To do this, type:

```
# pntadm -P network address
pntadm -P 129.156.203.0
Client ID                                    Flags   Client IP         Server
IP       Lease Expiration

53554E572C5348454C465F49443D313233343536[1]        00      129.156.203.240
129.156.202.163 01/03/2003
53554E572C5357495443485F49443D3132333435362C30[2]  00      129.156.203.241
129.156.202.163 01/03/2003
53554E572C5357495443485F49443D3132333435362C31[3]  00      129.156.203.242
129.156.202.163 01/03/2003
```

Key to sample output:
1. Client ID of active System Controller
2. Client ID of switch in SSC0
3. Client ID of switch in SSC1

where *network address* is the network address of your management network. The devices in the list are each identified by a hexadecimal string representing their client identifiers.

In the example, the first device listed is the active System Controller (which is the one using the alias IP address), the second is the switch in SSC0, and the third is the switch in SSC1. You need to translate the hexadecimal strings in your output into their alpha-numeric equivalents in order to see which device has been allocated which client IP address.

(Note that in the sample output two columns of information to the right of the "Lease Expiration" column have been omitted for lack of space. These are the "Macro" and "Comments" columns.)

TABLE 1-2 Sample Client ID Translation for an Active System Controller

	Active System Controller	Serial Number of Chassis
Hex	53554E572C5348454C465F49443D	313233343536
Alpha-numeric	SUNW,SHELF_ID=	123456

TABLE 1-3 Sample Client ID Translation for (Optional Private IP Address) of SC[1] in SSC0

	Active System Controller	Serial Number of Chassis	Suffix for SSC0
Hex	53554E572C5353435F49443D	313233343536	2C30
Alpha-numeric	SUNW,SC_ID=	123456	,0

1. System Controller

TABLE 1-4 Sample Client ID Translation for a Switch in SSC1

	Switch in SSC1	Serial Number of Chassis	Suffix for SSC1
Hex	53554E572C5357495443485F49443D	313233343536	2C31
Alpha-numeric	SUNW,SWITCH_ID=	123456	,1

1.8.4 Accessing the System Controller Using Telnet

To telnet into the active System Controller when you have configured a DHCP server to provide the IP addresses required:

1. **If your chassis is already powered, you need to power cycle the chassis by removing the IEC power cables.**

2. **When the chassis is powered, type the following at a remote terminal:**

```
% telnet alias ip address or host name
Trying alias ip address
Connected to alias ip address or host name
Escape character is '^]'

Sun Advanced Lights Out Manager for Blade Servers 1.2
Copyright 2003 Sun Microsystems, Inc. All Rights Reserved.
ALOM-B 1.2

username:
```

where *alias ip address* is the IP address of the active System Controller. (Alternatively you can specify a host name on the command line.)

1.9 Returning to the sc> Prompt From a Switch or Blade Console

Before proceeding to configure the chassis, you will find it useful to have a note of the escape sequence for returning to the System Controller's sc> prompt from a blade or switch console. The sequence is #. (that is, the hash '#' character, followed by the dot '.' character).

When you are following the instructions in this manual, it is from the sc> prompt that you will need to access the server blade and switch consoles.

What to do Next

Proceed to Chapter 2 to perform the preliminary setup tasks for the system chassis.

CHAPTER 2

Setting the Passwords, Date, and Time on the SSCs

This chapter tells you how to log into the active System Controller and both switches (if you have two SSCs installed) to perform the preliminary tasks necessary before you can configure the blade system chassis for use in your network environment.

You need to configure the active System Controller but not the standby one. The active one propagates the information you configure it with to the standby System Controller (if present) so that the standby one can take over if it ever becomes necessary.

The user login and password information for the switches is separate from the user login and password information for the System Controllers. It therefore has to be configured separately.

The chapter contains the following sections:

Follow all the instructions in both sections.

Note – To configure the system chassis, you need to use the command-line interface to the active System Controller. However, from this interface, you will need to access the consoles to the two switches and the consoles to the server blades. When you are at a switch or blade console, type **#.** to return to the System Controller's `sc>` prompt.

2.1 Logging into the System Controller, Setting a Password, and Setting the Time

This section tells you how to log into the active System Controller as user `admin` (the default user) and how to specify a password for that user.

Note – The user login and password information you configure on the System Controllers is entirely separate from the user login and password information you configure on the switches. For information about configuring this information on the switches, see Section 2.2, "Logging into the Switch as the Default User and Setting the Passwords" on page 2-4.

This section assumes you have set up a serial or telnet connection to the active System Controller. (You cannot set up a telnet connection to the standby System Controller.) If you have connected by telnet using the alias IP address, then you will be connected to whichever System Controller is currently the active one.

If you are using a serial connection, you need to know that, in the chassis's factory default configuration, the active System Controller is the one in SSC0. If you connect to SSC1 (and SSC1 contains the standby System Controller), you will see a message telling you that you are connected to the standby System Controller. In this case, set up a connection to SSC0. In any case, we recommend you maintain serial connections to both SSCs.

To begin setting up the blade system chassis, do the following:

1. **At the `username:` prompt, type the default user name (`admin`).**

```
Sun Advanced Lights Out Manager for Blade Servers 1.0
ALOM-B 1.0

username: admin
```

2. **At the `sc>` prompt, set a password for the default user.**

   The default user (`admin`) is pre-configured and cannot be deleted. This user initially has permission only to set its own password. When the password has been set, it acquires full user permissions. To enable yourself to proceed with the configuration of the blade system chassis, you must set a password for the default user (`admin`).

   The first password you specify must:

   - Begin with an uppercase or lowercase alphabetic character and contain at least two uppercase or lowercase alphabetic characters,

- Contain at least six characters (although it can contain up to eight),
- Contain at least one numeric character, or period (.), underscore (_), or hyphen (-).
- Not be identical with the default user login name (`admin`), or with the default user login name in reverse (`nimda`), or with any re-arrangement of the characters of this name that would retain their sequence in a continuous circular reading of the name (for example, `dmina`, `minad`, `inadm`, and `nadmi` are all forbidden).

For more information about setting up named users for the System Controller, refer to the *Sun Fire B1600 Blade System Chassis Administration Guide*.

To set a password for user `admin`, type:

```
sc> password
Enter new password:
Enter new password again:
New password set for user admin successfully
sc>
```

3. Set the date and time on the active System Controller.

Note – When you set the date and time, you must use Co-ordinated Universal Time (UTC). The server blades work out the local time for your time-zone by using an offset from Co-ordinated Universal Time on the System Controller. They receive the time from the System Controller.

The command for setting both the date and time is the same: it is the `setdate` command. The syntax for this command is as follows:

```
sc> setdate [mmdd]HHMM[.SS] | mmddHHMM[cc]yy[.SS]
```

where *mm* is the month (two digits), *dd* is the day (two digits), *HH* is the hour (two digits), *MM* is the minutes (two digits), *SS* is seconds (two digits), *cc* is the century (20), *yy* is the year (two digits).

- **To set the time (24-hour clock)**

Type the hour (two digits), followed by the minutes (two digits). For example, to set the time to 11:42, type the following:

```
sc> setdate 1142
```

- **To set the month, day, and time (24-hours to the nearest minute)**

  Type the (two-digit) number of the month in the year, followed by the (two-digit) number of the day in the month, followed by the hour (two digits), followed by the minutes (two digits). For example, to set the date and time to 11.42 am on March 27, type the following:

```
sc> setdate 03271142
```

- **To set the day, month, time (24-hour), year, and seconds**

  Type the (two-digit) number of the day in the month, followed by the (two-digit) number of the month in the year, followed by the hour (two digits), followed by the minutes (two digits), followed by the year (either four characters or two; for example, “2002” or “02”), followed, optionally, by a dot and the seconds (two digits). For example, to set the date and time to 47 seconds after 11.42 am on 27 March, 2002, type the following:

```
sc> setdate 2703114202.47
```

2.2 Logging into the Switch as the Default User and Setting the Passwords

This section tells you how to log into the switch and how to set and save its passwords.

Note – The user login and password information you configure on the switches is entirely separate from the user login and password information you configure on the System Controllers.

---

1. **Type:**

```
sc> console sscn/swt
```

where *n* is either 0 or 1 depending on whether you are configuring the switch in SSC0 or SSC1. For example, to configure the switch in SSC0, type:

```
sc> console ssc0/swt
```

2. **When prompted for a user name and password, type** `admin` **for both.**

```
Username admin
Password *****

CLI session with the host is opened.
To end the CLI session, enter [Exit].
```

3. **At the** `console#` **prompt, type:**

```
Console#configure
```

4. **Set at least the first one of the switch's following three passwords:**

   a. **Set a password to give yourself access to the switch's Privileged Exec command mode.**

   This is the command mode that enables you to view and change the whole of the switch's configuration. The default user `admin` (see Step 2) has Privileged Exec rights. For security, we recommend you change the password for this user. Type:

```
Console(config)#username admin password 0 password
```

   where *password* is a string of 1-8 characters in length. The 0 indicates to the switch that the password is specified in plain text. There is an alternative parameter (7) which indicates that the password is specified in encrypted text. But there are no practical situations in which you will need to use this parameter. For more information, see "Understanding Why the Switch Needs to be Told That a Password is Not Encrypted" on page A-19.

   b. **For information about the use of plain or encrypted text for passwords, see the** *Sun Fire B1600 Blade System Chassis Switch Administration Guide***.)**

c. Set a password for the user `guest`.

The user `guest` can view some switch configuration and status information and can also execute ping commands. This user cannot alter any of the switch's configuration settings. The default password for this user is `guest`. To set a new password for it, type:

```
Console(config)#username guest password 0 password
```

where *password* is a string of 1-8 characters in length. (The zero indicates that the password is specified in plain text.)

d. Set a password for the `enable` command.

The `enable` command enables a user logged in as `guest` to acquire Privileged Exec rights. If this user types `enable` on the command line, he or she will be prompted for a password. The default password for the `enable` command is `super`. To set a new password for it, type:

```
Console(config)#enable password level 15 0 password
```

where *password* is a string of 1-8 characters in length. The number 15 specifies that anyone who is authorized to run the `enable` command will have Privileged Exec rights. The zero indicates that the password is specified in plain text.

Note – For more information about the different command modes available on the integrated switch, refer to the *Sun Fire B1600 Blade System Chassis Switch Administration Guide*.

---

5. Leave the switch's configuration mode by typing:

```
Console(config)#end
```

or

```
Console(config)#exit
```

6. **Because you have changed the switch's configuration, you must now save the configuration.**

   The method of doing this is to copy the running configuration firmware to the startup configuration firmware.

   Type:

```
Console#copy running-config startup-config
Startup configuration file name []:filename
Write to FLASH Programming
-Write to FLASH finish
Success

Console#
```

   where *filename* is the name you want to give to the file that will contain your new startup configuration.

7. **If you are using DHCP to provide the switch's IP configuration, then we recommend that you configure the second switch now either by:**

   - Repeating Step 1 through Step 6 above on the second switch, or by
   - Following the instructions in Section A.10, "Copying the Configuration of the First Switch to the Second" on page A-10. When you copy the switch configuration, the login and password information you have configured will also be copied.

   If you are not using DHCP, you do not need to configure the second switch at this point. The instructions in Chapter 3 tell when to do this, but you need to do some more configuration of the first switch before you copy the configuration over.

What to Do Next

If you are installing:

- B100s blades, go to Chapter 3 to perform a simple network installation, then set up the server blades by following the instructions in Chapter 4. If you need to perform a more sophisticated network configuration, read Chapter 5, Chapter 6, and Chapter 7.
- B100x and/or B200x blades, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide* for information about installing the Linux or Solaris x86 operating system onto a blade.

CHAPTER 3

Installing a Chassis into a Simple Network

This chapter contains the following sections:

Note – For information about supporting blades running Linux and/or Solaris x86, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

3.1 Taking Advantage of Having Two Switches in the System Chassis

This chapter tells you how to set up a Sun Fire B1600 blade system chassis for use in a simple network where there is no separation of the data and management networks. The instructions enable you to take advantage of the presence of two switches inside the chassis to give the server blades two connections each to your network.

Note – For information about setting up your wider network to support B100x and B200x blades running Linux and/or Solaris x86, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

Section 3.2, "Preparing the Network Environment Using DHCP" on page 3-5 and Section 3.3, "Preparing the Network Environment With Static IP Addresses and Host Names" on page 3-6 assume that you have B100s (SPARC Solaris) blades installed in the chassis.

FIGURE 3-1 shows a sample network containing a Sun Fire B1600 blade system chassis. The subsequent sections use this diagram and the IP addresses marked on it to illustrate the steps you need to perform.

This chapter also includes a sample `/etc/hosts` file, and sample `/etc/ethers` and `/etc/netmasks` files. These illustrate how to edit the files on a Name Server to prepare your network environment to receive the chassis. Use these sample administration files for guidance, substituting your own IP addresses and host names for the ones used in the sample network illustrated in FIGURE 3-1.

Note – When you are considering how to integrate the system chassis into your network environment, remember that the chassis contains two switches and that each server blade has either one interface or two interfaces to each switch. Although only one of the chassis's System Controllers is active at any one time, both of its switches are active all the time. This means that, in a chassis that is working normally, *both* switches will provide the server blades with continuous network connectivity. However, if one switch fails, the other switch will continue to provide network connectivity.

This chapter tells you how to take advantage of this element of network redundancy by configuring different IP addresses for each Ethernet interface on the blades. Note also that, if the active System Controller fails, the switch inside the SSC whose System Controller has failed continues to provide network connectivity.

To take advantage of the redundancy offered by the second switch inside the system chassis, we recommend you to:

- Operate the system chassis always with two SSCs installed.
- Make sure that the cable connections from the eight data network uplink ports to the subnets on your wider network are exactly duplicated on the eight uplink ports of the second switch.
- Duplicate the configuration of the first switch on to the second switch. For information about how to do this, see Section A.10, "Copying the Configuration of the First Switch to the Second" on page A-10.
- Specify IP addresses for both Ethernet interfaces (`ce0` and `ce1`) on each B100s server blade if you are using a DHCP server to provide the chassis's IP network configuration.
- Specify IP addresses for both Ethernet interfaces (`ce0` and `ce1`) on each server blade if you set up an `/etc/hosts` file on your Name Server (see CODE EXAMPLE 3-2) to provide a static (non-DHCP) IP configuration for the chassis.
- Specify the MAC and IP addresses for both ethernet interfaces on each server blade when you set up an `/etc/ethers` file on your boot server to provide a static (non-DHCP) IP configuration for the chassis.
- To maximize the advantage of the redundant interfaces provided from each server blade to dual integrated switches in the chassis you need to use IPMP (on Solaris server blades (see Chapter 5) and bonding on Linux blades. However, the bonding feature is not available for the Linux blades at the time of printing.

3.1.1 Finding Out the MAC Addresses of Each Blade's Two Ethernet Interfaces

When you set up an `/etc/ethers` file on your boot server, you need to know the MAC address of `ce0` and `ce1` on each server blade. To find this out:

1. **Log into the active System Controller (see Chapter 2).**
2. **At the `sc>` prompt, type:**

```
sc> showplatform -v
```

3. **The output includes the MAC address for `ce0` on each server blade (labeled s0 through s15).**

   Calculate the MAC address for `ce1` as the next contiguous hexadecimal number after the number used for `ce0` on each blade. For example, if the MAC address for `ce0` were `00:03:ba:29:ef:32` then for `ce1` it would be `00:03:ba:29:ef:33`; or, if the MAC address for `ce0` were `00:03:ba:29:ef:4f` then for `ce1` it would be `00:03:ba:29:ef:50`.

3.2 Preparing the Network Environment Using DHCP

The server blades, the System Controllers and the switches in the system chassis can receive their IP addresses dynamically from a DHCP server.

For information about configuring the DHCP server to supply the IP addresses for the Switch and System Controllers in the chassis, see Chapter 1.

For information about configuring the DHCP server to supply IP addresses for the server blades, see Appendix B.

Note – If you use DHCP to configure the IP settings for Solaris server blades, you cannot use IPMP to provide network resiliency.

Make sure you configure the DHCP server to provide an IP address for each interface on each server blade. For information about setting up a DHCP server to provide IP configuration parameters dynamically, refer to the *Solaris DHCP Administration Guide* (806-5529). This document is available on the Sun documentation web site at:

`http://docs.sun.com`

To configure the Network Install Server to operate with dynamically assigned IP addresses, you will need to supplement the information in both the *Solaris Advanced Installation Guide* and the *Solaris DHCP Administration Guide* (806-5529) with the information in Appendix B.

3.3 Preparing the Network Environment With Static IP Addresses and Host Names

FIGURE 3-1 shows a Sun Fire B1600 blade system chassis with two SSCs installed and slots configured for 16 blades. The `ce0` interface for each blade in a system chassis has a connection to the switch in SSC0, and the `ce1` interface on each blade has a connection to the switch in SSC1. One or more of the eight uplink ports on the switch are connected to an external switch that has a Network Install Server (also containing a Name Server) connected to it. This external switch has a router (with IP address 192.168.1.1) connected to it that acts as the default gateway from the Sun Fire B1600 blade system chassis to the wider network. Finally, on both SSCs the 100Mbps network management port (labeled NETMGT on the rear of the chassis) is also connected to the external switch.

All of the IP addresses assigned to the system chassis are on the same subnet.

To prepare a simple network environment (like the one illustrated in FIGURE 3-1) to receive a system chassis containing B100s blades, you need to edit the `/etc/hosts`, `/etc/ethers`, and `/etc/netmasks` files on your Solaris Name Server:

- CODE EXAMPLE 3-1 is a sample `/etc/hosts` file containing IP addresses and hostnames for the network configuration illustrated in FIGURE 3-1.
- CODE EXAMPLE 3-2 is a sample `/etc/netmasks` file containing netmasks for the IP network numbers used in the sample network in FIGURE 3-1.

Note – If you are installing a chassis containing B100x and/or B200x blades, you need to boot the blades using a PXE boot environment to install the operating system. For instructions, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

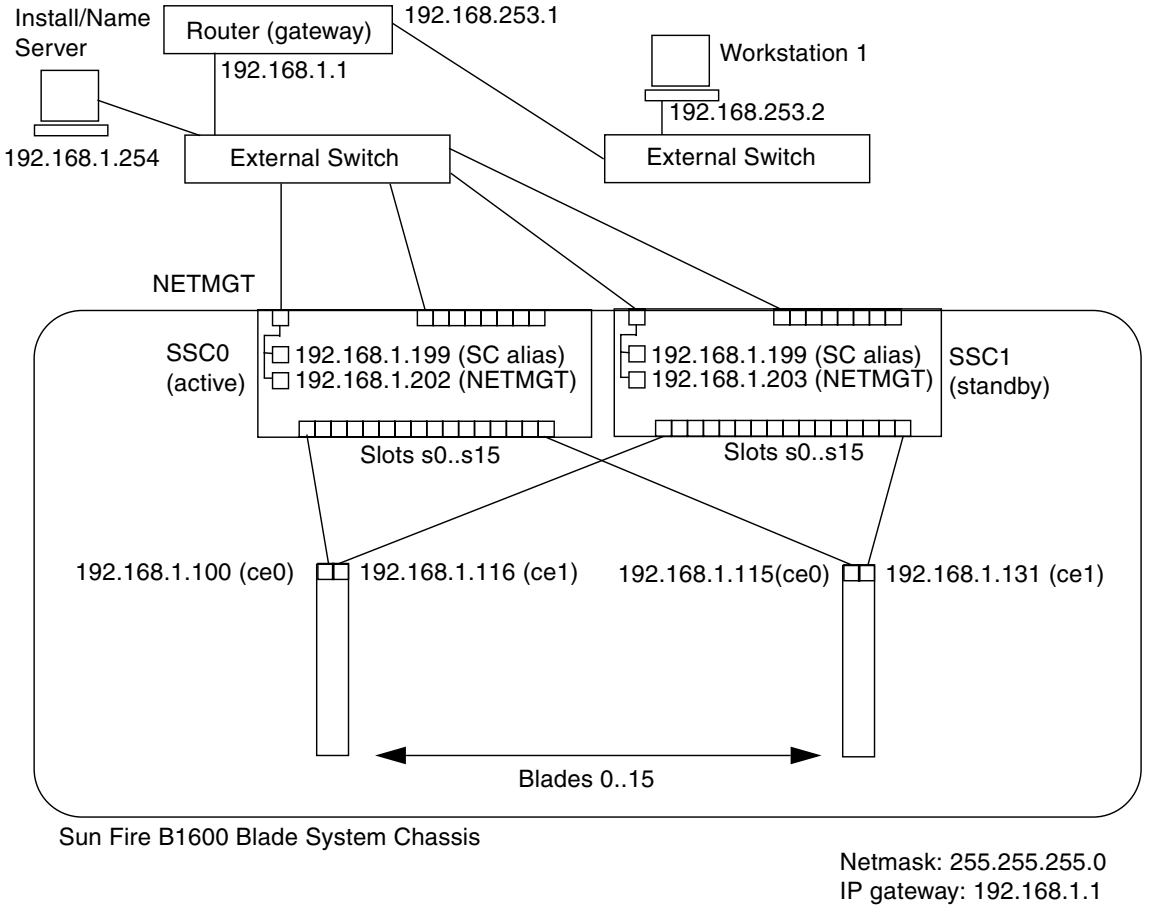

FIGURE 3-1 Sample Configuration Using No VLANs

CODE EXAMPLE 3-1 Sample `/etc/hosts` File on the Name Server

```
# Internet host table
127.0.0.1      localhost
192.168.1.254  datanet-nameserver   # Data network name server
192.168.1.1    datanet-router-1     # Data network router (default gateway)
192.168.253.1  datanet-router-253   # Data network router (client side)
192.168.253.2  dataclient-ws1       # Data client network workstation

192.168.1.199   medusa-sc           # Medusa - active SC (alias IP address)
192.168.1.200   medusa-sc0          # Medusa - SSC0/SC (private IP address)
192.168.1.201   medusa-sc1          # Medusa - SSC1/SC (private IP address)
192.168.1.202   medusa-swt0         # Medusa - SSC0/SWT
192.168.1.203   medusa-swt1         # Medusa - SSC1/SWT

192.168.1.100   medusa-s0-0
192.168.1.101   medusa-s1-0
192.168.1.102   medusa-s2-0
192.168.1.103   medusa-s3-0
192.168.1.104   medusa-s4-0
192.168.1.105   medusa-s5-0
192.168.1.106   medusa-s6-0
192.168.1.107   medusa-s7-0
192.168.1.108   medusa-s8-0
192.168.1.109   medusa-s9-0
192.168.1.110   medusa-s10-0
192.168.1.111   medusa-s11-0
192.168.1.112   medusa-s12-0
192.168.1.113   medusa-s13-0
192.168.1.114   medusa-s14-0
192.168.1.115   medusa-s15-0
192.168.1.116   medusa-s0-1
192.168.1.117   medusa-s1-1
192.168.1.118   medusa-s2-1
192.168.1.119   medusa-s3-1
192.168.1.120   medusa-s4-1
192.168.1.121   medusa-s5-1
192.168.1.122   medusa-s6-1
192.168.1.123   medusa-s7-1
192.168.1.124   medusa-s8-1
192.168.1.125   medusa-s9-1
192.168.1.126   medusa-s10-1
192.168.1.127   medusa-s11-1
192.168.1.128   medusa-s12-1
192.168.1.130   medusa-s14-1
192.168.1.131   medusa-s15-1
```

CODE EXAMPLE 3-2 Sample `/etc/netmasks` File on the Name Server

```
# The netmasks file associates Internet Protocol (IP) address
# masks with IP network numbers.
#
#       network-number  netmask
#
# The term network-number refers to a number obtained from the
# Internet Network Information Center. Currently this number is
# restricted to being a class A, B, or C network number. In the
# future we intend to support arbitrary network numbers
# as described in the Classless Internet Domain Routing
# guidelines.
#
# Both the network-number and the netmasks are specified in
# "decimal dot" notation, e.g:
#
#               128.32.0.0 255.255.255.0
#
192.168.1.0     255.255.255.0
192.168.253.0   255.255.255.0
#
```

3.4 Configuring the System Controllers and Switches

To follow the instructions in this section you need a serial (or telnet) connection to the active System Controller (by default, the System Controller in SSC0).

For information about logging into the System Controller, see Chapter 1 and Chapter 2 above.

For information about setting up serial connections to the System Controllers, see the *Sun Fire B1600 Blade System Chassis Hardware Installation Guide*.

For information about setting up a telnet connection to the active System Controller, see Chapter 1.

3.4.1 Setting up the System Controllers

Note – You can only access the command-line interface on the active System Controller. However, the `setupsc` command described in this section configures both System Controllers. Note that, although only one System Controller is active at any one time, both switches are always active.

---

1. **Log into the active System Controller by following the instructions in Chapter 2.**
2. **Run the `setupsc` command.**

   At the `sc>` prompt, type:

```
sc> setupsc
Entering Interactive setup mode.
Use Ctrl-z to exit & save. Use Ctrl-c to abort.

Do you want to configure the enabled interfaces [y]?
Should the SC network interface be enabled [y]?
Should the SC telnet interface be enabled for new connections[y]?
Do you want to configure the network interface [y]?
```

   In response to the questions you are asked when you run `setupsc`, press [ENTER] to accept the default response (indicated in square brackets at the end of the question: `y` for yes, `n` for no).

   Accept the default answer of `y` for the first four questions.

3. **When asked if the System Controller (SC) should use DHCP to obtain its network configuration, answer yes or no.**

   If you answered yes, go to Step 5.

   If you answered no, then, when prompted, specify in turn the:

   - SC IP address (this is the IP address that the active System Controller, whether it is currently the SC in SSC0 or SSC1, will use to communicate with the wider network),
   - IP netmask for the System Controller,
   - default gateway for the System Controller.

4. **When asked if you want to configure the SC private IP addresses, answer yes or no.**

   Both the active and standby System Controller can have a private IP address. These private IP addresses must be different from each other and they must be different from the SC IP address (specified in Step 3).

   It is useful to specify addresses for them, because you can then ping these addresses as a method of checking the health of both System Controllers. You can also telnet into the active System Controller using its private IP address (as well as by using the advertised network address of the active System Controller). You cannot telnet into the standby System Controller even if it does have a private IP address.

5. **When asked if you want to enable a VLAN for the SC, answer yes or no.**

   If you answer yes, the System Controller's Ethernet port will accept and transmit only frames tagged for the VLAN you specify in answer to the next question.

   a. **When prompted, specify the VLAN Id (a number between 1 and 4094) for the management VLAN.**

      Specify the same number as you intend to use for the management VLAN on the switch. The default number for the management VLAN on the switch is 2. We do not recommend you use VLAN 1, because this is the default VLAN for the data network.

6. **When prompted, type the IP address of a DNS server on the same IP subnet as your System Controller(s) and switch(es).**

   You only need to do this if you want the command line prompt on the System Controller in the chassis to contain its hostname. This is useful, because if you have a number of B1600 chassis installed on your site, it enables you to tell at a glance which System Controller you are logged into.

7. **When prompted, type the Management network ping address.**

   This needs to be an IP address on the same IP subnet as the System Controller (in other words, on the management network). The active System Controller will ping this address periodically to monitor its own network connectivity. If one of these pings fails, the System Controller will fail over to the standby System Controller (assuming that a second SSC is installed). If no second SSC is installed, the active System Controller reports a critical fault but continues to provide service if possible.

8. **When prompted, specify the IP address of a System Management System (SMS).**

   Either press [ENTER] to proceed to the next question or type the address of a network management station you are using to run the Sun Management Center Software for the Sun Fire B1600 or the Sun SNMP Management Agent for the Sun Fire B1600.

9. **When asked if you want to configure the managed system interface, answer yes or no.**

   If you answer yes, you will be asked the following four questions about the System Controller's behavior with respect to blades whose operating system has hung or failed to load, or whose OBP (OpenBoot PROM) or BIOS firmware has failed to load.

   a. **When asked whether you want all server blades to be restarted automatically if hung, answer yes or no.**

      If you answer yes, the System Controller will attempt three times to restart the blade.

      If you answer no, you will be asked whether you want none of the blades to be restarted automatically. And if you answer no to this question, you will be asked about each blade individually.

   b. **When asked whether you want all server blades to restart if their firmware (F/W) hangs on startup, answer yes or no.**

      The firmware referred to is the OpenBoot PROM (OBP) firmware on a Solaris blade and the BIOS firmware on a Linux blade.

      If you answer no, you will be asked whether you want none of the blades to be restarted automatically. And if you answer no to this question, you will be asked about each blade individually.

   c. **When asked whether you want all server blades to be configured with the same Boot Timeout, answer yes or no.**

      If you answer yes, you are prompted to specify the timeout in seconds. The boot time is the time the operating system takes to boot. It is measured from when the OBP or BIOS firmware has finished loading to when the operating system is loaded and running. Solaris and Linux both take approximately 90 seconds to load onto a server blade. If you specify a Boot Timeout (in other words, a period after which the boot process is defined as having failed) make sure you allow enough time for normal booting to complete. The default is 300 seconds. Note that if you change the boot configuration for a blade because you want the boot process to involve more than the standard amount of diagnostic activity, the boot process will take longer. If you set the Boot Timeout to 0, the Boot Timeout facility is disabled.

      If you answer no, you will be asked whether you want none of the blades to be configured with the same Boot Timeout. And if you answer no to this question, you will be asked about each blade individually.

d. When asked whether you want all server blades to be automatically restarted if hung on bootup, answer yes or no.

If you answer yes, the System Controller will make three attempts to restart any server blades that do not complete the boot process. The boot process is treated as having failed to complete if the Boot Timeout is exceeded.

If you answer no, you will be asked whether you want none of the blades to be restarted automatically. And if you answer no to this question, you will be asked about each blade individually.

10. When asked if you want to configure the System Controller parameters, answer yes or no.

If you answer yes, the questions you are asked concern event reporting on the telnet interface, the setting of the System Controller's command prompt, the idle timeout period for System Controller user sessions, whether * characters should be echoed to the screen when a user types his or her password, and whether the System Controller uses the Network Time Protocol.

a. When asked if you want to enable CLI event reporting, type `y` if you want to receive event reports over telnet connections to the SSC.

Note that event reporting over the SSC's serial connection cannot be disabled.

b. For the level of events to be displayed (if you typed `y` in Step a), accept the default to see events of severity level 2 and above.

Level 2 means that MINOR, MAJOR, and CRITICAL events are displayed.

c. Specify the way you want the command-line prompt for the System Controller to be generated, or accept the default.

There are three possible types of prompt generation:

- `0=none` (This is the default setting):
  If you specify [`0=none`], the command-line prompt will appear as the prompt string that you specify in Step d below. The factory default string is `sc>`.
- `1=manual` (This option merely appends a > character to the prompt string):
  If you specify [`1=manual`], the command-line prompt will contain the string that you specify (in Step d below) but with a > appended to it. For example, if (in Step d) you type `B1600_1` as your prompt string, the prompt will appear as `B1600_1>`.
- `2=auto` (This option includes the hostname in the command prompt):
  If you specify [`2=auto`], the System Controller prompt will contain the System Controller's hostname (picked up from the DNS server but excluding any domain information). The position of the hostname in the prompt depends on the way you specify the prompt string in Step d below.

d. Type the characters you want to use to form the System Controller command-line prompt.

i. If you specified [2=auto] in Step c and you want the hostname to appear in the prompt, you must type at least the characters %m.

This macro (%m) indicates where in the prompt string you want the hostname to appear. If you do not type it, the hostname will not appear in the prompt. If you simply want the hostname to appear followed by an angle bracket > (for example, medusa-sc>), then type **%m** on its own.

If you want to include any characters before the hostname, type the characters you want and then type **%m**. For example, B1600-%m would yield the following prompt (assuming the sample hostname of medusa-sc): B1600-medusa-sc>.

If you want to include any characters after the hostname, type **%m** followed by the characters you want to include after it. For example, (assuming the sample hostname of medusa-sc), typing %m-B1600 would yield the following prompt: medusa-sc-B1600>.

Note – Do not include the > character in the string unless you want the prompt to contain two > characters. Also, remember that the string is limited to 16 characters.

---

ii. If you specified [1=manual] in Step c (to indicate that you want a > character appended to the prompt string), type [ENTER] to accept the string in its current form.

To change the prompt string, type a new prompt. Do not include the > character in the string unless you want the prompt to contain two > characters. For example, typing B1600_sc> would yield a prompt of B1600_sc>>.

iii. If you specified [0=none] in Step c (to indicate that the prompt should be exactly as specified with no hostname included and no > character appended), then type the prompt exactly as you want it to appear.

The factory default is sc>.

e. If you want to specify a particular console escape sequence for the switch, type it now.

The default escape sequence is #. (a hash followed by a dot).

f. Specify the timeout for the command-line interface.

The default of zero means that a user session does not time out after any period of inactivity.

g. Indicate whether you want the software to echo * characters to the screen when a user types his or her password.

h. Indicate whether you want to enable Network Time Protocol.

Answer yes if you have a time server on the network and you want to use it. Then, when prompted, type the IP address of the primary and secondary NTP servers.

11. **When asked if you want the network changes to take effect immediately, answer yes or no.**

    This question is only asked if you have made any changes to the System Controller's network settings. If you answer yes and you are configuring the System Controller using a telnet connection, you need to be aware that you might lose your telnet connection.

12. **Follow the instructions in Section 3.4.3, "Setting up the Switches in SSC0 and SSC1" on page 3-19 to set up the switch or switches.**

CODE EXAMPLE 3-3 Sample Output and Responses From `setupsc` (non-DHCP configuration)

```
sc> setupsc
Entering Interactive setup mode.
Use Ctrl-z to exit & save. Use Ctrl-c to abort.

Do you want to configure the enabled interfaces [y]?
Should the SC network interface be enabled [y]?
Should the SC telnet interface be enabled for new connections[y]?
Do you want to configure the network interface [y]?
Should the SC use DHCP to obtain its network configuration [n]?
Enter the SC IP address [192.156.203.139]:
Enter the SC IP netmask [255.255.255.0]:
Enter the SC IP gateway [192.168.1.1]:
Do you want to configure the the SC private addresses [y]?
Enter the SSC0/SC IP private address [192.168.1.200]:
Enter the SSC1/SC IP private address [192.168.1.201]:
Do you want to enable a VLAN for the SC [y]?
Enter VLAN ID [2]: 2
Enter the SMS IP address [0.0.0.0]:
Do you want to configure the managed system interface [y]? y
Should all blades be automatically restarted if OS is hung [y]?
Should all blades automatically restart F/W if hung on startup [y]?
Should all blades be configured with the same Boot Timeout [y]?
Should all blades be automatically restarted if hung on bootup [y]?
Should all blades be configured to power on automatically [y]?
Do you want to configure the System Controller parameters [y]?
Do you want to enable CLI event reporting via the telnet interface [y]?
Enter the level of events to be displayed over the CLI.
(0 = critical, 1 = major, 2 = minor) [2]:
Enter type of CLI prompt generation for SC ans switch:
(0 = none, 1 = manual, 2 = auto) [0]:
Enter the CLI prompt [sc>]:
Enter the CLI timeout (0, 60 - 9999 seconds) [0]:
Should the password entry echo *'s [y]?
Do you want to enable NTP [y]?
Enter the IP address of the primary NTP server [192.168.130.26]:
Enter the IP address of the secondary NTP server [192.168.130.26]:
Do you want the network changes to take effect immediately [y]?
sc>
```

CODE EXAMPLE 3-4 Sample Output and Responses From `setupsc` (DHCP configuration)

```
sc> setupsc
Entering Interactive setup mode.
Use Ctrl-z to exit & save. Use Ctrl-c to abort.

Do you want to configure the enabled interfaces [y]?
Should the SC network interface be enabled [y]?
Should the SC telnet interface be enabled for new connections[y]?
Do you want to configure the network interface [y]?
Should the SC use DHCP to obtain its network configuration [n]? y
Do you want to enable a VLAN for the SC [y]?
Enter VLAN ID [2]: 2
Enter the SMS IP address [0.0.0.0]:
Do you want to configure the managed system interface [n]?
Do you want to configure the managed system interface [y]? n
Do you want to configure the System Controller parameters [y]? n
Do you want the network changes to take effect immediately [y]?
sc>
```

3.4.2 Viewing the Configuration of the System Controller

To view the configuration of the System Controller, run the `showsc -v` command. All of the configurable properties of the System Controller are listed.

- **Type:**

CODE EXAMPLE 3-5 Default Configuration of Chassis (`showsc -v`)

```
sc> showsc -v
Sun Advanced Lights Out Manager for Blade Servers 1.0
ALOM-B 1.0

Release: 0.2.0, Created: 2003.01.10.11.03

Parameter                                Running Value      Stored Value
-------------------------------------------------------------------
Bootable Image:                          0.2.0 (Jan 10 03)
Current Running Image:                   0.2.0 (Jan 10 03)
SC IP address:                           192.156.203.139   129.156.203.139
SC IP netmask address:                   255.255.255.0     255.255.255.0
SC IP gateway address:                   192.168.1.1       192.168.1.1
SSC1/SC (Active) IP private address:     192.168.1.200     192.168.1.200
SSC0/SC (Standby) IP private address:    192.168.1.201     192.168.1.201
SMS IP address:                          0.0.0.0           0.0.0.0
```

CODE EXAMPLE 3-5 Default Configuration of Chassis (`showsc -v`) *(Continued)*

```
SC VLAN:                                  Disabled          Disabled
SC DHCP:                                  Enabled           Enabled
SC Network interface is:                  Enabled           Enabled
SC Telnet interface is:                   Enabled           Enabled
NTP:                                      Disabled          Disabled

Blade auto restart when hung:
S0                                        Disabled          Disabled
S1                                        Disabled          Disabled
S2                                        Disabled          Disabled
:
Blade auto poweron:
S0                                        Disabled          Disabled
S1                                        Disabled          Disabled
S2                                        Disabled          Disabled
:
The CLI prompt is set as:                 sc>               sc>
Event Reporting via telnet interface:     Enabled           Enabled
The CLI event level is set as:            CRITICAL          CRITICAL
The CLI timeout (seconds) is set at:      0                 0
Mask password with *'s:                   Disabled          Disabled

FRU   Software Version            Software Release Date
-------------------------------------------------------
S0    v1.1T30-SUNW,Serverblade1    Oct 24 2002 16:22:2
S1    v1.1T30-SUNW,Serverblade1    Oct 24 2002 16:22:24
S2    v1.1T30-SUNW,Serverblade1    Oct 24 2002 16:22:24
S3    v5.1.0-SUNW,Sun-Fire-B100x  Jun  5 2003 10:27:31
S4    v5.1.0-SUNW,Sun-Fire-B100x  Jun  5 2003 10:27:31
S5    v5.1.0-SUNW,Sun-Fire-B100x  Jun  5 2003 10:27:31
S6    v5.1.0-SUNW,Sun-Fire-B100x  Jun  5 2003 10:27:31
S7    v5.1.0-SUNW,Sun-Fire-B100x  Jun  5 2003 10:27:31
S8    v5.1.0-SUNW,Sun-Fire-B100x  Jun  5 2003 10:27:31
S9    v1.1T30-SUNW,Serverblade1    Oct 24 2002 16:22:24
S10   v1.1T30-SUNW,Serverblade1    Oct 24 2002 16:22:24
S11   v1.1T30-SUNW,Serverblade1    Oct 24 2002 16:22:24
S12   v1.1T30-SUNW,Serverblade1    Oct 24 2002 16:22:24
S13   v1.1T30-SUNW,Serverblade1    Oct 24 2002 16:22:24
S14   v1.1T30-SUNW,Serverblade1    Oct 24 2002 16:22:24
S15   v1.1T30-SUNW,Serverblade1    Oct 24 2002 16:22:24
sc>
```

3.4.3 Setting up the Switches in SSC0 and SSC1

This section tells you how to configure the IP address, netmask, and default gateway for the switches. By default the switches attempt to discover their IP configuration from DHCP. So if you have configured your DHCP server to provide the IP information for them, skip this section.

1. **To log into the switch in SSC0, type:**

```
sc> console ssc0/swt
```

2. **When prompted, type the username and password for the switch.**

3. **By default, the IP address and netmask for the switch are set by DHCP. You can set them manually by typing:**

```
Console#configure
Console(config)#interface vlan vlan id
Console(config-if)#ip address ip address  netmask
Console(config-if)#exit
```

where *vlan id* is the number of the VLAN containing the switch's network management port, NETMGT (if you are using the default switch configuration, this is 2), *ip address* is the IP address you want the switch to use, and *netmask* is the netmask you want to set.

For example, to specify the IP address and netmask for the switch in SSC0 in FIGURE 3-1, you would type:

```
Console#configure
Console(config)#interface vlan 2
Console(config-if)#ip address 192.168.1.202 255.255.255.0
Console(config-if)#exit
```

4. **By default, the default gateway is set by DHCP.**

You can set it manually by typing:

```
Console(config)#ip default-gateway ip address
Console(config)#exit
```

where *ip address* is the IP address of the device you are specifying as the default gateway.

5. **Save the new switch configuration.**

   Type the following in the switch console:

```
Console#copy running-config startup-config
Startup configuration file name []:filename
Write to FLASH Programming
-Write to FLASH finish
Success

Console#
```

   where *filename* is the name you want to give to the file that will contain your new startup configuration.

6. **Type `exit` to log out of the first switch.**

   Then type `#.` to exit the switch's command-line interface and return to the System Controller's `sc>` prompt.

7. **Now configure the second switch by following the instructions in Section A.10, "Copying the Configuration of the First Switch to the Second" on page A-10.**

   Alternatively, repeat Step 1 through Step 6 for the switch in SSC1.

What to Do Next

To set up the:

- B100s server blades, follow the instructions in Chapter 4.
- B100x and/or B200x Server blades, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

CHAPTER 4

Setting Up Server Blades and Performing Initial Diagnostics

This chapter tells you how to power on a server blade and access its console. It then tells you how to perform preliminary diagnostics using the various tools (apart from the Advanced Lights-out Management Software described in the *Sun Fire B1600 Blade System Chassis Software Setup Guide*) that are available.

For general information about running diagnostics on Solaris systems refer to the *OpenBoot Command Reference Manual* and the *SunVTS Users Guide*. These are available on the Software Supplement CD supplied with the Solaris Media Kit. You can also access them from:

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

The chapter contains the following sections:

Note – Whenever you are at a blade console, type **#.** to return to the active System Controller's `sc>` prompt.

4.1 Booting and Powering On Server Blades

4.1.1 Booting SPARC Solaris B100s Blades

When you apply power to a SPARC Solaris B100s server blade that is in its factory default state, the blade boots automatically from an operating system stub on its local hard disk. It then searches for a Network Install Server from which to complete the Operating Environment installation process.

To set up a Network Install Server, follow the instructions in the *Solaris Advanced Installation Guide* (supplied with the Solaris 8 12/02 media kit).

For supplementary information about using Web Start Flash Archives to speed up the process of configuring a series of server blades in a system chassis, refer to Appendix C in this manual.

4.1.2 Booting Linux or Solaris x86 B100x or B200x Blades for the First Time

Before you can use a Linux or Solaris x86 blade, you need to configure it temporarily to boot from the network. This is to enable it to perform the PXE boot process by which it first receives its operating system.

To set up a PXE server, follow the instructions in the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

Type the following command at the System Controller's `sc>` prompt to cause the blade to boot from the network

```
sc> bootmode bootscript="boot net" sn
```

where *n* is the number of the slot containing the blade.

Note – This command is effective for 10 minutes after that the BIOS reverts to its previous booting behavior. Therefore, to cause the blade to boot from the network you must power it on within 10 minutes of running the bootmode command. If the blade was already powered on when you ran the bootmode command, then to cause it to boot from the network you must reset the blade within 10 minutes by typing:
`sc> reset sn`

4.1.3 Powering on the Blades

When you are ready, power on a server blade and boot it by following the instructions below:

1. **Power on the blade.**

   Type:

   ```
   sc> poweron sn
   ```

   where *n* is the number of the slot containing the server blade.

2. **Log into the console of the server blade to view (and/or participate in) the booting process.**

   Type the following at the `sc>` prompt to access the blade's console:

   ```
   sc> console sn
   ```

   where *n* is the number of the slot containing the blade.

   Your next action depends on which of the Solaris installation methods you have chosen from the *Solaris Advanced Installation Guide*.

3. **For SPARC Solaris blades, if you require you can interrupt the boot process either to control it yourself or to run diagnostics.**

   To interrupt the boot process[1], type:

   ```
   sc> break sn
   ```

   where *n* is the number of the slot containing the blade.

1. For information about configuring a blade not to accept `break` commands, refer to the `kbd(1)` MAN page.

4. **Follow the instructions in the remainder of this chapter if you want to perform initial diagnostics on a SPARC Solaris server blade.**

   For information about performing diagnostics on a Sun Fire B10n Content Load Balancing Blade, refer to the *Sun Fire B10n Content Load Balancing Administration Guide*.

Note – Whenever you are at a blade console, type **#.** to return to the active System Controller's `sc>` prompt.

4.2 Using Power-on Self-test (POST) Diagnostics on B100s Blades

This section tells you how to control the POST diagnostic process that (by default) takes place on a B100s (SPARC Solaris) blade during booting.

4.2.1 Controlling the Amount of Diagnostic Testing

There are three levels of diagnostic testing available for POST diagnostics:

- `max` (maximum level)
- `min` (minimum level)
- `off` (no testing)

Set the level you require by using the OpenBoot PROM variable `diag-level`. The default setting for `diag-level` is `min`. To set it, type:

```
ok diag-level level
```

where *level* is `min`, `max`, or `off`.

4.2.2 Overriding the Blade's Diagnostic Settings From the System Controller

You can use the System Controller's `bootmode` command to override the `diag-level` and `diag-switch?` settings temporarily.

- **To cause the server blade to boot with diagnostics when it is not configured to do so:**

  a. **Type `#.` to return to the System Controller's command-line interface.**

  b. **Type:**

```
sc> bootmode diag sn
```

  where *n* is the number of the slot whose blade you are intending to configure.

  The effect of this command is equivalent to the effect of setting `diag-switch?` to `true` and `diag-level` to `min` for a single booting only. (If `diag-level` on the blade is set to `max` or `min`, the `bootmode` command does not alter its setting.)

- **To cause the server blade to boot without running diagnostics when it is configured to run diagnostics:**

  a. **Type `#.` to return the System Controller's command-line interface.**

  b. **Type:**

```
sc> bootmode skip_diag sn
```

  where n is the number of the slot whose blade you are configuring.

  The effect of this command is equivalent to the effect of setting `diag-switch?` to `false`.

4.2.3 Running POST Diagnostics

If the OpenBoot PROM (OBP) variable `diag-switch?` is set to `true`, then POST diagnostics will run automatically when you power on the server. However, the default setting for `diag-switch?` is `false`.

To initialize POST diagnostics, you need to set the `diag-switch?` variable to `true` and `diag-level` to `max` or `min` (and not `off`). When you have done this, you need to reset the server blade. Follow the instructions below:

1. **From the `ok` prompt on the server blade, type:**

```
ok setenv diag-switch? true
```

2. **Type `#.` to return to the System Controller's command-line interface.**

3. **Power cycle the blade:**

   Type:

```
sc> poweroff sn
```

   where *n* is the slot number of the blade.

   Then type:

```
sc> poweron sn
```

4. **Within two-to-three seconds (if possible) of powering on the blade, access the blade's console to view the diagnostics output.**

   Type:

```
sc> console sn
```

5. **When booting is complete, you can inspect the boot-time console output by typing #. to return to the System Controller's command-line interface and then typing:**

```
sc> consolehistory boot sn
```

   If POST detects an error, it displays an error message describing the failure.

   If POST detects a "fatal" error (for example, a hardware problem with the onboard memory or the CPU), it powers off the server blade and lights the blade's Fault LED).

4.3 Using OpenBoot Diagnostics (obdiag) on SPARC Solaris Blades

To run OpenBoot Diagnostics, do the following:

1. **From the** ok **prompt, type:**

```
ok setenv auto-boot? false
ok reset-all
```

2. **Type:**

```
ok obdiag
```

This displays the OpenBoot Diagnostics menu:

TABLE 4-1 The obdiag Menu

```
                              obdiag

1 bscv@0,0              2 ide@d                 3 network@a
4 network@b             5 pmu@3                 6 rtc@0,70
7 serial@0,3f8

       Commands: test test-all except help what setenv exit

               diag-passes=1 diag-level=max test-args=
```

The tests are described in TABLE 4-2. Note the number that corresponds to the test you want to perform, and use it with the test command. For example, to run a test on the primary Ethernet port, type:

```
obdiag> test 3
Hit the spacebar to interrupt testing
Testing /pci@1f,0/network@a ..........................passed
Pass:1 (of 1) Errors:0 (of 0) Tests Failed:0 Elapsed Time: 0:0:0:2

Hit any key to return to the main menu.
```

3. **When you have finished testing, exit OpenBoot Diagnostics and restore the value of `auto-boot?` to true.**

   To do this, type:

```
obdiag> exit
ok setenv auto-boot? true
ok auto-boot? true
ok boot
```

The function of each test is shown below.

TABLE 4-2 Open Boot Diagnostics Tests

1	`bscv@0,0`	tests the Blade Support Chip
2	`ide@d`	tests the ide controller
3	`network@a`	tests the primary Ethernet interface
4	`network@b`	tests the secondary ethernet interface
5	`pmu@3`	tests the power management unit
6	`rtc@0,70`	tests the real-time clock device
7	`serial@0,3f8`	tests the serial interface to the System Controller

4.4 Using Other OpenBoot PROM Commands on SPARC Solaris Blades

This section describes the OpenBoot PROM commands you can run and explains what each command does.

The `show-devs` *Command*

Use the OpenBoot PROM `show-devs` command to list the devices in the OBP device tree.

The `printenv` *Command*

Use the OpenBoot PROM `printenv` command to display the OpenBoot PROM configuration variables stored in the system NVRAM. The display includes the current values for these variables as well as the default values. You can also specify a variable to display the current value for that variable only. For example, typing `printenv diag-level` will print the current value for the `diag-level` variable.

The `watch-clock` *Command*

The `watch-clock` command displays a number that increments once per second. During normal operation the seconds counter repeatedly increments from 0 to 59. The following shows an example snapshot of output from the `watch-clock` command.

```
ok watch-clock
Watching the 'seconds' register of the real time clock chip.
It should be 'ticking' once a second.
Type any key to stop.
4
```

The `watch-net` *and* `watch-net-all` *Commands*

The `watch-net` and `watch-net-all` commands monitor Ethernet packets on the blade's Ethernet interfaces. Good packets received by are indicated by a period (.). Errors such as the framing error and the cyclic redundancy check (CRC) error are indicated with an X and an associated error description.

The following examples show `watch-net` and the `watch-net-all` command output.

```
ok watch-net
1000 Mbps FDXLink up
Link is -- up
Looking for Ethernet Packets.
'.' is a Good Packet. 'X' is a Bad Packet.
Type any key to stop.
.............................
ok
```

```
ok watch-net-all
/pci@1f,0/network@b
1000 Mbps FDXLink up
```

```
Link is -- up
Looking for Ethernet Packets.
'.' is a Good Packet. 'X' is a Bad Packet.
Type any key to stop.
.............................
/pci@1f,0/network@a
1000 Mbps FDXLink up
Link is -- up
Looking for Ethernet Packets.
'.' is a Good Packet. 'X' is a Bad Packet.
Type any key to stop.
.............................
ok
```

The `probe-ide` *Command*

The `probe-ide` command causes the IDE controller on the blade to send an enquiry to each of its four possible IDE devices (in fact there is only ever one device connected to the IDE controller). If you observe a `not present` response for the primary master device, this indicates a problem with the hard disk or with the connection to the hard disk from the IDE controller.

CODE EXAMPLE 4-1 `probe-ide` Output Message

```
ok probe-ide
 Device 0  ( Primary Master )
          ATA Model: TOSHIBA MK3019GAB

 Device 1  ( Primary Slave )
      Not Present

 Device 2  ( Secondary Master )
      Not Present

 Device 3  ( Secondary Slave )
      Not Present
```

4.5 Using SunVTS on SPARC Solaris Blades

SunVTS, the Sun Validation and Test Suite, is an online diagnostics tool that you can use to verify the configuration and functionality of hardware controllers, devices, and platforms. SunVTS is available from the *Software Supplement for the Solaris Operating Environment* CD.

You need to run it from a Solaris prompt:

- command line interface
- graphical interface within a windowed desktop environment

SunVTS software lets you view and control a testing session on a remotely connected server. Below is a list of example tests:

TABLE 4-3 SunVTS Tests

SunVTS Test	Description
disktest	Verifies local disk drives
fputest	Checks the floating-point unit
nettest	Checks the networking hardware on the system CPU board and on network adapters contained in the system.
pmem	Tests the physical memory (read only)
vmem	Tests the virtual memory (a combination of the swap partition and the physical memory)
bsctest	Tests the Blade Support Chip on the server blade.

Note – Sun VTS is not currently available for B100x and B200x blades running Solaris x86. For information about tools for performing memory diagnostics on these blades, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

4.5.1 Finding Out If SunVTS is Installed

To check whether SunVTS is already installed on a server blade, type:

```
# pkginfo -l SUNWvts
```

- If SunVTS software is loaded, information about the package will be displayed.
- If SunVTS software is not loaded, you will see the following error message:

```
ERROR: information for "SUNWvts" was not found
```

4.5.2 Installing SunVTS

SunVTS is distributed on the *Software Supplement for the Solaris Operating Environment* CD. For information about installing it, refer to the *Sun Hardware Platform Guide*. The default directory to use when you install SunVTS software is `/opt/SUNWvts`.

4.5.3 Running SunVTS

To test a Sun Fire B100s server blade by running a SunVTS session from a workstation using the SunVTS graphical user interface, follow the procedure below:

1. **Use the `xhost` command on the workstation to give the server blade access to the local display.**

   Type:

   ```
   # /usr/openwin/bin/xhost + remote_hostname
   ```

   where *remote_hostname* is the host name of the server blade.

2. **Remotely log into the server blade as superuser or root.**

3. **Type:**

   ```
   # cd /opt/SUNWvts/bin
   # ./sunvts -display local_hostname:0
   ```

   where *local_hostname* is the name of the workstation you are using.

Note – The directory `/opt/SUNWvts/bin` is the default directory for SunVTS software. If you have the software installed in a different directory, use that path instead.

When you start SunVTS software, the SunVTS kernel probes the test system devices and displays the results on the Test Selection panel. There is an associated SunVTS test for each hardware device on your system.

You can fine-tune your testing session by selecting the appropriate check boxes for each of the tests you want to run.

CHAPTER 5

Installing a Chassis Containing B100s Blades into Separated Data and Management Networks

This chapter contains the following sections:

Note – For information about setting up network redundancy on a chassis containing Linux and/or Solaris x86 blades, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

5.1 Taking Advantage of Having Two Switches in the System Chassis

This chapter tells you how to set up the Sun Fire B1600 blade system chassis for use in an environment that separates the data and management networks. If you have dual SSCs installed in the chassis, the instructions enable you to take advantage of the presence of two switches to give your SPARC server blades two connections each to your network.

FIGURE 5-1 shows a sample network containing a Sun Fire B1600 blade system chassis, and the subsequent sections use this diagram and the IP addresses marked on it to illustrate the steps you need to perform.

If you have Solaris blades installed in the chassis, the sample `/etc/hosts` (CODE EXAMPLE 5-1) and `/etc/netmasks` (CODE EXAMPLE 5-2) files provided in this chapter illustrate how to edit the files on your Name Server to simplify the process of configuring Solaris on the blades. Use these sample administration files for guidance, substituting your own IP addresses and host names for the ones used in the sample network illustrated in FIGURE 5-1.

Note – As noted in Chapter 3, if you have dual SSCs installed, then, when you are considering how to integrate the chassis into your network environment, you need to remember that the chassis contains two switches. Although only one of its System Controllers is active at any one time, both of its switches are active all the time. This means that, in a system chassis that is working normally, both switches are providing the server blades with continuous network connectivity. However, if for any reason one switch fails, the other switch continues to provide network connectivity. (Also, if either System Controller fails, the switch inside the same SSC module continues to provide network connectivity; the switches operate independently of the System Controllers even though they are physically located in the same enclosure.)

This chapter now tells you how to take advantage of the presence of two switches by using VLANs in conjunction with IPMP (IP Network Multipathing) on your B100s (SPARC Solaris) blades to provide fully redundant connections from Solaris server blades to the data and management networks.

If you have Linux blades installed, you cannot currently take advantage of the presence of two switches because the bonding facility (this is the Linux equivalent of IPMP) is not currently available.

To take advantage of the redundancy offered by the second switch inside the system chassis, we recommend you to:

- Operate the system chassis always with two SSCs installed.
- Make sure that the cable connections from the eight uplink ports to the subnets on your wider network are exactly duplicated on the eight uplink ports of the second switch.
- Copy the configuration file of the first switch you configure over to the redundant switch before setting the IP address, netmask, and default gateway for the switch. For information about how to do this, see Section A.10, "Copying the Configuration of the First Switch to the Second" on page A-10.
- Specify IP addresses (in the `/etc/hosts` file on the Name Server) suitable for an IP Network Multipathing (IPMP) configuration that supports redundant interfaces to the data network and the management network from each server blade (see CODE EXAMPLE 5-1). There are fewer IP addresses listed for the blades in CODE EXAMPLE 5-1 than there were in the sample `/etc/hosts` file in Chapter 3 (see CODE EXAMPLE 3-2). This is because only one published interface is required for each server blade when you use IPMP.
- Specify the MAC and IP addresses for both ethernet interfaces on each server blade when you set up the `/etc/ethers` file on your Name Server.

5.2 Preparing the Network Environment Using DHCP

If you are using DHCP, make sure that the DHCP server for the System Controllers and switches is on the management network, and that the DHCP server for the blades is on the data network.

For information about setting up the Network Install Servers and DHCP servers for B100s blades, see Chapter 1, Chapter 3, and Appendix B.

Note – If you use DHCP to configure the IP settings for the two interfaces on each server blade, you cannot use IPMP to configure redundant connections to the physical network or multiple connections to VLANs.

5.3 Preparing the Network Environment Using Static IP Addresses

FIGURE 5-1 shows a network similar to the sample configuration in Chapter 3 but with the 100Mbps network management port (NETMGT) on both SSCs now connected to a different switch from the data uplink ports. This new external switch is on a different subnet from the switch that the data uplink ports on the chassis are connected to. It is a subnet dedicated to network management traffic and it therefore also contains both of the System Controllers and switches in the chassis. A management VLAN (VLAN 2) contains both System Controller interfaces and both switch management ports. All the server blades and uplink ports are on VLAN 1.

FIGURE 5-1 also shows the connection of the `ce0` interface on each blade to the switch in SSC0, and the connection of the `ce1` interface on each blade to the switch in SSC1. Note that each server blade interface now has four IP addresses associated with it instead of one. These four addresses are used by the IPMP driver to enable the interfaces to function as redundant connections (see Section 5.5, "Setting up SPARC Solaris Server Blades Using IPMP for Network Resiliency" on page 5-9).

As in FIGURE 3-1 (see Chapter 3), one or more of the eight uplink ports on each switch in FIGURE 5-1 are connected to an external switch that has an Install Server (also containing a Name Server) connected to it. This external switch also has a router (with IP address 192.168.1.1) connected to it that acts as the default gateway from the chassis to the wider network.

Note – Note that there is no direct network connection in FIGURE 5-1 from the management port (NETMGT) in the switch to the server blade ports. This means that, by default, you cannot manage the server blades directly from the management network. This is a security feature to protect the management network from the possibility of hostile attack from the data network. For information about permitting specified traffic from the server blades to the management port, see Appendix A and Chapter 6.

Before installing the Sun Fire B1600 blade system chassis into an environment like the one illustrated in FIGURE 5-1 (in other words, one that separates the data and management networks), you need to edit the `/etc/hosts`, `/etc/ethers`, and `/etc/netmasks` files on your Solaris Name Servers on the data and management networks:

- CODE EXAMPLE 5-1 is a sample `/etc/hosts` file containing IP addresses and host names for the chassis on the data network in the environment illustrated in FIGURE 5-1.

- CODE EXAMPLE 5-1 is a sample `/etc/hosts` file containing IP addresses and host names for the components of the chassis (the two SSCs and switches) that are included in the management network illustrated in FIGURE 5-1.
- CODE EXAMPLE 5-2 is a sample `/etc/netmasks` file containing netmasks for the IP network numbers used in the sample network in FIGURE 5-1.

Note – For each B100s server blade, only the published IP addresses (not the test IP addresses used by IPMP) need to be registered in the `/etc/hosts` file on the name server. However, the test addresses for each blade must be clearly reserved in a comment so that other network administrators know they are not available for use (see CODE EXAMPLE 5-1).

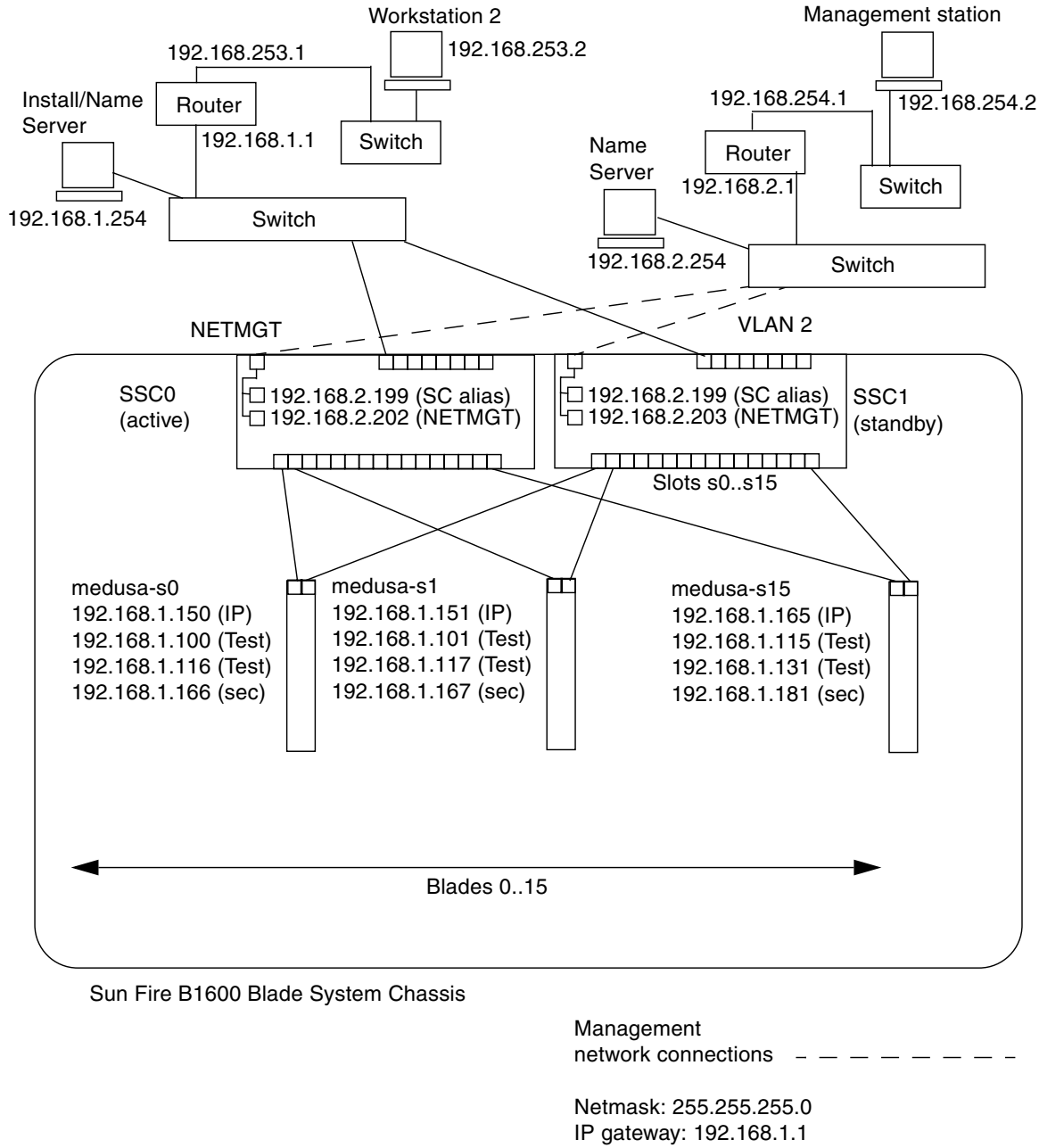

FIGURE 5-1 Sample Network Configuration Using a Management VLAN

CODE EXAMPLE 5-1 Sample `/etc/hosts` File on the Name Server (on the Data Network)

```
# Internet host table
127.0.0.1       localhost

192.168.1.254   datanet-nameserver    # loghost
192.168.1.1     datanet-router-1      # Data network router
                                      # (default gateway)
192.168.2.199   medusa-sc             # Medusa - alias address for active SC

192.168.253.1   datanet-router-253    # Data network router (client side)
192.168.253.2   dataclient-ws1        # Data client network workstation

# 192.168.1.100 -> 192.168.1.131 are reserved for private use by the
# Sun Fire B1600 Blade System Chassis called Medusa. They are test addresses for
# the IPMP driver on each server blade.
#
# Published IP addresses for server blades in Medusa.
192.168.1.150   medusa-s0
192.168.1.151   medusa-s1
192.168.1.152   medusa-s2
192.168.1.153   medusa-s3
192.168.1.154   medusa-s4
192.168.1.155   medusa-s5
192.168.1.156   medusa-s6
192.168.1.157   medusa-s7
192.168.1.158   medusa-s8
192.168.1.159   medusa-s9
192.168.1.160   medusa-s10
192.168.1.161   medusa-s11
192.168.1.162   medusa-s12
192.168.1.163   medusa-s13
192.168.1.164   medusa-s14
192.168.1.165   medusa-s15
```

CODE EXAMPLE 5-2 Sample `/etc/netmasks` File on the Name Server (on the Data network)

```
#
# The netmasks file associates Internet Protocol (IP) address
# masks with IP network numbers.
#
#       network-number  netmask
#
# The term network-number refers to a number obtained from the
# Internet Network Information Center. Currently this number is
# restricted to being a class A, B, or C network number.
#
# Routing guidelines.
#
# Both the network-number and the netmasks are specified in
# "decimal dot" notation, e.g:
#
#               128.32.0.0 255.255.255.0
#
192.168.1.0     255.255.255.0

192.168.2.0     255.255.255.0
192.168.253.0   255.255.255.0
#
```

5.4 Configuring the System Controllers and Switches

To configure the System Controllers and switches for the type of configuration illustrated in FIGURE 5-1, follow the instructions in Section 3.4, "Configuring the System Controllers and Switches" on page 3-9. However, remember that the IP addresses you assign to the System Controllers and switches need to be on the management subnet.

5.5 Setting up SPARC Solaris Server Blades Using IPMP for Network Resiliency

The instructions in this section tell you how to use the Solaris IP Network Multipathing (IPMP) facility to take advantage of the redundant connections from each server blade to the switches in the chassis. A server blade's two 1000Mbps Ethernet interfaces are labeled respectively `ce0` and `ce1` (`ce0` is connected to the switch in SSC0, and `ce1` is connected to the switch in SSC1). When the Sun Fire B1600 blade system chassis is fully operational, both switches are constantly active.

The IPMP driver on the server blade works by periodically pinging the default gateway from both Ethernet interfaces. If for any reason one of the pings fails (indicating that the path to the network is no longer available on the interface that was used to perform the ping) the IPMP driver ensures that network traffic uses only the interface that remains valid. Both interfaces can be active (in which case they each require a separate IP address). Alternatively one can be a standby interface that takes over the IP address of the active one if that interface fails.

The active/active configuration requires four IP addresses: one for each interface plus one test address for each interface. The active/standby configuration requires three IP addresses. In either case two test addresses are used privately by the IPMP driver to perform the ping process. If it does not receive a reply from a ping on the test address associated with one interface it knows that that interface has failed and it directs all network traffic for either interface over the valid one. If you have an active/active configuration, it simply stops using the non-valid interface. If you have an active/standby configuration and the non-valid interface is the active one, it assigns the IP address to the standby interface which now becomes the active interface.

Because both switches inside the chassis are active (when the chassis is working normally), the instructions in this chapter tell you how to perform an active/active configuration. For information about performing an active/standby configuration, refer to the *IP Network Multipathing Administration Guide* (816-0850).

The IP addresses you require for each physical interface on a blade are as follows:

- A primary IP address.
- A secondary IP address (this is only required for the active/active configuration).

  The primary and secondary IP addresses are (or can) both be registered on a Name Server. They are the addresses by which other devices on the network communicate with a blade.

- Two other IP addresses are required (one per interface) for the ping process described above. These two addresses are referred to in this manual as “test” addresses. They are private to the IPMP driver (in other words, they are not registered on the Name Server).

In this chapter, the instructions tell you how to set up IPMP for two physical interfaces. In the next chapter, instructions are provided for setting up multiple pairs of virtual IPMP interfaces, each pair providing redundant interfaces to separate VLANs.

5.5.1 Configuring the Solaris Server Blade

This section tells you how to configure IPMP on a server blade so that the two Ethernet interfaces both actively transmit and receive data. For purposes of illustration the instructions use sample configuration input from the network scenario described in Section 5.3, “Preparing the Network Environment Using Static IP Addresses” on page 5-4.

TABLE 5-1 summarizes the information you would need to give the IPMP driver on the server blade in Slot 0 of the system chassis illustrated in FIGURE 5-1.

Note – You need to perform the instructions in this section on each server blade that requires a redundant connection to the network.

TABLE 5-1 Sample IPMP Configuration for a Server Blade

IPMP Configuration Variable	Value for Sample Server Blade in Slot 0
Network adapter interfaces	`ce0` (active) `ce1` (active)
Interface group name	`medusa_grp0`
IP address and host name (primary)	192.168.1.150 (`medusa-s0`)
Secondary IP address and host name (secondary)	192.168.1.166 (`medusa-s0-sec`)
Test IP address and host name (`ce0`)	192.168.1.100 (medusa-s0-0)
Test IP address and host name (`ce1`)	192.168.1.116 (medusa-s0-1)
Netmask	255.255.255.0
Is the server blade to perform network routing?	No

1. **Perform a preliminary setup of Solaris by following the instructions in Chapter 3.**

   When you have done this, type #. to return from a server blade console to the `sc>` prompt.

2. **Log in as root to the console of the server blade whose interfaces you want to configure.**

   Type the following at the `sc>` prompt:

```
sc> console sn
```

   where *n* is the number of the slot containing the server blade you want to log into.

3. **Edit the `/etc/hosts` file on the server blade itself to add the blade's two test IP addresses.**

   For a blade using the sample addresses in TABLE 5-1, you would need to add the last two lines of the following file:

```
#
# /etc/hosts on the server blade in system chassis Medusa, slot 0
#
127.0.0.1       localhost       loghost

192.168.1.150   medusa-s0       # Data Address
192.168.1.166   medusa-s0-sec   # Secondary Data Address
192.168.1.100   medusa-s0-0     # Test Address for ce0
192.168.1.116   medusa-s0-1     # Test Address for ce1
```

4. **Set the netmask in the server blade's `/etc/netmasks` file.**

   For a blade using the sample addresses in TABLE 5-1, you would need to add the folowing line:

```
192.168.1.0     255.255.255.0
```

5. **Disable routing, because the server blade is not being used to perform routing.**

   Type:

```
# touch /etc/notrouter
# ndd -set /dev/ip ip_forwarding 0
```

6. **Create the network interfaces by typing:**

```
# ifconfig ce0 plumb
# ifconfig ce1 plumb
```

7. **Create an IPMP group named** `medusa_grp0` **containing network interfaces** `ce0` **and** `ce1`**:**

```
# ifconfig ce0 group medusa_grp0
# ifconfig ce1 group medusa_grp0
```

When you execute these commands, you might see the following `syslog` messages:

```
Sep  3 00:49:58 medusa-s0 in.mpathd[298]: Failures cannot be
detected on ce0 as no IFF_NOFAILOVER address is available
```

These messages simply warn that failures cannot be detected until test addresses have been established on the interfaces.

8. **Create an address on** `ce0` **and** `ce1` **for data transmission and mark it to** `failover` **if an interface failure is detected.**

```
# ifconfig ce0 medusa-s0 netmask + broadcast + failover up
Setting netmask of ce0 to 255.255.255.0

# ifconfig ce1 medusa-s0-sec netmask + broadcast + failover up
Setting netmask of ce1 to 255.255.255.0
```

9. **Configure a test address on each network interface.**

   These will be used by `mpathd` to detect interface failures. You need to use the `-failover` flag. This causes `in.mpathd` to use the address as a test address (in other words, an address that cannot pass to the other interface and therefore does not fail over):

```
# ifconfig ce0 addif medusa-s0-0 netmask + broadcast + -failover
deprecated up
Created new logical interface ce0:1
Setting netmask of ce0:1 to 255.255.255.0

# ifconfig ce1 addif medusa-s0-1 netmask + broadcast + -failover
deprecated up
Created new logical interface ce1:1
Setting netmask of ce1:1 to 255.255.255.0
```

10. **To enable the new interface configuration to survive a reboot, create a** `hostname.ce0` **and a** `hostname.ce1` **file in the** `/etc` **directory.**

    A sample file for `hostname.ce0` is as follows:

```
medusa-s0 netmask + broadcast + \
group medusa_grp0 up \
addif medusa-s0-0 deprecated -failover \
netmask + broadcast + up
```

    A sample file for `hostname.ce1` is as follows:

```
medusa-s0-sec netmask + broadcast + \
group medusa_grp0 up \
addif medusa-s0-1 deprecated -failover \
netmask + broadcast + up
```

11. **Inspect the configuration of the two network adapters.**

    Type:

```
# ifconfig -a
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
        inet 127.0.0.1 netmask ff000000
ce0: flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
        inet 192.168.1.150 netmask ffffff00 broadcast 192.168.1.255
        groupname medusa_grp0
        ether 0:3:ba:19:26:3
ce0:1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4, NOFAILOVER> mtu 1500 index 2
        inet 192.168.1.100 netmask ffffff00 broadcast 192.168.1.255
ce1: flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
        inet 192.168.1.166 netmask ffffff00 broadcast 192.168.1.255
        groupname medusa_grp0
        ether 0:3:ba:19:26:4
ce1:1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4, NOFAILOVER> mtu 1500 index 3
        inet 192.168.1.116 netmask ffffff00 broadcast 192.168.1.255
```

The output above shows that four addresses have been defined (the sample addresses from TABLE 5-1). The two IPMP test addresses (associated with `ce0:1` and `ce1:1` respectively) are marked NOFAILOVER. This means that they will not be transferred to the surviving interface in the event of a failure.

12. **Test IPMP by temporarily removing one SSC from the chassis.**

    This will cause the following error messages to be displayed on the console:

```
Sep 3 01:08:50 medusa-s0 in.mpathd[29]: NIC failure detected on
ce0 of group medusa_grp0
Sep 3 01:08:50 medusa-s0 in.mpathd[29]: Successfully failed over
from NIC ce0 to NIC ce1
```

Note – It takes approximately 10 seconds for the IPMP daemon to detect and recover from a network failure with the default configuration. The configuration of the IPMP daemon is defined in the `/etc/default/mpathd` file.

CHAPTER 6

Adding Blade Management and VLAN Tagging for SPARC Solaris Blades

This chapter tells you how to configure the system chassis to permit secure management of server blades from the management network.

This chapter contains the following sections:

Note – For information about setting up redundant virtual connections for Linux and/or Solaris x86 blades, refer to the *Sun Fire B100x and B200x Server Blade Installation and Setup Guide*.

6.1 Introduction

This chapter tells you how to refine the configuration in Chapter 5 to enable network administrators to perform management tasks on the server blades from the management network (that is by telnet connections direct to the server blades) without compromising the security of the management network.

In FIGURE 6-1 there are dotted lines from the server blade ports in the chassis's switches to the management port (NETMGT). There are also dotted lines from the server blades themselves to the management port in each switch. These dotted lines represent links between components or devices that are members of the management VLAN (VLAN 2). By default VLAN 2, which contains the management port (NETMGT) on the switch, does not include any server blade ports. So to configure the chassis to support a network environment like the one in FIGURE 6-1 you must reconfigure these ports manually. For information on how to do this, see Section 6.3, "Configuring the System Controller and Switches" on page 6-5.

Also, by default, no network traffic is allowed to pass from the server blade ports - through the switch's packet filter - to the management port. This is a security feature and you must exercise caution when configuring the switch to permit traffic to pass through its packet filter. The instructions in Section A.12, "Enabling Secure Management of Blades" on page A-16 tell you how to permit only specific protocols to pass through the packet filter.

Finally, because the instructions in this chapter tell you how to include the server blades in the management network (VLAN 2), they also tell you how to modify the IPMP setup on the server blades so that, not only does each blade have a redundant connection to the data network (as described in Chapter 5), but each one also has a redundant connection to the management network (VLAN 2).

6.2 Preparing the Network Environment

This section contains an illustration of the configuration from the previous chapter but with the enhancements described in the introduction above plus examples of the IPMP information required to create the redundant connections from each blade to the management network. The section also contains a sample `/etc/hosts` file for the Name Server on the management network. The administration files on the data network remain the same as in Chapter 5. However, the `/etc/hosts` file on the management network's Name Server needs to contain IP addresses (on the management subnet) for each server blade as well as for both SSCs and switches in the chassis (see CODE EXAMPLE 6-1).

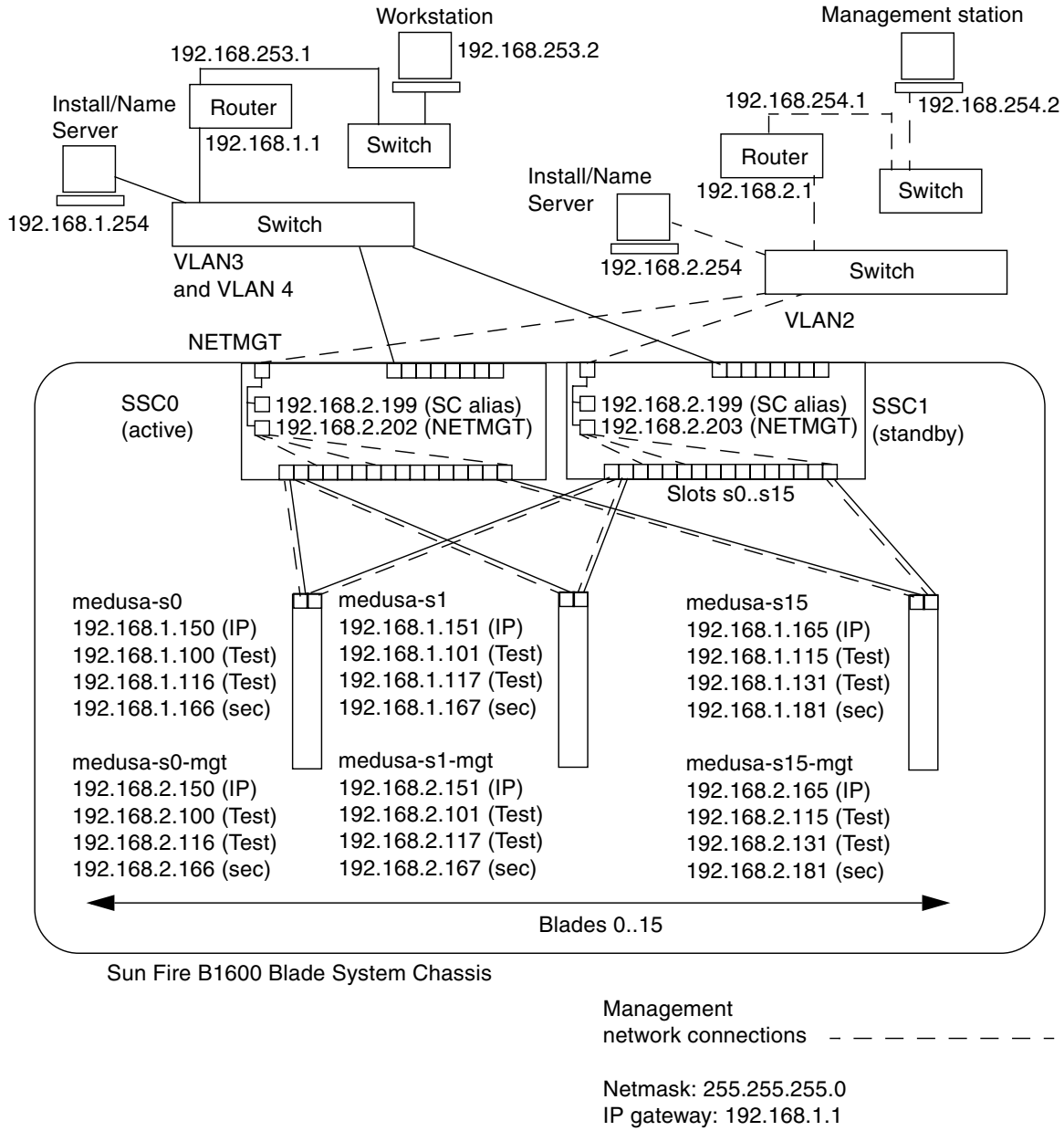

FIGURE 6-1 Sample Network Configuration With a Management VLAN that Includes Server Blades

CODE EXAMPLE 6-1 Sample `/etc/hosts` file on the Name Server (on the Management Network

```
# Internet host table
# This is the sample /etc/hosts file for the name-server on the management
# network.

192.168.2.1      mgtnet-router-1    # Management network router
#                                     (default gateway)
192.168.2.254   mgtnet-nameserver  # Management network install/name server
192.168.254.1   mgtnet-router-254  # Management network router (client side)
192.168.254.2   mgtnet-ws          # Management network workstation

192.168.2.199   medusa-sc          # Medusa - alias IP address for active SC
192.168.2.200   medusa-ssc0        # Medusa - ssc0/sc
192.168.2.201   medusa-ssc1        # Medusa - ssc1/sc
192.168.2.202   medusa-swt0        # Medusa - ssc0/swt
192.168.2.203   medusa-swt1        # Medusa - ssc1/swt

# 192.168.2.100 -> 192.168.2.131 are reserved for private use by the
# Sun Fire B1600 Blade System Chassis called medusa. They are test addresses for
# the IPMP driver on each server blade.

192.168.2.150   medusa-s0-mgt
192.168.2.151   medusa-s1-mgt
192.168.2.152   medusa-s2-mgt
192.168.2.153   medusa-s3-mgt
192.168.2.154   medusa-s4-mgt
192.168.2.155   medusa-s5-mgt
192.168.2.156   medusa-s6-mgt
192.168.2.157   medusa-s7-mgt
192.168.2.158   medusa-s8-mgt
192.168.2.159   medusa-s9-mgt
192.168.2.160   medusa-s10-mgt
192.168.2.161   medusa-s11-mgt
192.168.2.162   medusa-s12-mgt
192.168.2.163   medusa-s13-mgt
192.168.2.164   medusa-s14-mgt
192.168.2.165   medusa-s15-mgt
```

6.3 Configuring the System Controller and Switches

If you have already set up the System Controller and switches in the system chassis by following the instructions in the previous chapters, then go straight to Section 6.3.1, "Adding the Server Blades to the Management VLAN on the Switches in SSC0 and SSC1" on page 6-5.

Otherwise, follow the instructions in Chapter 5 but do not configure the switch in SSC1, because the instructions below (Section 6.3.1, "Adding the Server Blades to the Management VLAN on the Switches in SSC0 and SSC1" on page 6-5) involve copying the entire configuration of the switch in SSC0 onto the switch in SSC1.

6.3.1 Adding the Server Blades to the Management VLAN on the Switches in SSC0 and SSC1

The instructions in this section tell you how to add the server blades to the management VLAN, which is VLAN 2 by default (in other words, by default VLAN2 contains the management port, NETMGT). VLAN 1 is also set up by default on the switch. This VLAN contains all the switch's server blade and uplink ports. However, to demonstrate the use of the switch's VLAN configuration facilities, the instructions in this section will use VLAN 3 instead of VLAN 1 for the data network.

In these instructions the management VLAN (VLAN 2) and the data VLAN (VLAN 3) are tagged. However, the instructions also tell you to create an additional VLAN for blade booting (VLAN 4). This handles untagged traffic generated by the blades during the Solaris Operating Environment Network Install process.

This traffic on the boot VLAN (VLAN 4) can be tagged or untagged when it leaves the system chassis. In the sample commands in this section it is tagged. (The instructions assume that the devices outside the chassis are VLAN-aware, and VLAN 4 is assumed to contain the Network Install Server used by the server blades.)

Note – If you reset the switch while you are performing the instructions in this section, you must save the configuration first. If you do not, you will lose all of your changes. To save the configuration, follow the instructions in Section A.9, "Saving Your Switch Settings" on page A-10.

---

1. **From the `sc>` prompt, log into the console to configure the switch in SSC0.**

   To log into the switch in SSC0, type:

```
sc> console ssc0/swt
```

2. **When prompted, type your user name and password.**

3. **At the `Console#` prompt on the switch's command line, type:**

```
Console#configure
```

4. **Enter the switch's VLAN database by typing:**

```
Console(config)#vlan database
```

5. **Set up the VLAN for the data network and for the boot network by typing:**

```
Console(config-vlan)#vlan 3 name Data media ethernet
Console(config-vlan)#vlan 4 name Boot media ethernet
```

6. **Exit the vlan database by typing:**

```
Console(config-vlan)#end
```

7. **Add the server blade port `SNP0` to the management VLAN (VLAN 2), the data VLAN (VLAN 3), and to the VLAN that you are using for booting (VLAN 4).**

   To do this, type the following commands:

```
Console#configure
Console(config)#interface ethernet SNP0
Console(config-if)#switchport allowed vlan add 2 tagged
Console(config-if)#switchport allowed vlan add 3 tagged
Console(config-if)#switchport allowed vlan add 4
Console(config-if)#switchport native vlan 4
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#exit
Console(config)#
```

   The meaning of this sequence is as follows:

- The `interface ethernet SNP0` command specifies the blade port you are configuring (in the example, the interface is blade port `SNP0`).
- The `switchport allowed vlan add 2 tagged` command makes this blade port a member of VLAN 2 (the management network), and allows it to pass tagged traffic to the management network.
- The `switchport allowed vlan add 3 tagged` command makes the port a member of VLAN 3 (the new data network) and allows it to pass tagged traffic to the data network.
- The `switchport allowed vlan add 4` command makes the port a member of VLAN 4. It causes the port to accept untagged packets and to tag them as members of VLAN 4. By doing this, you are providing a path for untagged traffic generated by the blade (during booting) to reach the Network Install Server. In the next command, you will make this the native VLAN, in other words, the VLAN onto which all untagged frames are forwarded.
- The `switchport native vlan 4` command makes the port put any untagged frames it receives onto VLAN 4. (OBP and Jumpstart involve server blades in sending untagged frames.)
- The `switchport allowed vlan remove 1` command removes the port from VLAN 1 (the default VLAN on the switch for all the server blade ports and uplink ports).

Repeat Step 7 for all the remaining server blade ports (`SNP1` through `SNP15`). All of these ports need to be included in both the management network and the data network.

To inspect the port you have configured, type:

```
Console#show interfaces switchport ethernet SNP0
Information of SNP0
 Broadcast threshold: Enabled, 256 packets/second
 Lacp status: Disabled
 VLAN membership mode: Hybrid
 Ingress rule: Disabled
 Acceptable frame type: All frames
 Native VLAN: 4
 Priority for untagged traffic: 0
 Gvrp status: Disabled
 Allowed Vlan:    2(t), 3(t), 4(u)
 Forbidden Vlan:
Console#
```

8. **If you intend to combine any of the data uplink ports into aggreated links, do this now.**

   Follow the instructions in Section A.11, “Using Aggregated Links for Resilience and Performance” on page A-15.

9. **Add any data uplink ports (that are not combined into aggreated link) to the data VLAN (that is, VLAN 3) and to the boot VLAN (VLAN 4) by typing the following commands:**

```
Console#configure
Console(config)#interface ethernet NETP0
Console(config-if)#switchport allowed vlan add 3 tagged
Console(config-if)#switchport allowed vlan add 4
Console(config-if)#switchport native vlan 4
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport mode trunk
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#no switchport gvrp
Console(config-if)#switchport forbidden vlan add 2
Console(config-if)#end
Console(config)#
```

- The `interface ethernet NETP0` command specifies the uplink port you are configuring.
- The `switchport allowed vlan add 3 tagged` command adds this uplink port to the data network (VLAN 3).
- The `switchport allowed vlan add 4` command adds this uplink port to an untagged VLAN you are using for blade booting (VLAN 4). In the next command, you will make this the native VLAN (in other words, the VLAN onto which any untagged frames are forwarded by this data port).
- The `switchport native vlan 4` command makes the external data port put any untagged frames it receives onto VLAN 4. (The effect of this command is temporary; the subsequent commands will prevent the port from accepting untagged frames. The reason you need to type it is that the switch requires a native VLAN to be available until the `switchport mode trunk` command has been executed.)
- The `switchport allowed vlan remove 1` command removes this uplink port from VLAN 1 (the default VLAN). This VLAN can only be removed at this point, (that is, after VLAN 4 - the native, untagged VLAN - has been created).
- The `switchport ingress-filtering` command, the `switchport mode trunk` command, and the `switchport acceptable-frame-types tagged` command cause the port to reject any frames that are not tagged for the particular VLAN or VLANs that it is a member of.
- The `no switchport gvrp` command prevents the port from using GVRP to advertise the VLANs it is a member of (in this case, VLAN 3) to another switch that it is connected to.

- The `switchport forbidden vlan add 2` command prevents the uplink port from being added to vlan 2 in response to a GVRP request from another switch on the network.

To inspect a port that you have configured, type:

```
Console#show interfaces switchport ethernet NETP0
Information of NETP0
 Broadcast threshold: Enabled, 256 packets/second
 Lacp status: Disabled
 VLAN membership mode: Trunk
 Ingress rule: Enabled
 Acceptable frame type: Tagged frames only
 Native VLAN: 4
 Priority for untagged traffic: 0
 Gvrp status: Disabled
 Allowed Vlan:    3(t), 4(t)
 Forbidden Vlan:    2,
Console#
```

10. Add any aggregated link to the data VLAN (VLAN 3) by typing the commands below.

For more information about using aggregated links, see Chapter A.

In the example below, the aggregated link is called port-channel 1. The `interface port-channel 1` command specifies the aggregated link you are about to configure.

```
Console(config)#interface port-channel 1
Console(config-if)#switchport allowed vlan add 3 tagged
Console(config-if)#switchport allowed vlan add 4
Console(config-if)#switchport native vlan 4
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport mode trunk
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#no switchport gvrp
Console(config-if)#switchport forbidden vlan add 2
Console(config-if)#end
Console(config)#
```

11. **Add all uplink ports to VLAN 3 either individually or as aggregated links (see Step 9 and Step 10).**

    For example, if ports NETP1, NETP2, and NETP3 are combined into port-channel 1, and NETP4, and NETP5 are combined into port-channel 2, you will need to add ports NETP0, NETP6, and NETP7 plus port-channel 1 and port-channel 2 to VLAN 3.

12. **Follow the instructions in Section A.12, "Enabling Secure Management of Blades" on page A-16.**

13. **Save the changes you have made to the configuration of the switch in SSC0.**

    To do this, follow the instructions in Section A.9, "Saving Your Switch Settings" on page A-10.

14. **Copy the configuration of the switch in SSC0 on to the switch in SSC1.**

    Follow the instructions in Section A.10, "Copying the Configuration of the First Switch to the Second" on page A-10.

15. **Type `#.` to exit the switch's command-line interface and return to the System Controller.**

16. **From the `sc>` prompt, log into the switch in SSC1 by typing:**

```
sc> console ssc1/swt
```

17. **Type your user name and password.**

18. **Set the IP address, netmask, and default gateway for the switch in SSC1.**

    To do this, follow the instructions in Section A.7, "Setting the IP address, Netmask, and Default Gateway" on page A-7.

19. **Save the changes you have made to the configuration of the switch in SSC1.**

    To do this, follow the instructions in Section A.9, "Saving Your Switch Settings" on page A-10.

20. **Type `#.` to exit the switch command-line interface and return to the `sc>` prompt.**

21. **Follow the instructions in Section 6.4, "Setting up the SPARC Solaris Blades Using IPMP for Network Resiliency (VLAN Tagging)" on page 6-11.**

6.4 Setting up the SPARC Solaris Blades Using IPMP for Network Resiliency (VLAN Tagging)

The switch configuration you performed in the previous section uses tagged VLANs to separate the data and management networks. For IPMP to work with this switch configuration, you need four IP addresses for *each* VLAN that the server blade is a member of. (In other words, you need eight IP addressess, four for the management VLAN and four for the data VLAN.)

This is because the IPMP driver supports tagged VLANs by using a separate pair of logical Ethernet interfaces for each VLAN. These logical interfaces each have to be named manually according to a simple formula:

`ce`(*VLAN id* x 1000) + *instance*

where *VLAN id* is the number of the VLAN (as configured on the switch ports that the server blade is connected to inside the chassis), and *instance* is either 0 or 1 depending on whether the logical interface is associated with the physical interface `ce0` or `ce1`.

The effect of creating these pairs of logical Ethernet interfaces is to ensure that frames for one network go to that network and not to any other. Whenever the IPMP driver has a frame to send to the switch, it tags it for whichever VLAN is destined to receive it and then transmits it using either of the two logical interfaces available for that VLAN. One of the switches then receives the frame (on the port that is dedicated to the particular server blade that sent it). And, assuming that the switch has been configured to accept frames for the VLAN indicated by the tag, it forwards the frame onto that VLAN.

The important point is that the server blade's IPMP driver has transmitted the frame onto a particular VLAN, and has used a redundant virtual connection to that VLAN to do so. Any other VLANs that the server blade is a member of have been prevented from receiving the frame.

6.4.1 Configuring the Server Blade (VLAN Tagging)

This section tells you how to configure IPMP on a server blade so that the two Ethernet interfaces both provide two active logical interfaces (one each to the data VLAN and the management VLAN).

For purposes of illustration the instructions below use sample configuration input from the network scenario described in Section 6.2, "Preparing the Network Environment" on page 6-2. They assume that the server blade configuration for IPMP described in Chapter 5 has already been performed.

TABLE 6-1 summarizes the information you would need to give the IPMP driver on the server blade in Slot 0 of the system chassis illustrated in FIGURE 6-1.

Note – You need to perform the instructions in this section on each server blade that requires a redundant connection to the data network and the management network.

TABLE 6-1 Sample IPMP Configuration for a Server Blade (VLAN Tagging)

IPMP Configuration Variable	Value for Sample Server Bladein Slot 0
Network adapter interfaces	`ce2000` (active) `ce2001` (active) `ce3000` (active) `ce3001` (active)
Interface group names	`medusa_grp0-mgt` `medusa_grp0`
IP address and host name (`ce2000/1`) IP address and host name (`ce3000/1`)	192.168.2.150 (medusa-s0-mgt) 192.168.1.150 (medusa-s0)
IP address and host name (management network) IP address and host name (data network)	192.168.2.150 (medusa-s0-mgt) 192.168.1.150 (medusa-s0)
sec IP address and host name (management network) sec IP address and host name (data network)	192.168.2.166 (medusa-s0-mgt-sec) 192.168.1.166 (medusa-s0-sec)
Test IP address and host name (`ce2000`) Test IP address and host name (`ce2001`)	192.168.2.100 medusa-s0-0 192.168.2.116 medusa-s0-1
Test IP address and host name (`ce3000`) Test IP address and host name (`ce3001`)	192.168.1.100 medusa-s0-0 192.168.1.116 medusa-s0-1
Netmask	255.255.255.0
Is the server blade to perform network routing?	No

1. **Perform a preliminary setup of Solaris by following the instructions in Chapter 3.**

   When you have done this, type #. to return from a server blade console to the `sc>` prompt.

2. **Log into the console of the server blade whose interfaces you want to configure.**

   Type the following at the `sc>` prompt:

```
sc> console sn
```

   where *n* is the number of the slot containing the server blade you want to log into.

3. **Edit the `/etc/hosts` file on the server blade itself to add the IP addresses for the management interfaces.**

   For a blade using the sample addresses in TABLE 6-1, you would need to add the last two lines of the following file:

```
#
# /etc/hosts on the server blade in system chassis Medusa, slot 0
#
127.0.0.1       localhost          loghost

192.168.1.150   medusa-s0          # Data Address
192.168.1.166   medusa-s0-sec      # Secondary Data Address
192.168.1.100   medusa-s0-0        # Test Address for ce0
192.168.1.116   medusa-s0-1        # Test Address for ce1

192.168.2.150   medusa-s0-mgt       # Data Address
192.168.2.166   medusa-s0-mgt-sec   # Secondary Data Address
192.168.2.100   medusa-s0-mgt-0     # Test Address for ce0
192.168.2.116   medusa-s0-mgt-1     # Test Address for ce1
```

4. **Set the netmask in the server blade's `/etc/netmasks` file.**

   For a blade using the sample addresses in TABLE 6-1, you would need to add the following line:

```
192.168.1.0     255.255.255.0
192.168.2.0     255.255.255.0
```

5. **Disable routing, because the server blade is not being used to perform routing.**

   Type:

```
# touch /etc/notrouter
# ndd -set /dev/ip ip_forwarding 0
```

6. **Unplumb the existing network interfaces by typing:**

```
# ifconfig ce0 unplumb
# ifconfig ce1 unplumb
```

   If either or both these interfaces have not been previously configured, you may receive the following error message:

```
ifconfig: unplumb: SIOCGLIFFLAGS: ce1: no such interface
```

7. **Create the new interfaces by typing:**

```
# ifconfig ce2000 plumb
# ifconfig ce2001 plumb
# ifconfig ce3000 plumb
# ifconfig ce3001 plumb
```

8. **Create IPMP failover groups containing the new interfaces:**

```
# ifconfig ce2000 group medusa_grp0-mgt
# ifconfig ce2001 group medusa_grp0-mgt
# ifconfig ce3000 group medusa_grp0
# ifconfig ce3001 group medusa_grp0
```

   When you execute these commands, you might see the following type of `syslog` message:

```
Sep  3 00:49:58 medusa-s0 in.mpathd[298]: Failures cannot be
detected on ce0 as no IFF_NOFAILOVER address is available
```

   This simply warns you that failures cannot be detected until test addresses have been established on the interfaces.

9. **Create an address on each new interface for data transmission and mark it to `failover` if an interface failure is detected.**

```
# ifconfig ce2000 medusa-s0-mgt netmask + broadcast + failover up
Setting netmask of ce2000 to 255.255.255.0
#
# ifconfig ce2001 medusa-s0-mgt-sec netmask + broadcast + failover up
Setting netmask of ce2001 to 255.255.255.0
#
# ifconfig ce3000 medusa-s0 netmask + broadcast + failover up
Setting netmask of ce3000 to 255.255.255.0
#
# ifconfig ce3001 medusa-s0-sec netmask + broadcast + failover up
Setting netmask of ce3001 to 255.255.255.0
```

10. **Configure a test address on each network interface.**

    These will be used by `mpathd` to detect interface failures. To prevent them from being used by host applications for data communication use the word `deprecated` on the command line (see below).

    Also, you need to use the `-failover` flag. This causes `in.mpathd` to use the address as a test address (in other words, an address that cannot pass to the other interface and therefore does not fail over):

```
# ifconfig ce2000 addif medusa-s0-mgt-0 netmask + broadcast + -failover
deprecated up
Created new logical interface ce2000:1
Setting netmask of ce2000:1 to 255.255.255.0
# ifconfig ce2001 addif medusa-s0-mgt-1 netmask + broadcast + -failover
deprecated up
Created new logical interface ce2001:1
Setting netmask of ce2001:1 to 255.255.255.0
# ifconfig ce3000 addif medusa-s0-0 netmask + broadcast + -failover deprecated up
Created new logical interface ce3000:1
Setting netmask of ce3000:1 to 255.255.255.0
# ifconfig ce3001 addif medusa-s0-1 netmask + broadcast + -failover deprecated up
Created new logical interface ce3001:1
Setting netmask of ce3001:1 to 255.255.255.0
```

11. **To enable the new interface configuration to survive a reboot, create files called** `hostname.ce2000`, `hostname.ce2001`, `hostname.ce3000`, **and** `hostname.ce3001` **in the** `/etc` **directory.**

A sample file for `hostname.ce2000` is as follows:

```
medusa-s0-mgt netmask + broadcast + \
group medusa_grp0-mgt failover up \
addif medusa-s0-mgt-0 netmask + broadcast + \
deprecated -failover up
```

A sample file for `hostname.ce2001` is as follows:

```
medusa-s0-mgt-sec netmask + broadcast + \
group medusa_grp0-mgt failover up \
addif medusa-s0-mgt-1 netmask + broadcast + \
deprecated -failover up
```

A sample file for `hostname.ce3000` is as follows:

```
medusa-s0 netmask + broadcast + \
group medusa_grp0 failover up \
addif medusa-s0-0 netmask + broadcast + \
deprecated -failover up
```

A sample file for `hostname.ce3001` is as follows:

```
medusa-s0-sec netmask + broadcast + \
group medusa_grp0 failover up \
addif medusa-s0-1 netmask + broadcast + \
deprecated -failover up
```

12. **Inspect the configuration of the two network adapters by typing:**

```
# ifconfig -a
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
        inet 127.0.0.1 netmask ff000000
ce2000: flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,IPv4>
mtu 1500 index 3
        inet 192.168.2.150 netmask ffffff00 broadcast 192.168.2.255
        groupname medusa_grp0-mgt
        ether 0:3:ba:19:26:3
ce2000:1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER> mtu 1496 index 3
        inet 192.168.2.100 netmask ffffff00 broadcast 192.168.2.255
ce2001: flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,IPv4>
mtu 1500 index 4
        inet 192.168.2.166 netmask ffffff00 broadcast 192.168.2.255
        groupname medusa_grp0-mgt
        ether 0:3:ba:19:26:4
ce2001:1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER> mtu 1496 index 4
        inet 192.168.2.116 netmask ffffff00 broadcast 192.168.2.255
ce3000: flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,IPv4>
mtu 1500 index 5
        inet 192.168.1.150 netmask ffffff00 broadcast 192.168.1.255
        groupname medusa_grp0
        ether 0:3:ba:19:26:3
ce3000:1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER> mtu 1496 index 5
        inet 192.168.1.100 netmask ffffff00 broadcast 192.168.1.255
ce3001: flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,IPv4>
mtu 1500 index 6
        inet 192.168.1.166 netmask ffffff00 broadcast 192.168.1.255
        groupname medusa_grp0
        ether 0:3:ba:19:26:4
ce3001:1: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,NOFAILOVER> mtu 1496 index 6
        inet 192.168.1.116 netmask ffffff00 broadcast 192.168.1.255
```

The output above shows that eight addresses have been defined (the sample addresses from TABLE 6-1). The four IPMP test addresses are marked NOFAILOVER. This means that they will not be transferred to the surviving interface in the event of a failure.

13. **Test IPMP by temporarily removing one SSC from the chassis.**

This will cause the following error messages to be displayed on the console:

```
Sep  4 20:12:16 medusa-s0 in.mpathd[31]: NIC failure detected on
ce3001 of group medusa_grp0
Sep  4 20:12:16 medusa-s0 in.mpathd[31]: Successfully failed over
from NIC ce3001 to NIC ce3000
```

Note – It takes approximately 10 seconds for the IPMP daemon to detect and recover from a network failure with the default configuration. The configuration of the IPMP daemon is defined in the `/etc/default/mpathd` file.

CHAPTER 7

Sample Switch Configurations for Multiple Tenants

This chapter contains the following sections:

Note – If you reset the switch while you are performing the instructions in this section, you must save the configuration first. If you do not, you will lose all of your changes. To save the configuration, follow the instructions in Section A.9, "Saving Your Switch Settings" on page A-10.

7.1 Introduction

This chapter is intended mainly for Internet Service Providers (ISPs) who need to:

- Allocate server blades to different customers
- Enable those customers to manage their own blades
- Prevent any customers from receiving data from another customer's network
- Prevent any customers from accessing the console on another customer's blades
- Prevent any customers from accessing the console on either of the integrated switches

The chapter provides two sample switch configurations that illustrate the use of VLANs to allocate server blades to different customers. We shall refer to an ISP's customers in the rest of the chapter as the "tenants" of particular server blades.

The switch configurations assume that only the ISP has login and password access to the SC and switch command-line interfaces. Customers of the ISP can `ping` the NETMGT port on the switch, because they have their own management networks that include the NETMGT port. But unless you give them login and password access to the switch, they cannot access it. The VLAN configuration means that none of the customers has access to the SC's network port via telnet.

Although this chapter is intended mainly for ISPs, it might also be useful to network administrators with a general interest in ways of using VLANs to control network traffic on the Sun Fire B1600 blade system chassis.

This chapter does not provide instructions on configuring IPMP on the blades. For guidance with the configuration of the IPMP interfaces for complex VLAN setups, see Chapter 6.

Note – The instructions in this chapter are concerned with the use of VLANs. They assume that your wider network uses tagged VLANs. This means that the configurations in this chapter do not support Solaris installation across the network (because this requires the VLANs on the switch to handle untagged traffic). The instructions in this chapter are provided only to illustrate the use of the VLAN facilities on the switch.

For information about configuring the switch to remove VLAN tagging from frames that it sends onto the network (but to add a VLAN tag to untagged frames that it receives from the network), see Section 7.2.4, "Allocating Data Network Ports to Each Tenant" on page 7-10 and Section 7.3.4, "Sharing the Data Network Ports Between Tenants" on page 7-16.

7.2 Scenario A: Three Different Tenants With Their Own Blades and Data Ports

In this scenario, an Internet Service Provider (ISP) is assumed to own the blade system chassis and to be in overall responsibility for managing it. The ISP therefore has sole access to the switch's command-line interface on NETMGT.

The scenario also assumes three tenants: Tenant 1, Tenant 2, and Tenant 3. Each tenant has a single data VLAN assigned to him or her exclusively. This data VLAN includes a number of server blades (that is to say, a number of the switch's server blade down-link ports) and a number of external data ports.

The tenants also have a management VLAN each that gives them secure access to their own blades.

The switch configuration is summarised in TABLE 7-1.

TABLE 7-1 Scenario A: Three Tenants With Their Own Server Blades and Data Ports

Network Administrator	Management Port	Server Blade Ports	Uplink Ports	Data VLAN id	Management VLAN id
Internet Service Provider	NETMGT	None	None	None	2
Tenant 1	NETMGT	SNP0, SNP1, SNP2	NETP0, NETP1	11	21
Tenant 2	NETMGT	SNP3, SNP4, SNP5, SNP6, SNP7, SNP8, SNP9	NETP2, NETP3, NETP4	12	22
Tenant 3	NETMGT	SNP10, SNP11, SNP12, SNP13, SNP14, SNP15	NETP5, NETP6, NETP7	13	23

The rest of this section tells you how to create the configuration described in TABLE 7-1. It is divided into the following sub-sections:

- Section 7.2.1, "Creating and Naming All the VLANs" on page 7-6
- Section 7.2.2, "Allocating the Management Port (NETMGT) to Each Tenant" on page 7-7

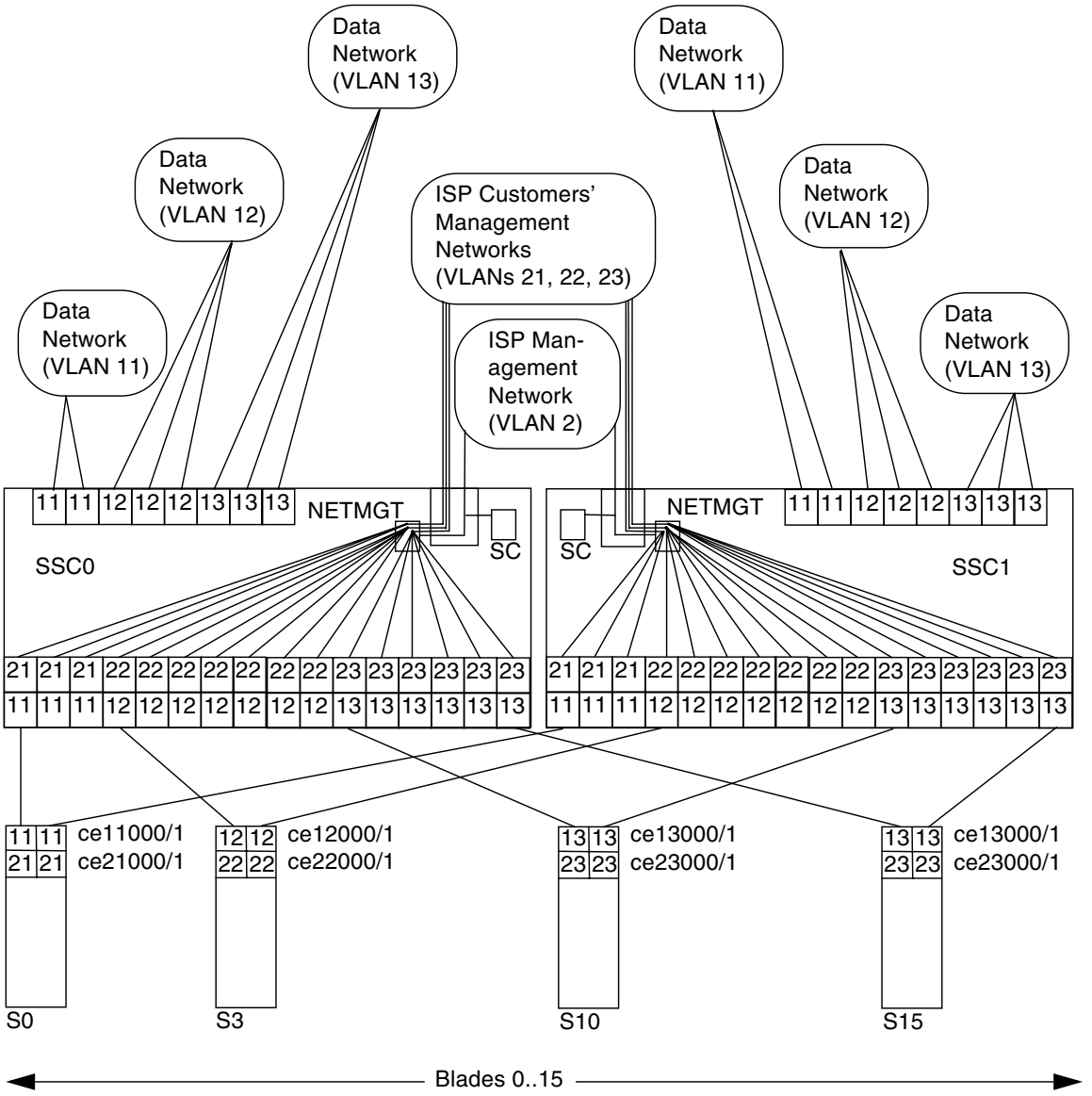

FIGURE 7-1 Scenario A: The Tenants' Data and Management VLANs and the ISP's Management VLAN

FIGURE 7-1 is a diagram expressing the same information as TABLE 7-1. At the center is the ISP's management VLAN, VLAN 2. This VLAN is exclusive to the ISP's network administrator. It includes the NETMGT port on the switch (thereby allowing the ISP network administrator to configure the entire switch via a telnet or web connection). It also includes the System Controller (thereby allowing the ISP to configure the entire chassis and to access the consoles of all server blades and both switches from the `sc>` prompt). Note, however, that the System Controller's membership of the VLAN is configured from the `sc>` prompt (specifically, by the `setupsc` command): it is not part of the switch configuration process.

Note – In this scenario it is assumed that the ISP network administrator does not give any of his or her customers password access to the command-line interface for either the System Controller or the switch. It is the responsibility of the network administrator to control access to the System Controller and switch interfaces.

Above VLAN 2 in the diagram are the three management VLANs for the ISP's individual customers. Each of these customers has access on a dedicated management VLAN to his or her own server blades. So, for example, Tenant 1 (whose management VLAN is number 21) can telnet into the server blades in slots 0, 1, and 2. Tenant 2 (management VLAN 22) can telnet into the server blades in slots 3 through 9. And Tenant 3 (management VLAN23) can telnet into slots 10 through 15.

At the bottom of the diagram, the first of each customer's server blades is shown. These blades each require two logical interfaces to their data network and two logical interfaces to their management network. These logical interfaces have to be provided by IPMP (see Chapter 6). The diagram shows the interface numbering required for the IPMP configuration. For example, Tenant 1's server blades contain two logical interfaces for VLAN 11 (data network) and two logical interfaces for VLAN 21 (management network). Following the formula provided in Chapter 6, the interface numbering for each of Tenant 1's blades is `ce11000` and `ce21000` (for the connection on `ce0` to the switch in SSC0) and `ce11001` and `ce21001` (for the connection on `ce1` to the switch in SSC1).

Finally the ISP's customers in this scenario each have dedicated network uplink ports. Tenant 1 has NETP0 and NETP1, Tenant 2 has NETP2, NETP3, and NETP4, and Tenant 3 has NETP5, NETP6, and NETP7. These ports are made exclusive to particular tenants by being included in the data network VLANs that contain the server blades belonging to each tenant. So, for example, Tenant 3's data network VLAN (13) includes server blade ports SNP10 through SNP15 plus the uplink ports NETP5, NETP6, and NETP7.

Note – If the uplinks belonging to the different tenants connect to the same external switch, then the Spanning Tree protocol will break some of the connections. We recommend you use a different external switch for each tenant. Alternatively you can turn off the Spanning Tree protocol (see Section 7.2.5, "Turning Off Spanning Tree" on page 7-11).

7.2.1 Creating and Naming All the VLANs

1. **To log into the switch in SSC0, type:**

```
sc> console ssc0/swt
```

2. **When prompted for a user name, type** `admin`**.**

   Type `admin` again as the password.

3. **Make sure that the switch is using the factory default configuration.**

   For information about how to do this, see Section A.5, "Restoring the Switch to its Factory Default State" on page A-5.

4. **If you have returned to the factory default configuration, or if you have not yet set your own password, do so now.**

   For information about how to do this, see Section 2.2, "Logging into the Switch as the Default User and Setting the Passwords" on page 2-4.

5. **Create and give names to the tenants' data VLANs.**

   To do this, type:

```
Console#configure
Console(config)#vlan database
Console(config-vlan)#vlan 11 name tenant1 media ethernet
Console(config-vlan)#vlan 12 name tenant2 media ethernet
Console(config-vlan)#vlan 13 name tenant3 media ethernet
Console(config-vlan)#end
```

6. **Create and give names to the tenants' management VLANs.**

   Type:

```
Console#configure
Console(config)#vlan database
Console(config-vlan)#vlan 21 name tenant1_managment media
ethernet
Console(config-vlan)#vlan 22 name tenant2_managment media
ethernet
Console(config-vlan)#vlan 23 name tenant3_managment media
ethernet
Console(config-vlan)#end
```

7.2.2 Allocating the Management Port (NETMGT) to Each Tenant

1. **Configure the switch's management port (NETMGT) to enable it to receive and transmit frames from and to the ISP's management VLAN (2) and all of the tenants' management VLANs (21, 22, 23).**

   The ISP uses the default management VLAN, VLAN 2.

   Type:

```
Console#configure
Console(config)#interface ethernet NETMGT
Console(config-if)#switchport allowed vlan add 21 tagged
Console(config-if)#switchport allowed vlan add 22 tagged
Console(config-if)#switchport allowed vlan add 23 tagged
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport mode trunk
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#no switchport gvrp
Console(config-if)#end
```

   The meaning of this sequence is as follows:

   - The `interface ethernet NETMGT` command specifies that you are configuring the management port.
   - The `switchport allowed vlan add 21` command adds NETMGT to the management VLAN (21) for Tenant 1 and allows it to pass tagged frames to that VLAN.

- The `switchport allowed vlan add 22` command adds NETMGT to the management VLAN (22) for Tenant 2 and allows it to pass tagged frames to that VLAN.
- The `switchport allowed vlan add 23` command adds NETMGT to the management VLAN (23) for Tenant 3 and allows it to pass tagged frames to that VLAN.
- The `switchport ingress-filtering` command, the `switchport mode trunk` command, and the `switchport acceptable-frame-types tagged` command cause NETMGT to accept and transmit only frames tagged for the particular VLANs it is a member of (VLANs 21, 22, 23, and the default management VLAN, VLAN 2).
- The `no switchport gvrp` command prevents NETMGT from using GVRP to advertise the VLANs it is a member of to another switch.

2. **Make sure the switch's IP packet filter is configured to permit traffic to pass from the server blades to the management network.**

   For information on how to do this, see Section A.12, "Enabling Secure Management of Blades" on page A-16.

7.2.3 Allocating Server Blade Ports to Each Tenant

1. **For Tenant 1, configure the server blade ports so that they will transmit and receive frames tagged for VLANs 11 and 21 only.**

   Type:

```
Console#configure
Console(config)#interface ethernet SNP0
Console(config-if)#switchport allowed vlan add 11 tagged
Console(config-if)#switchport allowed vlan add 21
Console(config-if)#switchport native vlan 21
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport mode trunk
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#end
```

   Repeat these commands for the other two server blade ports (SNP1 and SNP2) belonging to Tenant 1.

2. **For Tenant 2, configure the server blade ports so that they will transmit and receive frames tagged for VLANs 12 and 22 only.**

   Type:

```
Console#configure
Console(config)#interface ethernet SNP3
Console(config-if)#switchport allowed vlan add 12 tagged
Console(config-if)#switchport allowed vlan add 22
Console(config-if)#switchport native vlan 22
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport mode trunk
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#end
```

   Repeat these commands for the other server blade ports (SNP4 through SNP9) belonging to Tenant 2.

3. **For Tenant 3, configure the server blade ports so that they will transmit and receive frames tagged for VLANs 13 and 23 only.**

   Type:

```
Console#configure
Console(config)#interface ethernet SNP10
Console(config-if)#switchport allowed vlan add 13 tagged
Console(config-if)#switchport allowed vlan add 23
Console(config-if)#switchport native vlan 23
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport mode trunk
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#end
```

   Repeat these commands for the other server blade ports (SNP11 through SNP15) belonging to Tenant 3.

7.2.4 Allocating Data Network Ports to Each Tenant

Note – The network devices you connect the Sun Fire B1600 Blade System Chassis to must be VLAN-aware. For this reason the instructions include the `switchport mode trunk` command which causes a network port to transmit and receive only frames that are tagged for the particular VLANs (or in this case, the particular VLAN) it is a member of.

---

1. **Configure the network ports for Tenant 1 so that they will receive and transmit frames tagged for VLAN 11 only.**

   For NETP0, type:

```
Console#configure
Console(config)#interface ethernet NETP0
Console(config-if)#switchport allowed vlan add 11 \
Console(config-if)#switchport native vlan 11
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#switchport mode trunk
Console(config-if)#no switchport gvrp
Console(config-if)#end
```

   Repeat these commands for NETP1.

2. **Configure the network ports for Tenant 2 so that they will receive and transmit frames tagged for VLAN 12 only.**

   For NETP2, type:

```
Console#configure
Console(config)#interface ethernet NETP2
Console(config-if)#switchport allowed vlan add 12
Console(config-if)#switchport native vlan 12
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#switchport mode trunk
Console(config-if)#no switchport gvrp
Console(config-if)#end
```

   Repeat these commands for NETP3 and NETP4.

3. **Configure the network ports for Tenant 3 so that they will receive and transmit frames tagged for VLAN 13 only.**

   For NETP5, type:

```
Console#configure
Console(config)#interface ethernet NETP5
Console(config-if)#switchport allowed vlan add 13
Console(config-if)#switchport native vlan 13
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#switchport mode trunk
Console(config-if)#no switchport gvrp
Console(config-if)#end
```

   Repeat these commands for NETP5, NETP6, and NETP7.

7.2.5 Turning Off Spanning Tree

If the uplinks belonging to the different tenants connect to the same external switch, then the Spanning Tree protocol will break some of the connections. We recommend you use a different external switch for each tenant. Alternatively you can turn off the Spanning Tree protocol. To turn off Spanning Tree, type:

```
Console#configure
Console(config)#no spanning-tree
Console(config)#end
```

7.2.6 Saving the Switch Settings and Copying the Configuration to the Second Switch

1. **Save the switch settings.**

   To do this, follow the instructions in Chapter A.

2. **Copy the switch configuration onto the second switch.**

   To do this, follow the instructions in Chapter A.

7.3 Scenario B: Two Tenants With Eight Blades Each and Four Shared Data Ports

In this scenario, there is an Internet Service Provider (ISP) who is assumed to own the blade system chassis and to be in overall responsibility for managing it. There are also two tenants, Tenant 1 and Tenant 2. Both tenants have a data VLAN assigned to them, and the VLAN includes eight server blades (that is to say, eight of the switch's server blade ports) plus four of the switch's external data ports. In other words, the two tenants share four of the external data ports (neither has exclusive use of them).

The configuration of the switch for this scenario is summarised in TABLE 7-2.

TABLE 7-2 Scenario B: Two Tenants With Eight Server Blades and Eight Data Ports Each

Network Administrator	Management Port	Server Blade Ports	External Data Ports	Data VLAN id	Management VLAN id
Internet Service Provider	NETMGT	None	None	None	2
Tenant 1	NETMGT	SNP0, SNP1, SNP2, SNP3, SNP4, SNP5, SNP6, SNP7	NETP0 through NETP3	11	21
Tenant 2	NETMGT	SNP8, SNP9, SNP10, SNP11, SNP12, SNP13, SNP14, SNP15	NETP0 through NETP3	12	22

The rest of this section tells you how to create the configuration described in TABLE 7-2. It is divided into the following sub-sections:

- Section 7.3.1, "Creating and Naming All the VLANs" on page 7-14
- Section 7.3.2, "Allocating the Management Port (NETMGT) to Each Tenant" on page 7-14
- Section 7.3.3, "Allocating Server Blade Ports to Each Tenant" on page 7-15
- Section 7.3.4, "Sharing the Data Network Ports Between Tenants" on page 7-16

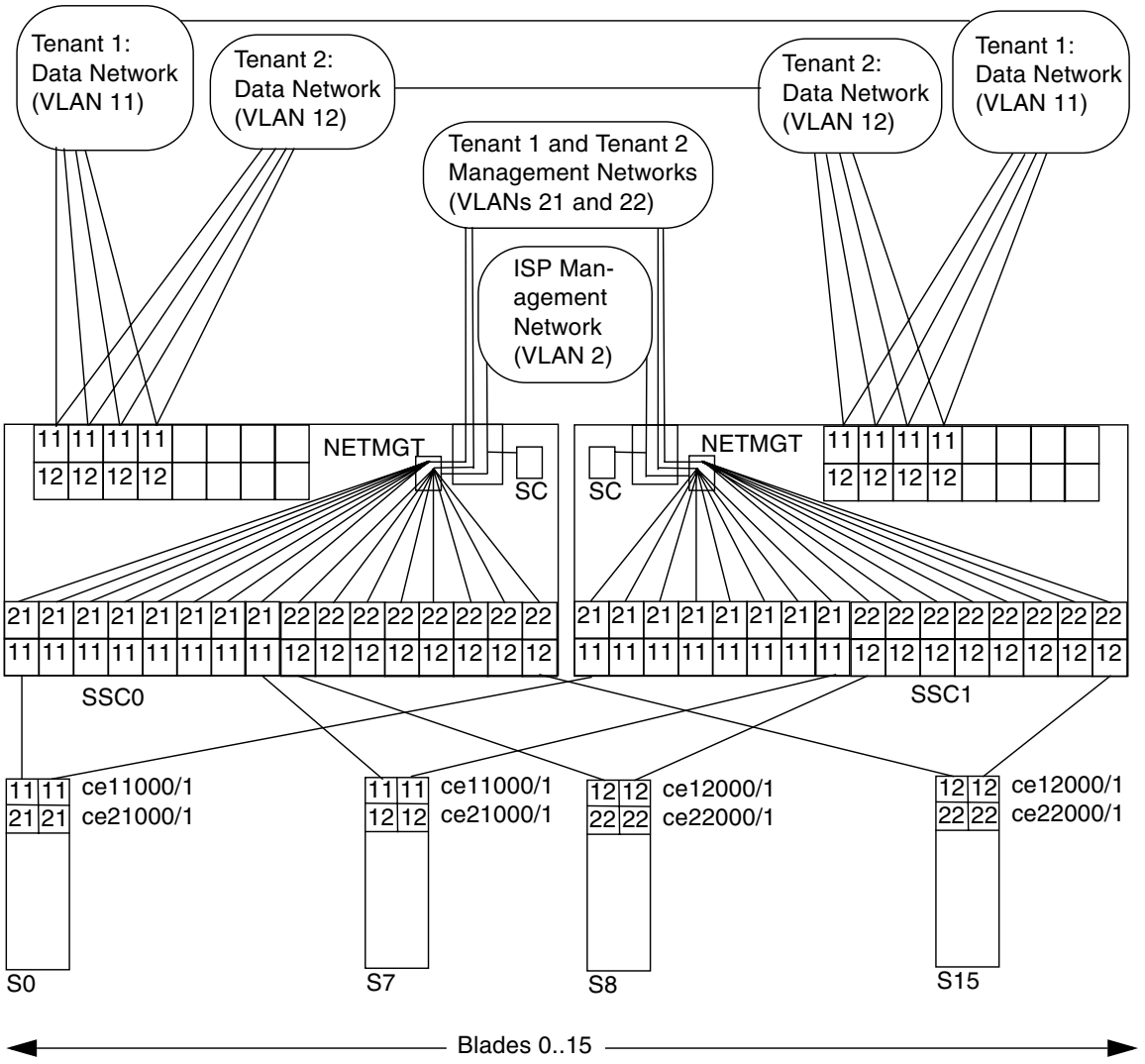

FIGURE 7-2 Scenario B: Two Tenants' Data and Management VLANs With Shared Uplink Ports

FIGURE 7-2 is a diagram expressing the same information as TABLE 7-2. In this scenario the principles are the same as in scenario A except that all of the network uplink ports are shared by the tenants of the server blades. In other words, both of the tenants' data VLANs (VLAN 11 for Tenant 1, and VLAN 12 for Tenant 2) include the uplink ports NETP0 through NETP3. This does not result in the tenants receiving data from each other's server blades, because any frames that leave ports NETP0 through NETP3 will be tagged for either VLAN 11 (Tenant 1) or VLAN 12 (Tenant 2).

7.3.1 Creating and Naming All the VLANs

1. **Create and give names to the tenants' data VLANs.**

   To do this, type:

```
Console#configure
Console(config)#vlan database
Console(config-vlan)#vlan 11 name tenant1 media ethernet
Console(config-vlan)#vlan 12 name tenant2 media ethernet
```

2. **Create and give names to the tenants' management VLANs.**

   Type:

```
Console#configure
Console(config)#vlan database
Console(config-vlan)#vlan 21 name tenant1_managment media ethernet
Console(config-vlan)#vlan 22 name tenant2_managment media ethernet
Console(config-vlan)#end
```

7.3.2 Allocating the Management Port (NETMGT) to Each Tenant

1. **Configure the switch's management port (NETMGT) to enable it to receive and transmit frames from and to the ISP's management VLAN (2) and both of the tenants' management VLANs (21 and 22).**

   Type:

```
Console#config
Console(config)#interface ethernet NETMGT
Console(config-if)#switchport allowed vlan add 21 tagged
Console(config-if)#switchport allowed vlan add 22 tagged
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#switchport mode trunk
Console(config-if)#no switchport gvrp
Console(config-if)#end
```

2. **Make sure the switch's IP packet filter is configured to permit traffic to pass from the server blades to the management network.**

   For information on how to do this, see Section A.12, "Enabling Secure Management of Blades" on page A-16.

7.3.3 Allocating Server Blade Ports to Each Tenant

1. **For Tenant 1, configure the server blade ports so that they will transmit and receive frames tagged for VLANs 11 and 21 only.**

   Type:

```
Console#configure
Console(config)#interface ethernet SNP0
Console(config-if)#switchport allowed vlan add 11 tagged
Console(config-if)#switchport allowed vlan add 21
Console(config-if)#switchport native vlan 21
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport mode trunk
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#end
```

   Repeat these commands for the other seven server blade ports (SNP1 through SNP7) belonging to Tenant 1.

2. **For Tenant 2, configure the server blade ports so that they will transmit and receive frames tagged for VLANs 12 and 22 only.**

   Type:

```
Console#configure
Console(config)#interface ethernet SNP8
Console(config-if)#switchport allowed vlan add 12 tagged
Console(config-if)#switchport allowed vlan add 22
Console(config-if)#switchport native vlan 12
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport mode trunk
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#end
```

   Repeat these commands for the other seven server blade ports (SNP9 through SNP15) belonging to Tenant 2.

7.3.4 Sharing the Data Network Ports Between Tenants

Note – The instructions in this section assume that the network devices you connect the Sun Fire B1600 Blade System Chassis to are VLAN-aware. For this reason the instructions include the `switchport mode trunk` command which causes a network port to transmit and receive only frames that are tagged for the particular VLANs it is a member of.

---

1. **Configure the network ports so that they will receive and transmit frames tagged for VLAN 11 and VLAN 12.**

   For NETP0, type:

```
Console#configure
Console(config)#interface ethernet NETP0
Console(config-if)#switchport allowed vlan add 11 tagged
Console(config-if)#switchport allowed vlan add 12
Console(config-if)#switchport native vlan 12
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#switchport acceptable-frame-types tagged
Console(config-if)#switchport mode trunk
Console(config-if)#no switchport gvrp
Console(config-if)#end
```

2. Repeat these commands for NETP1 through NETP3.

3. **Save the switch settings.**

   To do this, follow the instructions in Chapter A.

4. **Copy the switch configuration onto the second switch.**

   To do this, follow the instructions in Chapter A.

CHAPTER 8

Separating Blades Using VLANs Hidden From the External Network

This chapter is for customers whose network infrastructure does not fully support VLANs or for customers who want to minimize the wider network administration tasks associated with integrating a blade system chassis into their existing environment.

It contains the following sections:

Note – If you reset the switch while you are performing the instructions in this section, you must save the configuration first. If you do not, you will lose all of your changes. To save the configuration, follow the instructions in Section A.9, "Saving Your Switch Settings" on page A-10.

8.1 Introduction

This chapter describes two methods of configuring VLANs inside the chassis to provide network separation between server blades without the VLANs being visible to external devices on the network.

8.2 Using Dedicated VLANs for Each Blade

The configuration described in this section includes one dedicated VLAN for each blade. This VLAN is for traffic originating on the blade.

It also includes a VLAN containing all of the ports. This VLAN is used for traffic originating on the external network.

The configuration is illustrated in FIGURE 8-1. The dedicated VLANs for each blade are numbered from 10 (for the blade in slot 0) to 25 (for the blade in slot 15). Because a second SSC is installed, the ce1 interface on each blade is also used. All of the uplink ports are included in a shared VLAN (numbered 30).

Note – You can use this configuration with one or two switches. If you have a chassis containing two SSCs, you can replicate the configuration of the first switch on the second. This enables you to make use of the second interface on each blade for redundancy.

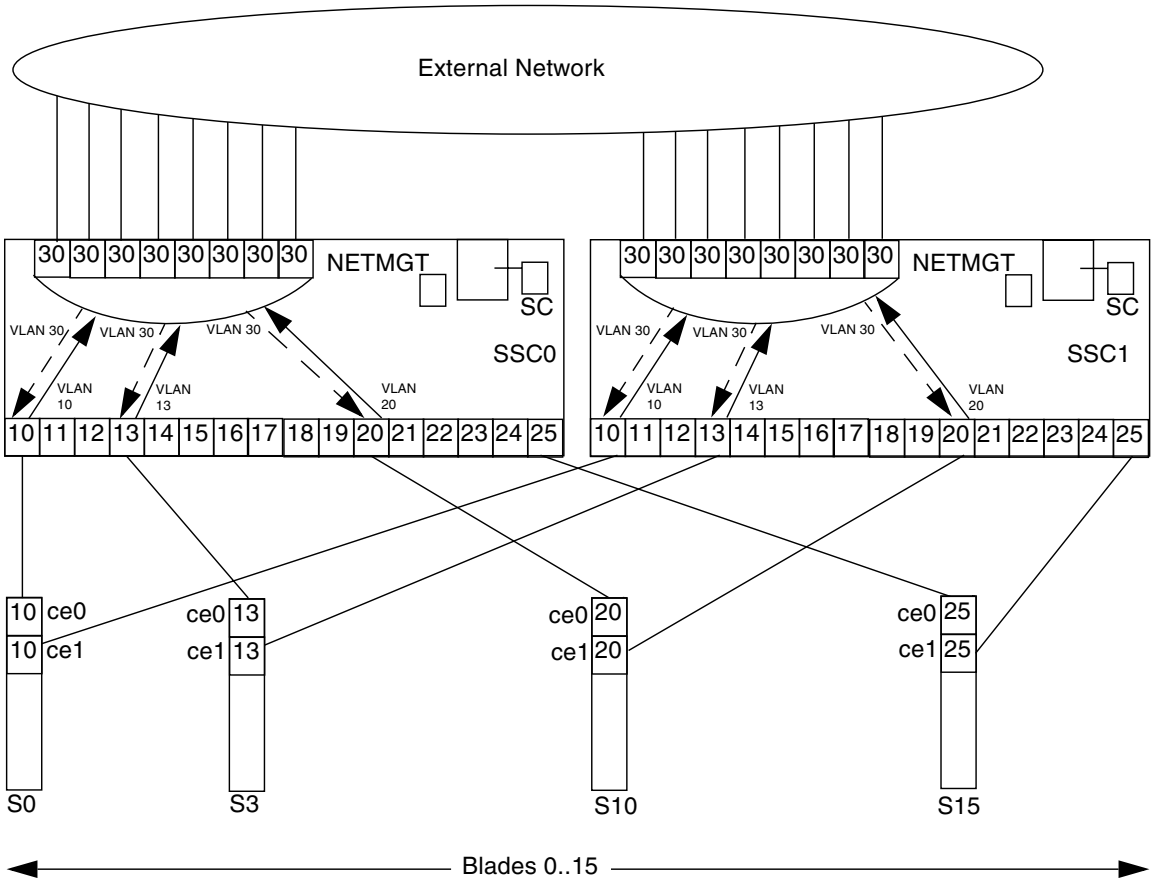

FIGURE 8-1 Sample Configuration Using a Dedicated VLAN for Each Blade and a Single VLAN for all Traffic From the Network

The steps you would need to perform to implement the configuration in FIGURE 8-1 are as follows:

1. **Make a note of the MAC address of each blade interface.**

   At the System Controller's `sc>` prompt, type:

```
sc> showplatform -v
```

   The output includes the MAC address for `ce0` on each server blade (labeled s0 through s15). Note this address and, if you are using the switch in a second SSC, also calculate the MAC address for `ce1`. For each blade this will be the next contiguous hexadecimal number after the number used for `ce0` on the blade. For example, if the MAC address for `ce0` were `00:03:ba:29:ef:32` then for `ce1` it would be `00:03:ba:29:ef:33` ; or, if the MAC address for `ce0` were `00:03:ba:29:ef:4f` then for `ce1` it would be `00:03:ba:29:ef:50`.

   You will need the MAC address information in Step 8.

2. **Log into the switch by typing the following from the System Controller's `sc>` prompt:**

```
sc> console sscn/swt
```

   where *n* is 0 or 1.

3. **Type your user name and password.**

4. **Set up the VLAN database on the switch. Type:**

```
Console#config
Console(config)#vlan database
Console(config-vlan)#vlan vlan ID name name media ethernet
Console(config)#end
```

   where *vlan ID* is the number of the dedicated VLAN for the blade and *name* is its name. To give the blade in slot 0 a VLAN with the ID number 10 and the name `snp0private`, you would type the following:

```
Console#config
Console(config)#vlan database
Console(config-vlan)#vlan 10 name snp0private media ethernet
```

Repeat Step 4 for every blade, making sure you increment the VLAN ID number and the number (if any) in the blade's VLAN name. In the example in FIGURE 8-1, the VLAN for the blade in slot 1 will have the ID number 11 and the name `snp1private`. The VLAN for the blade in slot 2 will have the ID number 12 and the name `snp1private`, and so on.

5. **Set up the shared VLAN that will include the uplink ports. Type:**

```
Console(config-vlan)#vlan vlan ID name name media ethernet
Console(config)#exit
```

where *vlan ID* is the number of the shared VLAN for the uplink ports and *name* is its name. In the example we are using, the VLAN has ID number 30 and VLAN name `shared`:

```
Console(config-vlan)#vlan 30 name shared media ethernet
Console(config-vlan)#exit
```

6. **Now configure each blade port on the switch as a member of both the blade's dedicated VLAN and the shared VLAN.**

The sample commands used in these instructions from now on will use the VLAN names and VLAN ID numbers appropriate to the sample configuration in FIGURE 8-1.

```
Console#config
Console(config)#interface ethernet snp0
Console(config-if)#no switchport gvrp
Console(config-if)#switchport allowed vlan add 10 untagged
Console(config-if)#switchport allowed vlan add 30 untagged
Console(config-if)#switchport native vlan 10
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#exit
```

Repeat Step 6 for each blade port from snp1 (VLAN 11) through snp15 (VLAN 25).

Note – By default the GVRP protocol is disabled on all ports. However, if you have changed this configuration, you need to include the `no switchport gvrp` command as shown above when you configure each port.

---

7. **Configure each uplink port as a member of the shared VLAN (30) and as a member of each blade's dedicated VLAN (10 through 25).**

```
Console#config
Console(config)#interface ethernet netp0
Console(config-if)#no switchport gvrp
Console(config-if)#switchport allowed vlan add 10-25,30 untagged
Console(config-if)#switchport native vlan 30
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#exit
```

Repeat Step 7 for each uplink port that you want to include in the shared VLAN. The sample configuration in FIGURE 8-1 includes all uplink ports netp0 through netp7.

8. **To ensure that the switch learns the MAC addresses for the blades on VLAN 30 (and therefore to prevent it from flooding all ingress traffic to all blades), manually enter the MAC addresses that you noted in Step 1 into the MAC address table for VLAN 30.**

The command is as follows:

```
Console#config
Console(config)#mac-address-table static mac address interface
ethernet snpn vlan 30 permanent
```

where *MAC address* is the 12-digit MAC address of the interface connected to the switch you are configuring (every two digits must be separated by a - character, for example, `00-03-ba-29-ef-32)`, and *n* is the number of the switch port it is connected to.

A sample command for the snp0 port connected to ce0 on the blade in slot 0, and assuming a MAC address of `00:03:ba:29:ef:32` would be:

```
Console#config
Console(config)#mac-address-table static 00-03-ba-29-ef-32
interface ethernet snp0 vlan 30 permanent
```

9. **Save the changes you have made. Type:**

```
Console#copy running-config startup-config
Startup configuration file name [default filename]:filename
Write to FLASH Programming
-Write to FLASH finish
Success

Console#
```

where *default filename* is the current startup configuration file, and *filename* is the name you want to give to a new startup configuration file. If you type [ENTER] instead of specifying a new file name, the running configuration will be written to the current startup configuration file.

8.3 Dedicating Each Uplink Port to a Particular Blade

The configuration described in this section enables you to dedicate each uplink port to a single blade port. The effect is to present the uplinks to the external network as if they are each simply the network port of a server. The switch has only eight uplink ports. Therefore, in this configuration the first eight blades (that is, ports snp0 through snp7) each have a dedicated connection to one of the uplink ports on the switch in SSC0, and the next eight (snp8 through snp15) each have a dedicated connection to one uplink port on the switch in SSC1 (see FIGURE 8-2).

Note – The configuration described in this section requires you to have two SSCs installed. This is because, while there are 16 blade ports in the chassis (snp0 through snp 15), there are only eight uplink ports on each switch (netp0 through netp8). In other words, you need to have two switches if you want 16 dedicated uplink ports.

Note – Note also that, because each uplink is dedicated to a single blade port and each blade port is connected to a single interface on the blade (see FIGURE 8-2), this configuration means that you cannot use the second blade interface for network redundancy.

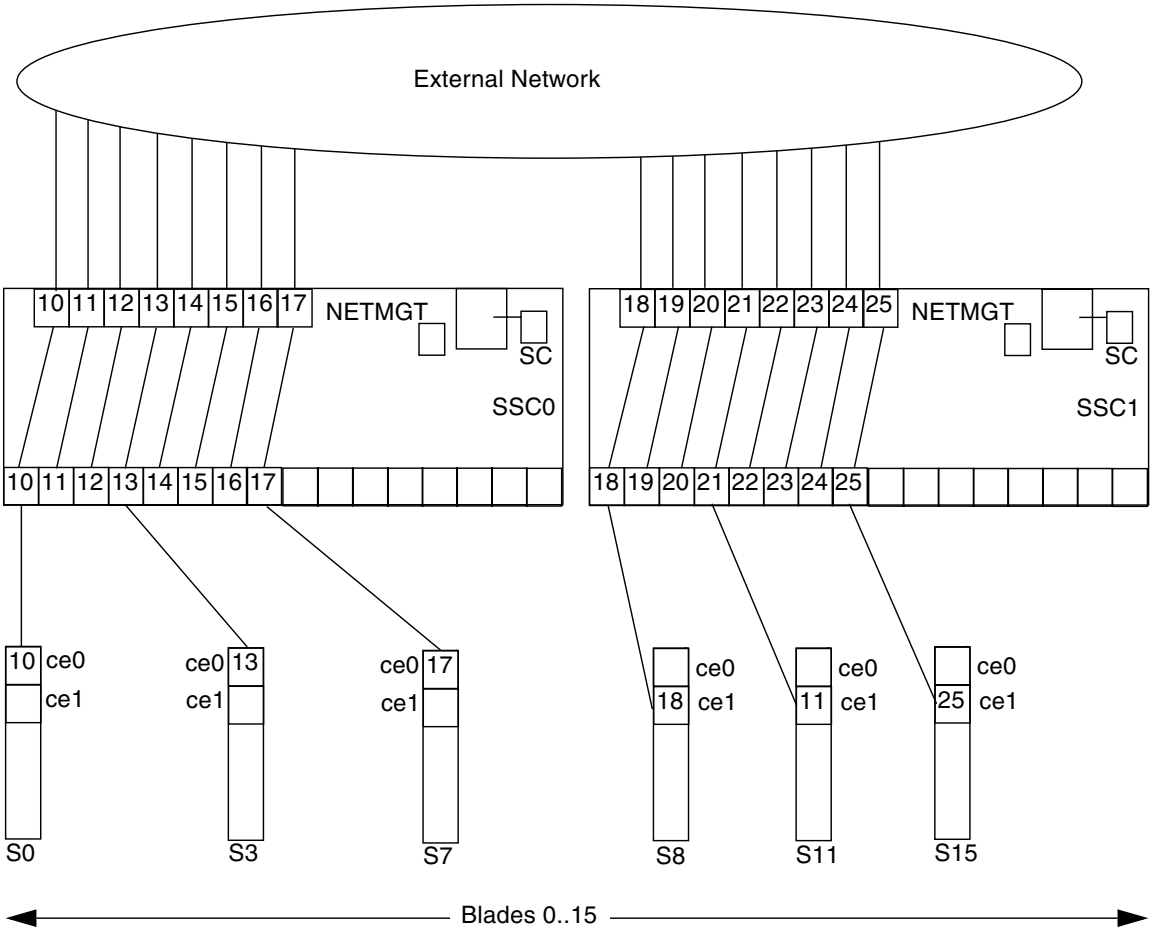

FIGURE 8-2 Sample Configuration With Each Uplink Port Dedicated to a Single Blade

Note – The `ce0` interface on each blade is connected to switch 0; the `ce1` interface is connected to switch 1. Note that, in this configuration, blades s0 through s7 can only use `ce0`, and blades s8 through s15 can only use `ce1`.

The steps you would need to perform to implement the configuration in FIGURE 8-2 are as follows:

1. **Make sure that Spanning Tree is disabled on the external switches that the chassis is connected to.**

   When you have completed the VLAN configuration, you must also disable Spanning Tree on the chassis's switch. Instructions for doing this are provided in Step 8.

2. **Log into the switch 0 by typing the following from the System Controller's `sc>` prompt:**

```
sc> console ssc0/swt
```

3. **Type your user name and password.**

4. **Set up the VLAN database on the switch. Type:**

```
Console#config
Console(config)#vlan database
Console(config-vlan)#vlan vlan ID name name media ethernet
```

   where *vlan ID* is the number of the VLAN for the blade and *name* is the VLAN name. In FIGURE 8-2, the VLAN for blade s0 has ID number 10 and a suitable VLAN name would be `snp0private`. To set up the entire VLAN database, type:

```
Console#config
Console(config)#vlan database
Console(config-vlan)#vlan 10 name snp0private media ethernet
Console(config-vlan)#vlan 11 name snp1private media ethernet
Console(config-vlan)#vlan 12 name snp2private media ethernet
Console(config-vlan)#vlan 13 name snp3private media ethernet
Console(config-vlan)#vlan 14 name snp4private media ethernet
Console(config-vlan)#vlan 15 name snp5private media ethernet
Console(config-vlan)#vlan 16 name snp6private media ethernet
Console(config-vlan)#vlan 17 name snp7private media ethernet
Console(config-vlan)#exit
```

5. **Configure the dedicated VLAN for each blade port from snp0 through snp7:**

```
Console#config
Console(config)#interface ethernet snp0
Console(config-if)#no switchport gvrp
Console(config-if)#switchport allowed vlan add 10 untagged
Console(config-if)#switchport native vlan 10
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#exit
```

Repeat these configuration commands for each blade port (and its VLAN) from snp1 (VLAN 11) through snp7 (VLAN 17).

6. **Include each uplink port in the dedicated VLAN for the blade it is to be dedicated to:**

```
Console#config
Console(config)#interface ethernet netp0
Console(config-if)#switchport allowed vlan add 10 untagged
Console(config-if)#switchport native vlan 10
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#exit
```

Repeat these configuration commands for each uplink port (and its associated blade VLAN) from netp1 (VLAN 11) through netp7 (VLAN 17).

7. **Shut down the switch ports that you are not using (snp8 through snp15):**

```
Console#config
Console(config)#interface ethernet snp8
Console(config-if)#shutdown
Console(config-if)#exit
```

Repeat Step 7 for blade ports snp9 through snp15.

8. **Disable spanning tree on the switch:**

```
Console#config
Console(config)#no spanning-tree
Console(config)#exit
```

9. **Save the changes you have made. Type:**

```
Console#copy running-config startup-config
Startup configuration file name [default filename]:filename
Write to FLASH Programming
-Write to FLASH finish
Success

Console#
```

where *default filename* is the current startup configuration file, and *filename* is the name you want to give to a new startup configuration file. If you type [ENTER] instead of specifying a new file name, the running configuration will be written to the current startup configuration file.

10. **Type #. to return to the System Controller's** `sc>` **prompt.**

11. **Log into switch 1 by typing the following from the System Controller's** `sc>` **prompt:**

```
sc> console ssc1/swt
```

12. **Type your user name and password.**

13. **Set up the VLAN database on the switch. Type:**

```
Console#config
Console(config)#vlan database
Console(config-vlan)#vlan vlan ID name name media ethernet
```

where *vlan ID* is the number of the VLAN for the blade and *name* is the VLAN name. In FIGURE 8-2, the VLAN for blade s8 has ID number 18 and a suitable VLAN name would be `snp0private`. To set up the entire VLAN database, type:

```
Console#config
Console(config)#vlan database
Console(config-vlan)#vlan 18 name snp18private media ethernet
Console(config-vlan)#vlan 19 name snp19private media ethernet
Console(config-vlan)#vlan 20 name snp20private media ethernet
Console(config-vlan)#vlan 21 name snp21private media ethernet
Console(config-vlan)#vlan 22 name snp22private media ethernet
Console(config-vlan)#vlan 23 name snp23private media ethernet
Console(config-vlan)#vlan 24 name snp24private media ethernet
Console(config-vlan)#vlan 25 name snp25private media ethernet
Console(config-vlan)#exit
```

14. **Configure the dedicated VLAN for each blade port from snp0 through snp7:**

```
Console#config
Console(config)#interface ethernet snp8
Console(config-if)#no switchport gvrp
Console(config-if)#switchport allowed vlan add 18 untagged
Console(config-if)#switchport native vlan 18
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#exit
```

Repeat these configuration commands for each blade port (and its VLAN) from snp9 (VLAN 19) through snp15 (VLAN 25).

15. **Include each uplink port in the dedicated VLAN for the blade it is to be dedicated to:**

```
Console#config
Console(config)#interface ethernet netp0
Console(config-if)#switchport allowed vlan add 18 untagged
Console(config-if)#switchport native vlan 18
Console(config-if)#switchport allowed vlan remove 1
Console(config-if)#switchport ingress-filtering
Console(config-if)#exit
```

Repeat these configuration commands for each uplink port (and its associated blade VLAN) from netp1 (VLAN 18) through netp7 (VLAN 25).

16. **Shut down the switch ports that you are not using (snp0 through snp7):**

```
Console#config
Console(config)#interface ethernet snp0
Console(config-if)#shutdown
Console(config-if)#exit
```

17. Repeat Step 7 for blade ports snp1 through snp7.

18. **Disable spanning tree on the switch:**

```
Console#config
Console(config)#no spanning-tree
Console(config)#exit
```

19. **Save the changes you have made. Type:**

```
Console#copy running-config startup-config
Startup configuration file name [default filename]:filename
Write to FLASH Programming
-Write to FLASH finish
Success

Console#
```

where *default filename* is the current startup configuration file, and *filename* is the name you want to give to a new startup configuration file. If you type [ENTER] instead of specifying a new file name, the running configuration will be written to the current startup configuration file.

Note – Because blades s0 through s7 only have an interface to switch 0, and blades s8 through s15 only have an interface to switch 1, you must configure the blades to use the interface that is available. On Solaris blades, type `boot net0` or `boot net1` on the Solaris command line. On Linux blades, power on the blade from the System Controller's sc> prompt; immediately type `sc>console s`*n* (where *n* is the slot number of the blade); when the BIOS banner apears press the [Del] key to enter the BIOS configuration menu; use the arrow keys to select the Boot menu; use the +/- keys to the make sure correct boot device (`snet0` or `snet1`, as appropriate) is at the top of the list of boot devices.

APPENDIX **A**

Useful Tasks You Will Need to Perform on the Switches

This appendix describes how to perform certain tasks that can only be performed at the command-line interface to a switch. You will need to refer to it when you are configuring the blade system chassis.

For instructions about logging into the switch command-line interface, see Chapter 2.

This chapter contains the following sections:

A.1 Navigating the Command Prompts

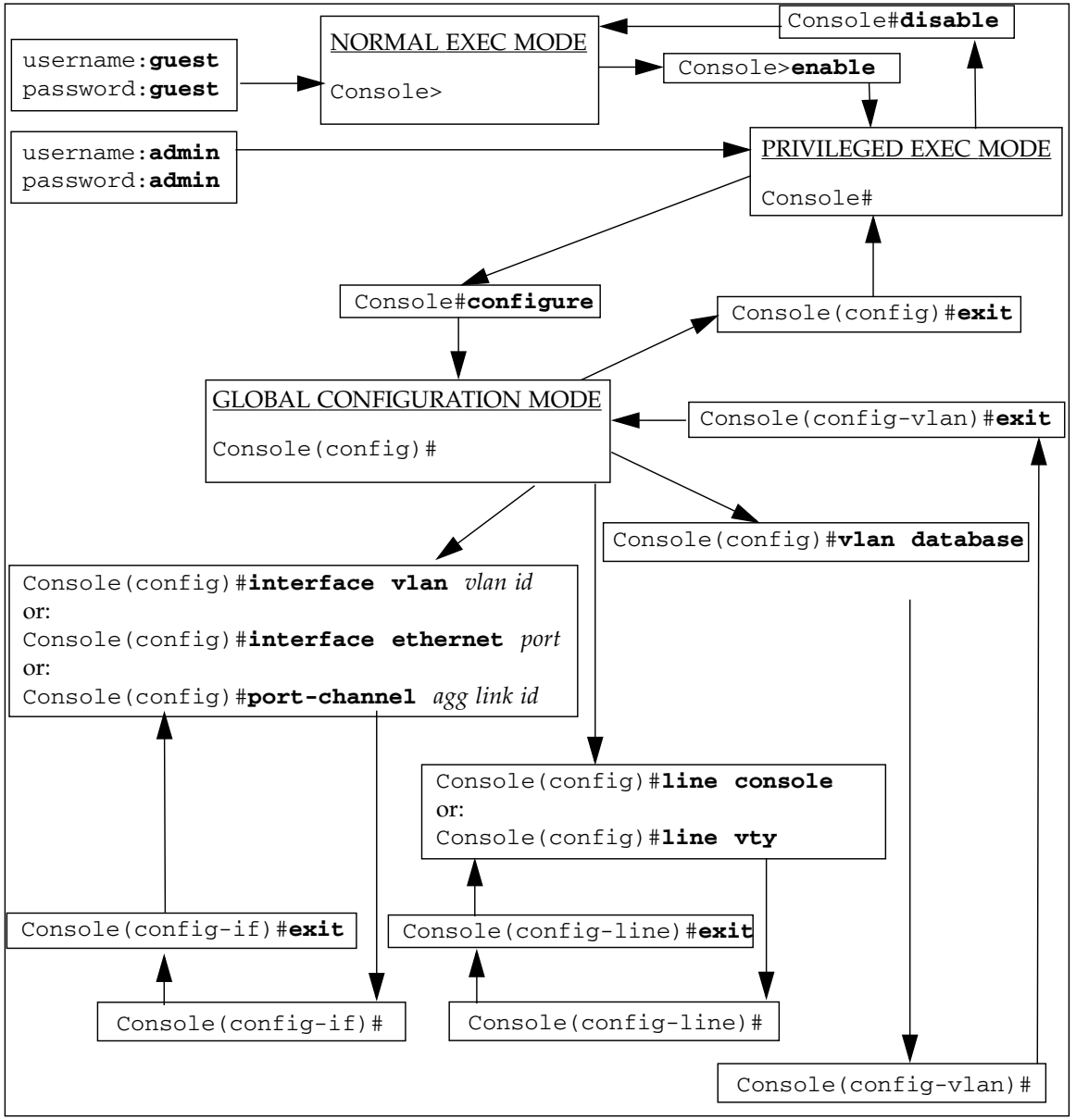

FIGURE A-1 Map of Command Prompts for the Switch

A.2 Exiting the Command-line Interface

A.2.1 Exiting From the Switch to the System Controller

- **To exit from the command-line interface on the switch and return to the System Controller's command-line interface, type the '#' character followed immediately by the '.' character.**

  Wherever you are in the switch's command-line interface, the '#.' escape sequence will return you to the System Controller's command-line interface.

  For example, type (but note that the characters are not echoed to the screen):

```
Console(config)##.
```

A.2.2 Exiting to the Switch's Login Prompt

- **To return to the switch's login prompt, type** `exit` **or** `end` **until you are at the** `Console#` **prompt, then type:**

```
Console#exit
```

A.3 Accessing the Web-based Graphical User Interface

You can use a telnet or a web connection to the switch provided that you have set up a DHCP server on your management network. To ensure that the switch receives the same address each time it boots (and makes a DHCP request), you need to specify the following client identifier on your DHCP server: SUNW,SWITCH_ID=*serial number of chassis*, 0 (for the switch in SSC0) or SUNW,SWITCH_ID=*serial number of chassis*, 1 (for the switch in SSC1).

To use the web-based GUI, type the IP address or host name of the switch into the web browser's URL field:

```
http://ip address or host name
```

The standard port number for an http server is 80 but it is possible to configure the switch to use a different port number for http (for information about how to do this, refer to the *Sun Fire B1600 Blade System Chassis Switch Administration Guide*). If you have configured the switch to use a different http port number, you need to specify that number after the ip address of the switch in the URL field of your browser. For example, if a switch with the host name `switch_1` has been configured to use port 75 for http, then to access it you would type:

```
http://switch_1 75
```

To access the same switch by IP address, you would type the IP address followed by the port number (75). For example:

```
http://192.168.1.202 75
```

For more information about using the web-based GUI to configure the switch, refer to the *Sun Fire B1600 Blade System Chassis Switch Administration Guide*.

A.4 Viewing Online Help for the Switch CLI

- **To view information about using online help, type `help` at any time.**
- **To view context-sensitive online help at any time, type `?`.**

  This displays a list of commands or parameters. If you are at a command prompt, typing `?` will display a list of the commands available for the current command mode. If you want to know the parameter or parameters required for a command, type the first word of the command followed by `?`. This will display a list of the parameters you can enter plus a description of each one. Each time you type `?` after typing an incomplete command, the part of the command you have already typed will be echoed to the console. This means that you do not need to re-type that information.

  Sample help information for using the `vlan database` command to go into the command mode for configuring VLANs is as follows:

```
Console(config)#vlan
% Incomplete command.
Console(config)#vlan ?
  database  Enter VLAN database mode
Console(config)#vlan database ?
  <cr>
```

where `<cr>` indicates that there are no more parameters required and you must press [ENTER] to return to the command prompt.

A.5 Restoring the Switch to its Factory Default State

For information about the factory default settings of the switch, refer to the *Sun Fire B1600 Blade System Chassis Switch Administration Guide*.

To restore the switch to its factory default settings, do the following:

1. **To check whether the switch is using its factory default configuration, type:**

```
Console#whichboot
                   file name  file type       startup  size (byte)
 -------------------------  -------------- -------- --------
                     diag74  Boot-Rom ima    Y      114248
             runtime_v00423  Operation Code  Y      1429204
Factory_Default_Config.cfg  Config File     Y      2574
```

   If the bottom line of the output from this command includes "`Factory_Default_Config.cfg`" in the `file name` column, then the switch is using the default configuration.

2. **To make the switch use its factory default configuration, type:**

```
Console#configure
Console(config)#boot system config Factory_Default_Config.cfg
Console(config)#exit
```

3. **Reboot the switch using the factory default configuration.**

   Type:

```
Console#reload
```

4. **When prompted for your user name and password, type** `admin` **for both.**

A.6 Resetting the Switch

A typical reason for resetting the switch is to revert to the startup configuration after making some changes to the running configuration (and you want to discard those changes).

Another reason for resetting the switch is if you have created or downloaded a new configuration file and you want to designate the new file as the default startup file.

Note – Before you reset the switch, save any changes you configuration changes you have made that you want to perserve.

- **To reset the switch from the switch's command line, type:**

```
Console#reload
```

- **Alternatively, you can reset the switch from the System Controller's command-line.**

  Type the following at the sc> prompt:

```
sc>reset sscn/swt
```

  where *n* is 0 or 1depending on whether you are resetting SSC0 or SSC1.

A.7 Setting the IP address, Netmask, and Default Gateway

1. **Set the IP address and netmask by typing:**

```
Console#configure
Console(config)#interface vlan vlan id
Console(config-if)#ip address ip address netmask
Console(config-if)#exit
```

   where:

   - *vlan id* is the number of the VLAN (by default, 2) that contains the switch's network management port, NETMGT. If you are uing the factory default configuration, specify 2.
   - *ip address* is the IP address you want the switch to use.
   - *netmask* is the netmask you want to set (for example, 255.255.255.0).

2. **To set the default gateway, type:**

```
Console(config)#ip default-gateway ip address
Console(config)#exit
```

   where *ip address* is the IP address of the device you are specifying as the default gateway.

3. **To confirm the change you have made to the setting for the default gateway, type:**

```
Console#show running-config
building running-config, please wait.....
:
!
interface ethernet NETMGT
 description External RJ-45 connector NETPMGT
 switchport allowed vlan add 2 untagged
 switchport native vlan 2
 switchport allowed vlan remove 1
 switchport forbidden vlan add 1
 spanning-tree edge-port
!
interface vlan 2
 ip address 129.156.203.3 255.255.255.0
 ip dhcp client-identifier text SUNW,SWITCH_ID=900002,0
!
!
!
ip default-gateway  129.156.203.8
:
Console#
```

The : characters in the sample output above indicate omitted information. The setting for the default gateway is near the end of the output from the `show running-config` command.

A.8 Setting up VLANs

By default the switch has a management VLAN (VLAN 2) containing its management port (NETMGT), and a data VLAN containing all other ports.

For more information about using VLANs, see Chapter 5, Chapter 6, and Chapter 7.

To create an additional VLAN, you need to set up the VLAN and add ports to it individually.

1. **From the `Console#` prompt, type:**

```
Console#configure
```

2. **Go into vlan configure mode by typing:**

```
Console(config)#vlan database
```

3. **Create the VLAN:**

```
Console(config-vlan)#vlan vlan identifier media ethernet
```

where *vlan identifier* is a number from 1 through 4094.

4. **To give the VLAN a name, type:**

```
Console(config-vlan)#vlan vlan identifier name media ethernet
```

where *vlan identifier* is the number of the VLAN and *name* is the name you want to use for the VLAN.

5. **Populate the VLAN by adding it to individual ports.**

   a. **To do this, first return to configure mode by typing:**

```
Console(config-vlan)#exit
```

   b. **Then enter the configure interface mode by typing:**

```
Console(config)#interface ethernet port
```

where *port* is the name of the port you want to include in the VLAN.

   c. **Add the VLAN to a port by typing:**

```
Console(config-if)#switchport allowed vlan add vlan identifier
```

   d. **Repeat Step a through Step c for each port you want to include on the new VLAN.**

A.9 Saving Your Switch Settings

Note – Make sure you save any switch settings that you want to persist beyond the next reboot of the switch.

- **To save any changes you have made, copy the running configuration firmware to the startup configuration firmware.**

  To do this, type the following in the switch console:

```
Console#copy running-config startup-config
Startup configuration file name [default filename]:filename
Write to FLASH Programming
-Write to FLASH finish
Success

Console#
```

  where *default filename* is the current startup configuration file, and *filename* is the name you want to give to a new startup configuration file. If you type [ENTER] instead of specifying a new file name, the running configuration will be written to the current startup configuration file.

A.10 Copying the Configuration of the First Switch to the Second

The procedure for transferring a configuration file from one switch to the other requires you to use TFTP. This means that to perform it you need to have a TFTP server available on your network. The instructions in this section tell you how to do this. They then tell you how to perform the file transfer.

If you have VLANs set up on the switch to separate the different regions of your network from each other, and you are also using IP Network Multipathing (IPMP) to give your server blades redundant connections to the network, you must make sure that the configuration of the second switch matches that of the first.

Caution – If the VLAN configuration of the second integrated switch does not match the VLAN configuration of the first, then data passing through the second switch will not be governed by the VLAN definitions on the first. Similarly, any protection of your management network that is enforced by the packet filter on the first switch will be lost if you do not duplicate it on the second switch.

To ensure that the second switch inside the Sun Fire B1600 blade system chassis has the same configuration as the first, follow the instructions in this section.

A.10.1 Setting up a TFTP Server

To configure a Solaris system on your network to serve TFTP requests, do the following:

1. **On the system that you intend to set up as the TFTP server, log in as root.**
2. **Use a text editor to un-comment the following line in the file** `/etc/inetd.conf`**:**

```
tftp  dgram udp6 wait root /usr/sbin/in.tftpd in.tftpd -s /tftpboot
```

3. **On the same system create a TFTP home directory by typing the following at the Solaris prompt:**

```
# mkdir /tftpboot
# chown root /tftpboot
# chmod 755 /tftpboot
# cd /tftpboot
# ln -s . tftpboot
```

4. **Restart** `inetd` **by typing:**

```
# pkill -HUP inetd
```

5. **Verify that TFTP is working.**

   To do this, use TFTP to get a file from the `/tftpboot` directory. Follow the instructions below:

a. On the system that you are using as the TFTP server, copy any file (for example, the Solaris `/etc/release` file) to the `/tftpboot` directory.

To copy the `/etc/release` file, at the Solaris prompt, type:

```
# cp /etc/release /tftpboot/filename
```

where *filename* is the name of the file you intend to make available on the TFTP server.

b. Make the file you have just copied read-only by all:

```
# chmod 444 /tftpboot/filename
```

where *filename* is the name of the file you intend to make available on the TFTP server

c. Get the file from the TFTP server you have created.

At the Solaris prompt on another system, type the following commands:

```
% tftp tftp server
tftp>get filename
```

where *tftp server* is the host name or IP address of the system running the TFTP server you have set up, and *filename* is the name of the file you want to try getting from the TFTP server.

d. Still on the Solaris system that you used to initiate the `get` command, check the content of the file by typing:

```
# cat filename
```

where *filename* is the name of the file you have transferred from the TFTP server.

Note – Note that TFTP is not the same as FTP. It does not display the same error messages as FTP, and you cannot use the `cd or ls` commands (or indeed most other commands) that FTP allows you to use.

A.10.2 Transferring the Switch Configuration File

When you have created a TFTP server, and you have also finished configuring the switch in SSC0 or SSC1, then duplicate the configuration of the switch you have configured onto the second switch.

To do this, follow the instructions below. (The instructions assume you are duplicating the configuration of the switch in SSC0 onto the switch in SSC1, but of course you can duplicate the switch in SSC1 onto the switch in SSC0.)

1. **Configure switch 0 according to your requirements by following the instructions in Chapter 2, Chapter 3, Chapter 5, Chapter 6, and/or Chapter 7.**

2. **Save the configuration of switch 0 to a file called, for example,** `standard.cfg`**.**

   To do this, at the switch `Console#` prompt, type:

```
Console#copy running-config file
Destination configuration file name: standard.cfg
Write to FLASH Programming
-Write to FLASH finish
Success.

Console#
```

3. **Upload the** `standard.cfg` **file to the TFTP server.**

   To do this:

   a. **Log into the TFTP server as root.**

   b. **Change directory to** `/tftpboot`**.**

   c. **Create an empty file called** `standard.cfg`**.**

```
#>standard.cfg
```

4. **Make the file read-writeable by all:**

```
#chmod 666 standard.cfg
```

5. **At the command-line interface to the switch, type:**

```
Console#copy file tftp
Choose file type:
1. config: 2.opcode: <1-2>:1
Source file name: filename
TFTP server ip address: IP address
Desitination file name: filename
Console#
```

where *filename*, in both cases, is `standard.cfg` (if this is the name of the file that you saved your switch configuration to) and *IP address* is the IP address of the TFTP server.

6. **On the TFTP server, use a text editor to open the `standard.cfg` file.**

Change the entry for the host name of switch 0 so that it contains the host name of switch 1:

```
!
hostname host name of switch 1
```

If you have chosen to have manually assigned IP addresses for the switches, you must change the entry for the IP address and netmask so that it contains the IP address and netmask of switch 1 instead of those of switch 0:

```
interface vlan 2
  ip address ip address  netmask
```

If you are using DHCP there is no need to change the IP address and netmask or the DHCP client identifier. The IP address and netmask will be automatically assigned by your DHCP server. And the DHCP client identifier will be automatically assigned by the active System Controller whenever the switch is reset.

7. **Save this file with a suitable name, for example, `standard1.cfg`.**

8. **Log into switch 1 and (if the switch has not been assigned an IP address by DHCP) set a temporary management IP address on it.**

If you have already configured the login and password information for switch 1, then log in using these. If not, log in using the factory default user name (`admin`) and password (`admin`).

To set the IP parameters, follow the instructions in Section A.7, “Setting the IP address, Netmask, and Default Gateway” on page A-7

9. **Download** `standard1.cfg` **from the TFTP server to Switch 1.**

   To do this, type:

```
Console#copy tftp file
TFTP server ip address:IP address
Choose file type:
1. config: 2.opcode: <1-2>:1
Source file name:standard1.cfg
Destination file name:standard1.cfg
Console#
```

10. **Make this the startup configuration for switch 1.**

    Type:

```
Console#configure
Console(config)#boot system config standard1.cfg
Console(config)#exit
Console#
```

11. **Reload the switch firmware.**

    Type:

```
Console#reload
```

A.11 Using Aggregated Links for Resilience and Performance

If you have external data ports that connect to the same switch as each other, we recommend that you combine them into aggregated links. This gives you both resilience and added performance.

For example, if you had four separate connections to the same external switch and one of those connections failed because of a cabling fault, any communication on the broken connection would be lost. But, if you had set up an aggregated link to include all four connections to the external switch, then if one connection failed, communication would continue on the remaining connections defined in the aggregated link.

As long as no connections are broken, the integrated switch treats all the connections in the aggregated linkas a single high-bandwidth connection to the same network.

Note – If you have duplicate connections to an external switch, hub, or router and you do not make them into an aggregated link, then the integrated switch's Spanning Tree facility will block all of them except one. Therefore, although your network will still benefit from redundancy, none of the duplicated connections will be active until the single unblocked connection fails.

The following sample commands create an aggregated link using ports NETP2, NETP3, and NETP4:

```
Console(config)#interface port-channel 1
Console(config-if)#exit
Console(config)#interface ethernet NETP2
Console(config-if)#channel-group 1
Console(config-if)#exit
Console(config)#interface ethernet NETP3
Console(config-if)#channel-group 1
Console(config-if)#exit
Console(config)#interface ethernet NETP4
Console(config-if)#channel-group 1
Console(config-if)#exit
Console(config)#
```

A.12 Enabling Secure Management of Blades

The switch contains a packet filter which by default blocks all traffic from the server blades to the switch's management port (NETMGT). This prevents any possible hostile attack on your management network being launched from a server blade (in the event, for example, of a hacker gaining access to a blade from the public network). However, it means that you cannot communicate directly with the server blades through the management port until you have configured the packet filter to permit management traffic to pass from the server blades to the management port. This section tells you how to do that.

Note – By default the packet filter permits no traffic to pass from the server blades to the management port (NETMGT). Exercise caution when deciding to enable traffic to pass through the packet filter, and in any case only enable the protocols you know you require.

The instructions below tell you which commands to use to permit DHCP, BOOTP, TFTP, SUNRPC, SNMP and NFS frames to pass from the server blades through the packet filter to the management port. This is the minimum set of protocols required to enable the server blades to be administered via the management port:

1. **Enable DHCP and BOOTP frames to pass through the packet filter.**

   In the switch console, type:

```
Console#configure
Console(config)#ip filter permit udp 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 67-68
```

2. **Enable TFTP frames to pass through the packet filter.**

   Type:

```
Console#configure
Console(config)#ip filter permit udp 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 69
Console(config)#ip filter permit udp 0.0.0.0 0.0.0.0 699-65535
0.0.0.0 0.0.0.0 699-65535
```

3. **Enable SunRPC frames to pass through the packet filter.**

   Type:

```
Console#configure
Console(config)#ip filter permit udp 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 111
Console(config)#ip filter permit tcp 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 111
```

4. **Enable SNMP frames to pass through the packet filter.**

   Type:

```
Console#configure
Console(config)#ip filter permit udp 0.0.0.0 0.0.0.0 161 0.0.0.0
0.0.0.0
Console(config)#ip filter permit tcp 0.0.0.0 0.0.0.0 161 0.0.0.0
0.0.0.0
Console(config)#ip filter permit udp 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 162
Console(config)#ip filter permit tcp 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 162
```

Note – Note that the 161 port is the port for SNMP requests on a managed device, and the 162 port is the port for SNMP traps on a managed device. The SNMP traps originate on the managed device. The SNMP commands originate on an SNMP management station.

---

5. **Enable NFS frames to pass through the packet filter.**

   Type:

```
Console#configure
Console(config)#ip filter permit udp 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 2049
Console(config)#ip filter permit tcp 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 2049
```

Note – For more information about using the `ip filter permit` command, refer to the *Sun Fire B1600 Blade System Chassis Switch Administration Guide*. For a list of the port numbers associated with particular protocols, refer to the `/etc/services` file or the `/etc/inet/services` file on a Unix system. For a complete list of the port numbers associated with IP services, refer to the web site of the Internet Assigned Numbers Authority (`http://www.iana.org`).

A.13 Setting Up a Named User on the Switch

1. **In the switch console, type:**

```
Console#configure
```

2. **Type:**

```
Console(config)#username username access-level 15
```

where *username* is the name you want the user to type when he or she logs in.

The number 15 in the first command signifies that the new user will have access to Privileged Exec mode. (To give a user access to Normal Exec mode only, type 0 instead of 15.)

3. **Type:**

```
Console(config)#username username password 0 password
```

where *username* is the name you want the user to type when he or she logs in, and *password* is the new user's password.

The 0 in this command signifies that the value typed for *password* is not encrypted. If you were to enter the value in its encrypted form, you would have to indicate this by typing 7 instead of 0 before the encrypted text that you specify as the password. However, there is no reason to enter the password in its encrypted form. The switch stores the password

A.13.1 Understanding Why the Switch Needs to be Told That a Password is Not Encrypted

When you set up a password for a user, you need to tell the switch (by putting a 0 or a 7 on the command line) whether you are specifying the user's password in plain text or encrypted form. In practice, you will always specify a 0 to indicate plain text.

The parameter 7 (indicating encrypted text) exists only for the switch's internal use at boot-time. It stores its passwords in encrypted form for security in the configuration file, because the passwords are viewable in this file. For example, you can view them by typing:

```
Console#> show running-config
:
:
username admin access-level 15
username admin password 7 21232f297a57a5a743894a0e4a801fc3
username guest access-level 0
username guest password 7 084e0343a0486ff05530df6c705c8bb4
enable password level 15 7 1b3231655cebb7a1f783eddf27d254ca
:
:
```

However, at boot time the switch processes the passwords from this file. And it is therefore during the boot process that it needs to be able to distinguish between passwords presented as plain text and passwords presented as encrypted text. In other words, the parameter 7 that it is possible to enter on the command line (when you set up a user with a password) is of no practical use to you as a user.

Note – You cannot use passwords stored in encrypted form as a method of accessing the switch if you have forgotten the passwords, because you need password access to the switch in order to view them.

A.13.2 The Default User Names and Passwords for the Switch

The default user name (with full access rights) is `admin`.
The password is `admin`.

The default user name for guest access (with limited privileges) is `guest`.
The password is `guest`.

The default password for the `enable` command (to transition from `guest` access to full access) is `super`.

A.14 Viewing Information About the Switch and its Configuration

This section contains the following information:

A.14.1 Checking the IP Address and VLAN Id

- **To check the IP address and VLAN Id of the management port, at the `Console#` prompt, type:**

```
Console#show ip interface
 IP address and netmask: 129.156.223.215 255.255.255.0 on VLAN 2,
 and address mode: User specified.
```

A.14.2 Checking the VLAN Configuration

- **To check the VLAN configuration of the switch, at the `Console#` prompt, type:**

```
Console#show vlan

VLAN Type    Name              Status    Ports/Channel groups
---- ------- ----------------- --------- ---------------------------------
   1  Static      DefaultVlan   Active   SNP0    SNP1    SNP2    SNP3    SNP4
                                         SNP5    SNP6    SNP7    SNP8    SNP9
                                         SNP10   SNP11   SNP12   SNP13   SNP14
                                         SNP15   NETP0   NETP1   NETP2   NETP3
                                         NETP4   NETP5   NETP6   NETP7
   2  Static         MgtVlan   Active  NETMGT
```

A.14.3 Finding Out Who is Logged On

- **To find out who is logged into the command-line and web interfaces, at the** `Console#` **prompt, type:**

```
Console#show users
Username accounts:
  Username Privilege
  -------- ---------
     admin        15
     guest         0

 Online users:
  Line        Username Idle time (h:m:s) Remote IP addr.
  ----------- -------- ----------------- ---------------
* 0   console    admin           0:00:00
```

A.14.4 Inspecting the Current or Startup Configuration

- **To view the current configuration of the switch, at the** `Console#` **prompt, type:**

```
Console#show running-config
```

If anybody has changed any switch settings since the last time the switch booted, the running configuration will differ from the start-up configuration

- **To view the configuration that the switch received when it last booted (and which it will receive the next time it boots), at the** `Console#` **prompt, type:**

```
Console#show startup-config
```

A.14.5 Finding Out Firmware Version Numbers

- **To find out firmware (and other) version information, at the** `Console#` **prompt, type:**

```
Console#show version

Unit1
 Serial number           :
 Service tag             :
 Hardware version        :r0b
 Number of ports         :25
 Main power status       :up
 Redundant power status  :not present

Agent(master)
 Unit id                 :1
 Loader version          :0.0.6.7
 Boot rom version        :1.0.0.8
 Operation code version  :1.0.0.6
Console#
```

A.14.6 Viewing MAC Address and General System Information

- **To find out the MAC address, and to find out firmware (and other) version information, at the `Console#` prompt, type:**

```
Console#show system

System description: Sun Fire B1600
System OID string: 1.3.6.1.4.1.42.2.24.1

System information

 System Up time: 0 days, 7 hours, 41 minutes, and 4.4 seconds
 System Name            : [NONE]
 System Location        : [NONE]
 System Contact         : [NONE]
 MAC address            : 08-00-20-7A-92-0B
 Web server             : enable
 Web server port        : 80
 Web secure server      : enable
 Web secure server port : 443

 POST result

--- Performing Power-On Self Tests (POST) ---
UART Loopback Test ..................... PASS
Timer Test ............................. PASS
DRAM Test .............................. PASS
I2C Initialization ..................... PASS
Runtime Image Check .................... PASS
PCI Device Check ....................... PASS
AN983 Initialization ................... PASS
AN983 Internal Loopback Test ........... PASS
Switch Driver Initialization ........... PASS
Switch Internal Loopback Test .......... PASS
------------------- DONE --------------------
Console#
```

APPENDIX **B**

Setting up DHCP to Configure the IP Addresses for Solaris Blades

This appendix supplements the instructions in the *Solaris Advanced Installation Guide* and the *DHCP Administration Guide*. It enables you to complete the configuration of the Network Install Server and the DHCP server on your data network so that the server blades on the system chassis can receive their IP addresses dynamically.

The instructions assume that you have added the Solaris image to a Network Install Server and that you have a DHCP server on the data network.

This appendix contains the following sections:

B.1 Network Install Server Tasks

- **On the Network Install Server, run** `add_install_client` **with the** `-d` **option.**

  This command copies a DHCP-capable `inetboot` file from the Solaris image to the `/tftpboot` directory. To execute the command, type:

```
# cd path/Solaris_8/Tools
# ./add_install_client -d -s installserv:/images/2.8 -c
configsrv:/config -p configsrv:/config SUNW.Serverblade1 sun4u

To enable SUNW.Serverblade1 in the DHCP server, add an entry to
the server with the following data:

Install server      (SinstNM)  : installserv
Install server IP   (SinstIP4) : 192.168.160.12
Install server path (SinstPTH) : /images/2.8
Root server name    (SrootNM)  : installserv
Root server IP      (SrootIP4) : 192.168.160.12
Root server path    (SrootPTH) : /images/2.8/Solaris_8/Tools/Boot
Profile location    (SjumpsCF) : configsrv:/config
sysidcfg location   (SsysidCF) : configsrv:/config
```

  where *path* is the location of the Solaris image on the Network Install Server. (Note that the second command in the above example has wrapped onto the next line.) The output above uses sample IP data.

B.2 DHCP Server Tasks

1. **On the DHCP server create the options that you want passed to the blades during the Solaris Jumpstart.**

   (This is the information that would be gathered from the `/etc/bootparams` file during a non-DHCP Jumpstart.)

The options you need to create are listed in TABLE B-1.

TABLE B-1 The DHCP Options That Must Be Passed to the Blade During Jumpstart

Option name	Description
SrootIP4	IP address of root server
SrootNM	Hostname of root server
SrootPTH	Path to the boot image (for example, /images/2.8/Solaris_8/Tools/Boot)
SinstIP4	IP address of the Network Install Server
SinstNM	Hostname of Network Install Server
SsysidCF	Location of the sysidcfg file (for example, configsrv:/config)
SjumpsCF	Location of profile and rules.ok directory (for example, configsrv:/config)
SbootFIL	Path to the kernel (for example, /platform/sun4u/kernel/sparcv9/uni)
Sterm	Terminal type used during the install

The following are sample commands for creating the options listed in TABLE B-1:

```
# dhtadm -A -s SrootIP4 -d 'Vendor=SUNW.Serverblade1,2,IP,1,1'
# dhtadm -A -s SrootNM -d 'Vendor=SUNW.Serverblade1,3,ASCII,1,0'
# dhtadm -A -s SrootPTH -d 'Vendor=SUNW.Serverblade1,4,ASCII,1,0'
# dhtadm -A -s SbootFIL -d 'Vendor=SUNW.Serverblade1,7,ASCII,1,0'
# dhtadm -A -s SinstIP4 -d 'Vendor=SUNW.Serverblade1,10,IP,1,1'
# dhtadm -A -s SinstNM -d 'Vendor=SUNW.Serverblade1,11,ASCII,1,0'
# dhtadm -A -s SinstPTH -d 'Vendor=SUNW.Serverblade1,12,ASCII,1,0'
# dhtadm -A -s SsysidCF -d 'Vendor=SUNW.Serverblade1,13,ASCII,1,0'
# dhtadm -A -s SjumpsCF -d 'Vendor=SUNW.Serverblade1,14,ASCII,1,0'
# dhtadm -A -s Sterm -d 'Vendor=SUNW.Serverblade1,15,ASCII,1,0'
```

2. **Create macros containing the options you require (including the options you created in the Step 1).**

TABLE B-2 The Macros You Need to Create

Macro name	Macro contents (macros can contain other macros)
`Solaris`	`SrootIP4`, `SrootNM`, `SinstIP4`, `SinstNM`, `Sterm`, `SjumpsCF`, `SsysidCF`
`sparc`	`SrootPTH`, `SinstIP4`
`sun4u`	`Solaris`, `sparc`
`SUNW.Serverblade1`	`SbootFIL`, `sun4u`
*network name**	`Subnet`, `Router`, `Broadcst`, and `BootSrvA`

**network name* is the IP address that identifies the network containing the clients. You need to create one of these macros for each client subnet except the subnet containing the primary interface of the DHCP server.

The following are sample commands for creating the macros you require:

```
# dhtadm -A -m Solaris -d ':SrootIP4=192.168.160.12:SrootNM=
"bootsrv":SinstIP4=192.168.160.15:SinstNM="installsrv":Sterm=
"xterm":SjumpsCF="configsrv:/config":SsysidCF=
"configsrv:/config":'
# dhtadm -A -m sparc -d ':SrootPTH=
"/images/2.8/Solaris_8/Tools/Boot":SinstPTH="/images/2.8":'
# dhtadm -A -m sun4u -d ':Include=Solaris:Include=sparc:'
# dhtadm -A -m SUNW.Serverblade1 -d ':SbootFIL=
"/platform/sun4u/kernel/sparcv9/unix":Include=sun4u:'
# dhtadm -A -m 192.168.160.0 -d ':Subnet=255.255.255.0:Router=
192.168.160.254:Broadcst=192.168.160.255:BootSrvA=
192.168.160.12:'
```

3. **Add the client's host name and IP address to the hosts database (that is, to `/etc/hosts`).**

4. **Map the `SUNW.Serverblade1` macro to the client.**

   Type:

```
# pntadm -A dhcpclient01 -i 01MACaddress -m SUNW.Serverblade1 -
s DHCP server network name
```

   where:

MACaddress is the client's MAC address,

DHCP server is the DHCP server's hostname, and

network name is the IP address that identifies the network containing the client (note that the sample comand line above has wrapped onto the next line).

B.3 Server Blade Tasks

When the network environment is configured to provide the server blades with two IP addresses each, follow the instructions in this section. These instructions assume that the server blade you are configuring has booted from the network and been provided with an IP configuration for its primary (`ce0`) interface.

1. **From the System Controller's `sc>` prompt, access the blade console.**

   Type:

   ```
   sc> console sn
   ```

   where *n* is the slot number for the blade you are configuring.

2. **From Solaris prompt, type:**

   ```
   # ifconfig ce1 plumb
   ```

3. **Finally, type:**

   ```
   # ifconfig ce1 auto-dhcp up
   ```

APPENDIX **C**

Setting Up Solaris Blades Using Web Start Flash Archives

This appendix supplements the instructions in the *Solaris 8 Advanced Installation Guide* on how to configure a Network Install Server.

When you have booted the first Solaris blade from the Network Install Server, you can add the application software you intend to run on the blade, then follow the instructions in the *Solaris Advanced Installation Guide* to create a Web Start Flash Archive.

Using Web Start Flash Archives on the Sun Fire B100s Solaris server blades (in the Sun Fire B1600 Blade System Chassis) enables you to replicate one blade's Operating Environment and application software on other blades.

This appendix contains the following sections:

C.1 Using Web Start Flash Archives to Speed up Blade Configuration

A Flash Archive is a snap-shot of a Solaris system and, as such, includes all the files on that system (or, more accurately, all the files that you specify should be included). You need to create the Flash Archive after all the software has been installed onto the blade but before the blade has gone into use. Depending on the software involved and the intended use for the blade, the Flash Archive might need to be created after the software has been installed but before it has been configured. For example, a database server would need its Flash Archive created after the database management software had been installed but before the databases had been created.

If you do not yet know what software applications you want to run on the blades, you can still use the Web Start Flash Archive method to replicate Solaris on multiple blades. This will be quicker than performing individual Jumpstarts.

C.1.1 Creating the Web Start Flash Archive

To create a Flash Archive of the software on a blade, follow the instructions on creating Flash Archives in the *Solaris Advanced Installation Guide*.

C.1.2 Installing an Archived Blade Image Onto Other Blades

To install the archived image onto other blades, follow the instructions on installing a Flash Archive in the *Solaris Advanced Installation Guide*.

C.1.3 Increasing the Performance of the Web Start Flash Archive Installation

You can take advantage of the Gigabit interconnect between the server blades to increase the performance of the Web Start Flash Archive installation.

To do so, use NFS to share the location of the Flash Archive you created (that is, to share the archived image of the first blade's software). Follow the instructions below:

1. **At the Solaris prompt on the blade whose image you intend to replicate, type:**

```
#share -F ufs flash location
```

where *flash location* is the location of the blade's Flash Archive. For example:

```
#share -F ufs /var/tmp
```

2. **On the Network Install Server, modify the install profile to make it point to the location of the Flash Archive on the first blade.**

APPENDIX **D**

System Controller Commands

This appendix lists the commands available from the System Controller's `sc>` prompt.

This appendix contains the following sections:

D.1 Power Commands for the Entire Chassis

Note – You can power off (or power down to a ready-to-remove or standby) state all components at once except the active System Controller. The blade system chassis is designed so that you cannot power off or down the active System Controller in a single command. For information about powering down the active System Controller, refer to the *Sun Fire B1600 Blade System Chassis Administration Guide.*

TABLE D-1 Commands for Powering On or Off or Powering Down all Components

Command and Option (if any)	Effect of the Command
`sc> poweron ch`	Powers on all components. Use this command to recover all components at once from a powered down, ready-to-remove, or standby power state.
`sc> poweroff ch`	Powers off all components on the chassis except the active System Controller.
`sc> poweroff -f ch`	Powers off all components (except the active System Controller) even if an orderly shutdown of the operating system on a component has failed.
`sc> poweroff -F ch`	Powers off all components (except the active System Controller) even if there are commands (from other users) involving some components that have not finished executing.
`sc> poweroff -y ch`	Powers off all compononents (except the active System Controller) without displaying the confirmation prompt.
`sc> poweroff -s ch`	Powers down all components (except the active System Controller) to standby mode (equivalent of `standbyfru ch` command).
`sc> poweroff -r ch`	Powers down all components (except the active System Controller) to a state in which it is safe for them to be removed. The `-r` option also turns on the "ok to remove" LED for each component (equivalent of the `removefru ch` command).
`sc> standbyfru ch`	Powers down all components (except the active System Controller) to standby mode (equivalent of the `poweroff -s ch` command).

TABLE D-1 Commands for Powering On or Off or Powering Down all Components

Command and Option (if any)	Effect of the Command
`sc> standbyfru -f ch`	Powers down all components (except the active System Controller) to standby mode even if an orderly shutdown of the operating system on a component has failed.
`sc> standbyfru -F ch`	Powers down all components (except the active System Controller) to standby mode even if there are commands (from other users) involving some components that have not finished executing.
`sc> standbyfru -y ch`	Powers down all components (except the active System Controller) to standby mode without displaying the confirmation prompt.
`sc> removefru ch`	Powers down all components (except the active System Controller) to a state in which it is safe for them to be removed; this command also turns on the "ok to remove" LED for each component (equivalent of the `poweroff -r ch` command).
`sc> removefru -f ch`	Powers down all components (except the active System Controller) to a state in which it is safe for them to be removed even if an orderly shutdown of the System Controller's operating system has failed. This command also turns on the "ok to remove" LED for each component.
`sc> removefru -F ch`	Powers down all components (except the active System Controller) to a state in which it is safe for them to be removed even if there are commands (from other users) involving some components that have not finished executing.
`sc> removefru -y ch`	Powers down all components (except the active System Controller) to a state in which it is safe for them to be removed but does not display the confirmation prompt before doing so. This command also turns on the "ok to remove" LED on for each component.

D.2 Power Commands for the System Controllers

Note – You can only power off or down the standby System Controller. For information about powering down the active System Controller, refer to the *Sun Fire B1600 Blade System Chassis Administration Guide*.

TABLE D-2 Commands for Powering On or Off or Powering Down an SSC

Command and Option (if any)	Effect of the Command
`sc> poweron [-F] sscn`	Powers on SSC *n* (where *n* designates the standby SSC and is either 0 or 1 depending on whether the standby System Controller is in SSC0 or SSC1). Use this command to recover the standby SSC from a powered down, ready-to-remove, or standby power state. The `-F` option powers on the standby SSC even if there are commands (from other users logged on to the active System Controller) that involved the standby SSC in its non-powered state and that have not yet finished executing.
`sc> poweroff sscn`	Powers off SSC *n* (where *n* is 0 or 1 depending on whether the standby System Controller is in SSC0 or SSC1).
`sc> poweroff -f sscn`	Powers off the standby System Controller (SSC0 or SSC1) even if an orderly shutdown of the System Controller's operating system has failed.
`sc> poweroff -F sscn`	Powers off the standby System Controller (SSC0 or SSC1) even if the active System Controller has not finished executing commands (from another user) involving the standby System Controller.
`sc> poweroff -y sscn`	Powers off the standby System Controller (SSC0 or SSC1) without displaying the confirmation prompt.
`sc> poweroff -s sscn`	Powers the standby System Controller (SSC0 or SSC1) down to standby-power mode (equivalent of `standbyfru` command).
`sc> poweroff -r sscn`	Powers the standby System Controller down to a state in which it is safe for it to be removed; the `-r` option also turns on the "ok to remove" LED (equivalent of the `removefru` command).

TABLE D-2 Commands for Powering On or Off or Powering Down an SSC

Command and Option (if any)	Effect of the Command
`sc> standbyfru sscn`	Powers the standby System Controller down to standby-power mode (equivalent of the `poweroff -s` command).
`sc> standbyfru -f sscn`	Powers the standby System Controller down to standby-power mode even if an orderly shutdown of its operating system has failed.
`sc> standbyfru -F sscn`	Powers the standby System Controller down to standby-power mode even if the active System Controller has not finished executing commands (from another user) involving the standby System Controller.
`sc> standbyfru -y sscn`	Powers the standby System Controller down to standby-power mode without displaying the confirmation prompt.
`sc> removefru sscn`	Powers the standby System Controller down to a state in which it is safe for it to be removed; this command also turns on the "ok to remove" LED on the SSC's rear panel (equivalent of the `poweroff -r` command).
`sc> removefru -f sscn`	Powers the standby System Controller down to a state in which it is safe for it to be removed even if an orderly shutdown of the System Controller's operating system has failed. This command also turns on the "ok to remove" LED on the SSC's rear panel.
`sc> removefru -F sscn`	Powers the standby System Controller down to a state in which it is safe for it to be removed even if the active System Controller has not finished executing commands (from another user) involving the standby System Controller.
`sc> removefru -y sscn`	Powers the standby System Controller down to a state in which it is safe for it to be removed but does not display the confirmation prompt before doing so. This command also turns on the "ok to remove" LED on the SSC's rear panel.

D.3 Power Commands for the Server Blades

TABLE D-3 Commands for Powering On or Off or Powering Down a Server Blade

Command and Option (if any)	Effect of the Command
`sc> poweron sn`	Powers on the blade in slot *n*. Use this command to recover the blade from a powered down, ready-to-remove, or standby power state.
`sc> poweroff sn`	Powers off the blade in slot *n*.
`sc> poweroff -f sn`	Powers off the blade in slot *n* even if an orderly shut down of the System Controller's operating system has failed.
`sc> poweroff -F sn`	Powers off the blade in slot *n* even if the active System Controller has not finished executing commands (from another user) involving the specified blade.
`sc> poweroff -y sn`	Powers off the blade in slot *n* without displaying the confirmation prompt.
`sc> poweroff -s sn`	Powers the blade in slot *n* down to standby mode (equivalent of `standbyfru` command).
`sc> poweroff -r sn`	Powers the blade in slot *n* down to a state in which it is safe for it to be removed; the `-r` option also turns on the blue "ok to remove" LED on the front of the blade (equivalent of the `removefru` command).
`sc> standbyfru sn`	Powers the blade in slot *n* down to standby mode (equivalent of the `poweroff -s` command).
`sc> standbyfru -f sn`	Powers the blade in slot *n* down to standby mode even if an orderly shut down of the blade's operating system has failed.
`sc> standbyfru -F sn`	Powers the blade in slot *n* down to standby mode even if the active System Controller has not finished executing commands (from another user) involving the specified blade.
`sc> standbyfru -y sn`	Powers the blade in slot *n* down to standby mode without displaying the confirmation prompt.
`sc> removefru sn`	Powers the blade in slot *n* down to a state in which it is safe for it to be removed; this command also turns on the blue "ok to remove" LED on the front of the blade (equivalent of the `poweroff -r` command).

TABLE D-3 Commands for Powering On or Off or Powering Down a Server Blade

Command and Option (if any)	Effect of the Command
`sc> removefru -f s`*n*	Powers the blade in slot *n* down to a state in which it is safe for it to be removed. This command performs power down even if an orderly shutdown of the blade's operating system has failed. Thecommand also turns on the blue "ok to remove" LED on the front of the blade.
`sc> removefru -F s`*n*	Powers the blade in slot *n* down to a state in which it is safe for it to be removed. This command performs power down even if the active System Controller has not finished executing commands (from another user) involving the specified blade.
`sc> removefru -y s`*n*	Powers the blade in slot *n* down to a state in which it is safe for it to be removed but does not display the confirmation prompt before doing so. This command also turns on the blue "ok to remove" LED on the front of the blade.

D.4 Reset Commands for the System Controllers, Switches, and Blades

TABLE D-4 Commands for Resetting Components of the System Chassis

Command and Option (if any)	Effect of the Command
`sc> reset sn`	Resets the server blade in slot *n*.
`sc> reset sn sy`	Resets the server blades in slots *n* and *y*. (Specify the blades you want to reset in a space-separated list.)
`sc> reset -y sn`	Resets the blade in slot *n* without displaying the confirmation prompt.
`sc> reset -x sn`	Performs an externally-initiated reset on the blade in slot *n*.
`sc> reset -F sn`	Forces the specified blade to reset even if the active System Controller has not finished executing commands (for another user) involving that blade.
`sc> reset sscn/swt`	Resets the switch in SSC*n* (where *n* is 0 or 1).
`sc> reset -y sscn/swt`	Resets the switch in SSC*n* without displaying the confirmation prompt.
`sc> reset -F sscn/swt`	Resets the switch in SSC*n* even if the active System Controller has not finished executing commands (for another user) involving that switch.
`sc> reset -x sscn/swt`	Performs an externally-initiated reset on the switch in SSC*n*.
`sc> reset sscn/sc`	Resets the standby System Controller (where *n* is 0 or 1 depending whether the standby System Controller is in SSC0 or SSC1).
`sc> reset -F sscn/sc`	Forces the standby System Controller to reset even if the active System Controller has not finished executing commands (from another user) involving the standby System Controller.
`sc> resetsc`	Resets the active System Controller. Neither of the switches is affected by this reset. You will lose your user session when you reset the System Controller using this command.
`sc> resetsc -y`	Resets the active System Controller without displaying the confirmation prompt.

TABLE D-4 Commands for Resetting Components of the System Chassis

Command and Option (if any)	Effect of the Command
`sc> resetsc -F`	Resets the active System Controller without waiting for any outstanding `flashupdate` or `setupsc` commands (from another user) to finish executing. By default, the System Controller will not initiate the reset of itself until it has finished executing these commands for another user.
`sc> reset sscn`	Resets the standby System Controller (*n* cannot be the active System Controller), both switches, and all server blades installed in the chassis.
`sc> break sn`	If Solaris is running (and it is configured to handle breaks in this way), the `break` command causes a Solaris blade to drop from Solaris into either kadb or OBP, depending on the mode in which Solaris was booted.
`sc> break -y sn`	As above, but the -y option means that you are not prompted to confirm the `break` command that you have initiated.
`sc> break sn sy sx`	As above, but this command applies the break to blades *n*, *y*, and *x*.

D.5 Monitoring Commands

TABLE D-5 Commands for Monitoring the Chassis and its Components

Command and Option (if any)	Effect of the Command
`showsc [-v]`	Displays a summary of the configuration of the active System Controller.
`showplatform [-v]|[-p] {[sscn][sscn/swt][psn][sn][ch]}`	Displays the status (Ok, Faulty, Not Present) of each component. It also displays the status of the Operating System in all domains on the chassis (that is, in the System Controllers, switches, and blades). If you use the -p option, the status of the Operating System in all the domains is not displayed. If you use the -v option, the primary MAC address and serial number of components are displayed.
`showenvironment [-v] {[sscn][psn][sn]}`	Displays the status of the environmental sensors in the various components of the chassis. For example, this command tells you the internal temperatures of the components, the speeds of their fans, and the level of current on their supply rails.
`showfru {sscn|sn|ch|psn}`	Displays the contents of a specified component's (or of all components') FRUID database. Each component maintains extensive information about itself. This includes static data (for example, hardware version information) and dynamic data (for example, recent event messages generated by the component).
`showdate`	Displays the current date and time (in UTC format) according to the System Controller.
`showlogs [-b]|[-e] [-g] [-v] {sscn|sn}`	Displays the events that have been logged for a specified blade, switch, or System Controller. Specify `-b` to view the first *n* events, `-e` to view the last *n* events, `-g` to specify the number of lines of output you want to view before a pause in the display, and `-v` to view all events in the log.
`showlocator`	Tells you whether the locator LED is on or off.

TABLE D-5 Commands for Monitoring the Chassis and its Components

Command and Option (if any)	Effect of the Command
`consolehistory [-b]|[-e] [-g] [boot|run] sscn/swt|sn`	Displays the contents of the switch or blade console's boot- or run-time buffer. Specify `-b` to view the first *n* lines of information, `-e` to view the last *n* lines, `-g` to specify the number of lines of output you want to view before a pause in the display.
`showusers`	Shows the users currently logged into the System Controller.
`usershow [`*username*`]`	Shows details of the specified user's login account. If no user is specified, the command shows details of all user accounts. The output indicates users' permissions and whether they have a password assigned or not.

D.6 System Controller Configuration Commands

TABLE D-6 Commands for Configuring the System Controller

Command and Option (if any)	Effect of the Command
`setupsc`	Enables you to configure the active System Controller interactively. (There is no non-interactive method available.) The standby System Controller automatically uses the same configuration as the active.
`flashupdate [-v] [-F] -s` *IP address* `-f` *path* `sscn|sn`	Enables you to upgrade new firmware to a System Controller or to a server blade. *IP address* is the IP address of the TFTP server on which the firmware is stored. *Path* is the location of the firmware on the TFTP server. The `-v` option displays information about the upgrade process as it takes place. By default, the System Controller waits until outstanding commands (for example from different telnet sessions) involving a specified blade have finished executing before it initiates a flashupdate. However, the `-F` option forces the flashupdate to proceed even if there are System Controller commands involving the specified blade that have not finished executing.
`setfailover [-F]`	Tells you which System Controller is the active and which the standby System Controller. It also prompts you to confirm that you want to force the current standby System Controller to take over from the active one. If you have only used the command to find out which System Controller is active, just answer no. The `-F` option causes the failover process to be initiated even if there are commands from other users that have not finished executing (by default, the System Controller waits until there are no outstanding commands before it initiates a failover).

TABLE D-6 Commands for Configuring the System Controller

Command and Option (if any)	Effect of the Command
`setdefaults [-y]`	Returns the active System Controller (but not its switch) to the factory default settings. The -y option causes the the SSC to revert to the factory default settings without issuing a confirmation prompt.
`setdate [`*mmdd*`]`*HHMM*`[.`*SS*`]` | *mmddHHMM*`[`*cc*`]`*yy*`[.`*SS*`]`	Enables you to set the time of day on the System Controller, switches, and any currently inserted server blades. When you set the date and time, you must use Co-ordinated Universal Time (UTC). The Solaris server blades will work out the local time for your time-zone by using an offset from UTC. And the blades discover UTC from the System Controller. The variables are as follows: *mm* is the month (two digits) *dd* is the day (two digits) *HH* is the hour (two digits) *MM* is the minutes (two digits) *SS* is seconds (two digits)
`setlocator on|off`	Turns the chassis locator LED on and off.

D.7 Commands Relating to the Switches and Blades

Note – Whenever you are at a switch or blade console, type **#.** to return to the active System Controller's `sc>` prompt.

TABLE D-7 Commands for Accessing and Configuring the Switches and Blades

Command and Option (if any)	Effect of the Command
`console [-f]|[[-r] sscn/swt|sn`	Access the console of a switch or blade. Use the `-f` command to force into "read-only" mode any other user who is currently logged in. Use the `-r` command to log in yourself using "read-only" mode.
`consolehistory [-b]|[-e] [-g] [boot|run] sscn/sc|sscn/swt|sn`	Displays the contents of the specified System Controller, switch, or blade console's boot- or run-time buffer. Specify `-b` to view the first *n* lines of information, `-e` to view the last *n* lines, `-g` to view a specified number of lines of information before a pause in the display.
`bootmode reset_nvram|diag|skip_diag|normal|bootscript="string" sn {sn}`	This command allows you to specify a boot mode for a blade. You need to use it to boot Linux blades for the first time from a PXE environment (see Chapter 4). For more information, refer to the *Sun Fire B1600 Chassis Administration Guide*.
`flashupdate -s` *IP address* `-f` *path* `[-v] sscn/sc|sn`	Enables you to upgrade new firmware to the active System Controller or to a server blade. *IP address* is the IP address of the TFTP server on which the firmware is stored. *Path* is the location of the firmware on the TFTP server. The `-v` option displays information about the upgrade process as it takes place.

D.8 Commands for Administering User Accounts

TABLE D-8 Commands for Administering User Accounts

Command and Option (if any)	Effect of the Command
`useradd` *username*	Adds a named user to the list of permitted System Controller users.
`userdel` *username*	Deletes a user from the list of permitted System Controller users.
`userpassword` *username*	This command allows a user with `a`-level permissions to alter another user's password.
`password`	This command allows a user to change his or her own password (in other words, to change the password of the user that he or she is currently logged in as).
`userperm` *username* [a][u][c][r]	This command specifies the named user's permission levels. `c` gives console access to blades and switches: `a` gives administration privileges (enabling the named user to change the configuration of the System Controller), `u` gives user administration privileges (enabling the named user to administer user accounts), and `r` gives reset permissions (enabling the named user to reset components of the chassis or to power them on and off).
`usershow` [*username*]	Shows details of the specified user's login account. If no user is specified, the command shows details of all user accounts. The output indicates users' permissions and whether they have a password assigned or not.
`showusers`	Lists all users currently logged into the System Controller.

APPENDIX **E**

The Active and Standby System Controllers

This appendix provides a detailed explanation of the relationship between the chassis's active and standby System Controllers (if two SSCs are installed). It also describes the limitations of this relationship.

E.1 The Events That Cause a Failover

The blade system chassis contains two System Controllers. Only one of these is active at a given time, therefore only one can be accessed by means of the ALOM command-line interface. However, even though the other System Controller is quiesced (in other words, is in standby mode) its associated switch remains active, and the standby System Controller is also able to take over as the active System Controller in the event of:

- the removal of the currently active System Controller,
- a major failure of the System Controller software application on the active System Controller, or a hardware fatal error,
- an execution of the `setfailover` command by the user to force the System Controllers to swap roles.

E.2 The Activities of the Standby System Controller

The standby System Controller performs the following activities despite its main software application being in a quiesced state:

- Monitors the health of the currently active System Controller and takes over if that System Controller is physically removed, if a major failure of its main software application occurs, if a hardware fatal error occurs, or in response to the use of the `setfailover` command on the active System Controller.
- Receives the configuration parameters that the user enters for the `setupsc` command on the active System Controller. (This enables it to take over transparently as the active System Controller.)
- Receives all event messages so that event logs on the standby System Controller are always up to date.
- Permits console access from the active System Controller to the switch in the SSC module containing the standby System Controller. (Note that, if the booting of the standby System Controller is interrupted for any reason, the standby System Controller cannot provide console access to its associated switch).

- Helps maintain the integrity of the user login and host ID information for the chassis as a whole. (The host ID information is required for the server blades; the user login information is required for the System Controllers.) These two sets of information are stored mainly on the midplane. However, the two System Controllers are involved in their preservation.

  In the case where a new SSC (in its factory default state) is introduced into a chassis that is already in use, the new SSC simply inherits the user login and host ID information that is currently stored on the midplane.

  In the reverse case, where the chassis is new (and its user login and host ID information are therefore unconfigured) but the SSC has been previously in use, the midplane takes the user login and host ID information from the System Controller.

  However, in the case where an SSC is introduced into a chassis and both already contain user login and host ID information but the SSC and chassis differ in respect of either or both the outcome is more complicated to predict. In this case the standby System Controller, if it is available, plays an arbitrating role. It compares its own user login and host ID information with the information held on the SSC containing the active System Controller and with the information held on the midplane. If its own host ID information agrees with that stored on either the active SSC or the midplane, then that information prevails. Similarly if its own user login information agrees with that stored on either the active SSC or the midplane, then that information prevails. For each set of information, if the standby System Controller finds that its own data differs from that of both the active SSC and the midplane, then the data in the midplane prevails.

E.3 Limitations of the Failover Relationship Between the Two System Controllers

There is no impact on the running of the server blades or switches during the failover process. However, you need to be aware that:

- When one System Controller takes over from the other the chassis is temporarily (for approximately 15 seconds) without an active System Controller. (This is because both System Controllers are reset as part of the failover process.) In consequence there will be no console logs gathered for the period during the failover, and when you log into the new active System Controller all the event logs on both System Controllers will be empty.
- During the failover process, no user management of any of the chassis's components is possible via the System Controllers. It is, however, still possible to telnet into the switches or blades, and it is still possible to use the switch's web-based graphical user interface.
- During the failover process, it is not possible to perform any upgrades of the firmware on the components of the chassis.
- To upgrade System Controller firmware you must make the System Controller that you want to upgrade into the standby one (if it is not currently the standby one). To do this use the `setfailover` command at the `sc>` prompt on the currently active System Controller.
- There is no access permitted via telnet to the standby System Controller. Use the alias IP address instead. However, you need to be aware that telnet connections are dropped when failover takes place from one System Controller to another.

Index

T

U

V

W