Reference Guide
SF6090 Switch Fabric Blade

Clavister AB
Sjögatan 6J
SE-89160 Örnsköldsvik
SWEDEN

Phone: +46-660-299200
Fax: +46-660-12250

www.clavister.com

Published 2010-01-27

Reference Guide
SF6090 Switch Fabric Blade

Published 2010-01-27

Table of Contents

List of Figures

Preface

Target Audience

The target audience for this guide is the user who has taken delivery of a packaged Clavister SF6090 Switch Fabric Blade and going through the installation phase. The guide takes the user from unpacking and installation to power-up and initial network connection.

Notes to the Main Text

Special sections of text which the reader should pay special attention to are indicated by icons on the left hand side of the page followed by a short paragraph in italicized text. There are the following types of such sections:

Note
This indicates some piece of information that is an addition to the preceding text. It may concern something that is being emphasised or something that is not obvious or explicitly stated in the preceding text.

Tip
This indicates a piece of non-critical information that is useful to know in certain situations but is not essential reading.

Caution
This indicates where the reader should be careful with their actions as an undesirable situation may result if care is not exercised.

Important
This is an essential point that the reader should read and understand.

Warning
This is essential reading for the user as they should be aware that a serious situation may result if certain actions are taken or not taken.

Chapter 1: Overview

The Switch Fabric Blade (SFB) supports inter-node connectivity by providing Base and Fabric Ethernet switching and system clock synchronization. The SFB can act as a Shelf Manager and also includes a site for a COM Express processing module, which can be used for system management.

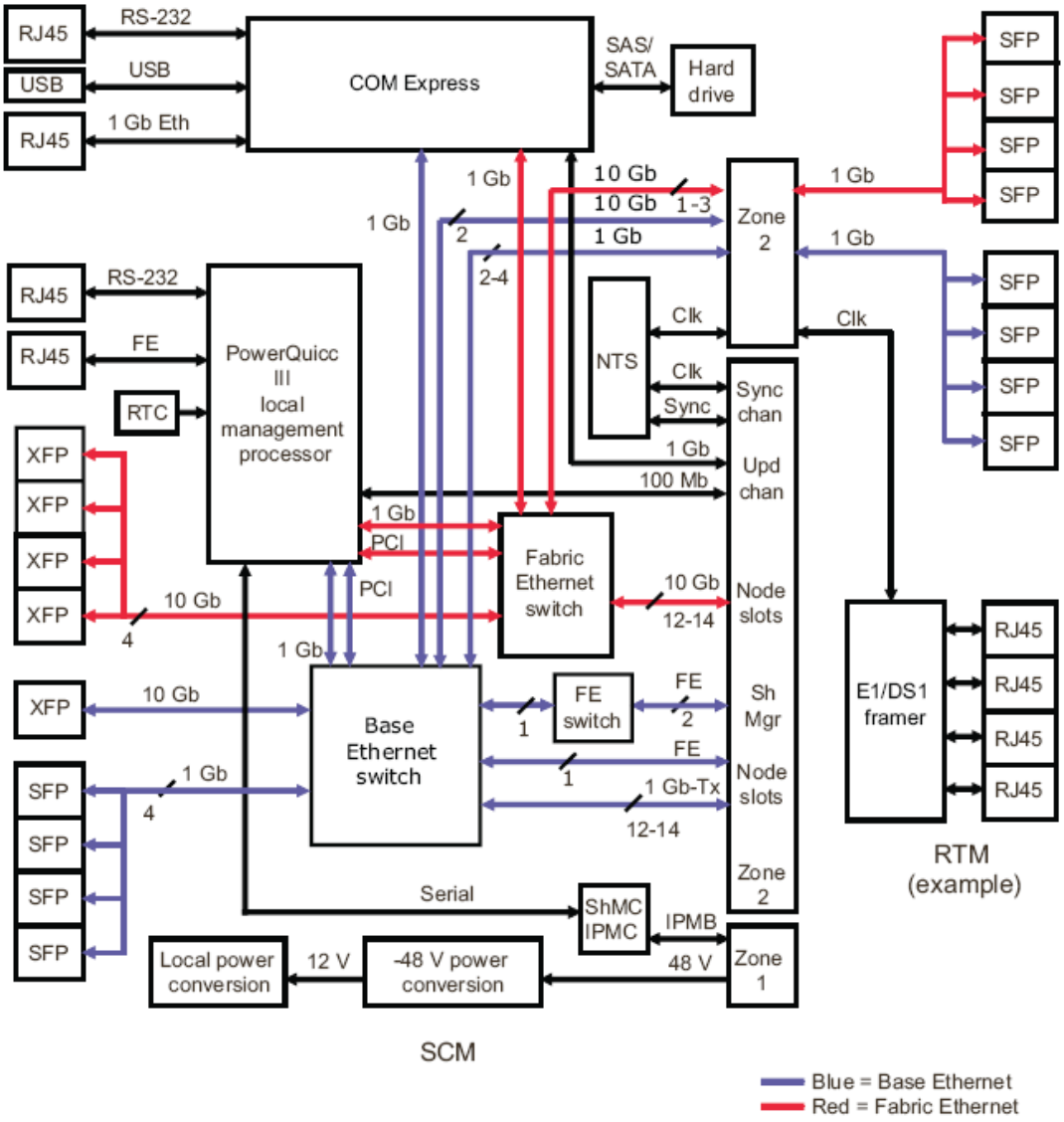


Figure 1.1. Switch Fabric Blade Block Diagram

The Switch Fabric Blade is designed for applications in third generation (3G) wireless and wire-line infrastructures since it provides highly integrated, centralized common equipment functions—switching, chassis management, and network timing.

The SFB as a Shelf Manager

The SFB is configured to act as the *Shelf Manager* by default (the word "shelf" in this manual is synonymous with the word "chassis"). Management duties are divided between the local management processor (LMP) and the Intelligent Platform Management Controller (IPMC). The IPMC on the SFB is equivalent to a Shelf Management Controller (ShMC).

Figure 1.2. The SF6090 Switch Fabric Blade

The Shelf Manager is responsible for monitoring and controlling the components within the chassis. The ShMC communicates with the IPMCs on the chassis components.

Redundancy

Typically in a high availability system like the Switch Fabric Blade, two SFBs are installed in the platform for redundancy purposes. Redundancy provides a means of continuing functionality even if a failure occurs on one of the SFBs:

- Each SFB contains switches for both the Base Ethernet and the Fabric Ethernet interfaces. If either SFB fails or is removed, the other SFB switches all the traffic for both the Base and the Fabric interface. The SFBs operate on an active/active basis, which means the switches in both SFBs are always operating.

- Both SFBs run the same switch configuration and management software. If one SFB fails or is removed, the copy of the configuration and management software on the remaining SFB continues to control the switches.

- Each SFB has a Shelf Manager. If one SFB fails or is removed, the Shelf Manager on the remaining SFB operates as the active Shelf Manager. The dual Shelf Managers operate on an active/standby basis, which means only one Shelf Manager is active at any time.

- SFBs act and operate independently from each other and, in most cases, need to be configured separately. However, some management settings do not require a separate configuration on each SFB. These chassis-specific configuration changes are only performed on the SFB containing the active Shelf Manager.

Note
The SFBs must be of the same model.

Local Management Processor

The functions on the SFB are managed by a powerful on-board LMP block, based on a PowerQUICC III processor. The LMP manages the Ethernet switches and the network timing subsystem (NTS) and provides access to the hardware management subsystem by way of the ShMC. The LMP may also function as the Shelf Manager, depending on the software configuration.

The LMP has four Ethernet ports, two PCI buses, two serial ports, a local inter-integrated circuit (I2C) bus, a SPI port, DRAM and flash memory. A 10/100 Ethernet channel from the PowerQUICC III is routed to the front panel to provide an interface for switch management. The memory bus provides interfaces to the DRAM and flash memory. The PCI buses provide the interfaces to the Base Ethernet and the Fabric Ethernet switches.

Intelligent Platform Management

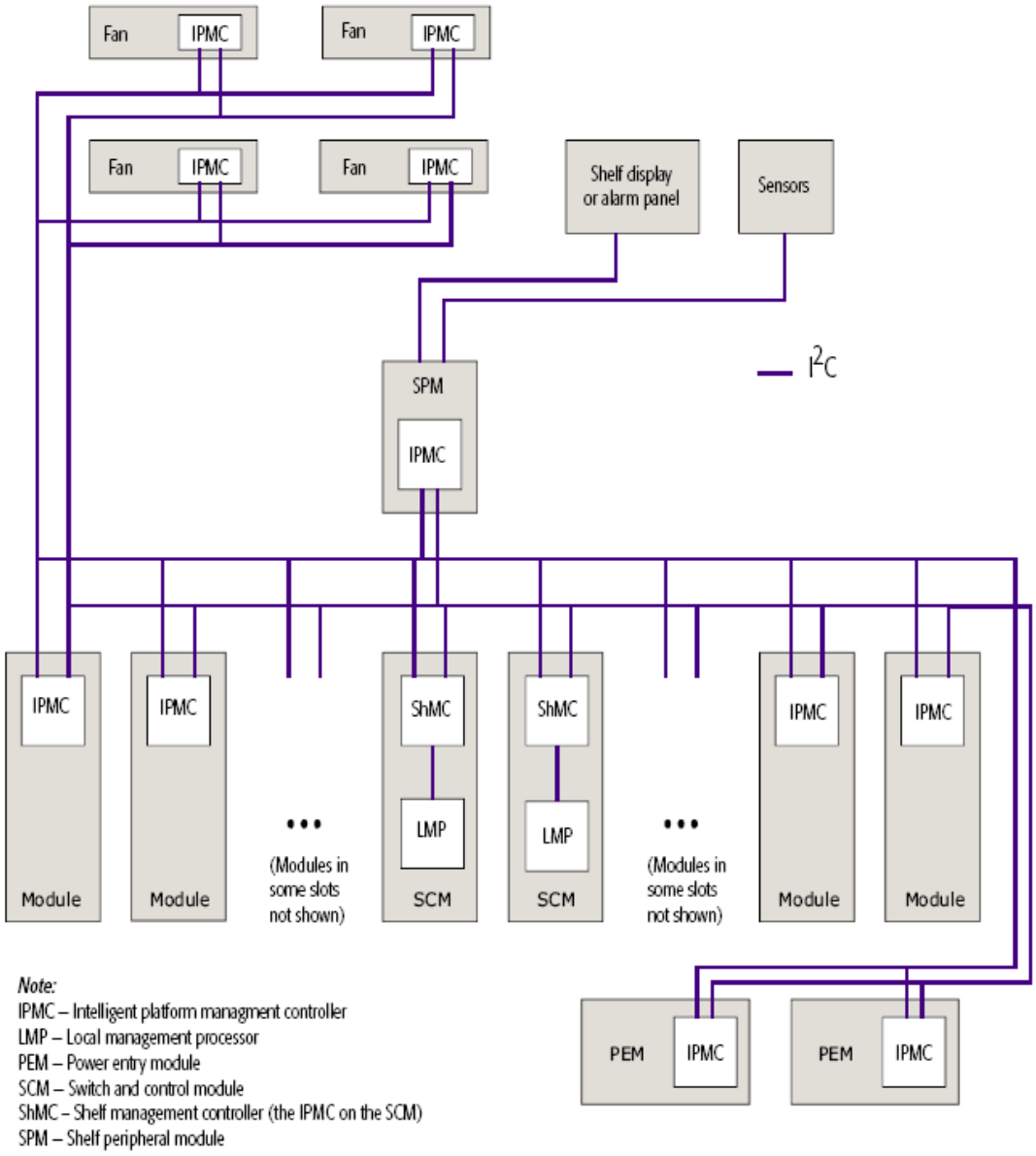
Intelligent Platform Management (IPM) is a subset of chassis management. IPM handles aspects of chassis management that involve communication between the Shelf Manager and the IPMCs on the FRUs.

A Renesas H8S/2166 micro-controller is used for the IPMC on the SFB. This device manages commands and data as part of the hardware management subsystem, which includes the IPMCs and the sensors from installed modules and field replaceable units (FRUs) and their communication with the Shelf Manager.

The Shelf Manager retrieves FRU information from the IPMCs and stores the information. The capabilities, sensor readings, and hot-swap states are among the possible types of information that can be retrieved from a FRU. To provide redundancy, the active Shelf Manager also sends any collected FRU information to the inactive Shelf Manager, which also stores the FRU information.

The Shelf Manager and all the IPMCs communicate through the Intelligent Platform Management Bus (IPMB). This communication uses the Intelligent Platform Management Interface (IPMI) protocol. To implement full redundancy, the IPMB is set up as two I2C buses. This dual-bus architecture prevents chassis management from being interrupted by the failure of either bus. Each ShMC and each IPMC connects to both I2C buses, so if one bus becomes unavailable, communication takes place on the other bus. To prevent a failed ShMC or an IPMC from significantly disrupting either or both buses, each ShMC and IPMC has associated circuitry that isolates the ShMC or the IPMC from both buses if its associated watchdog timer fails to be triggered periodically.

The illustration below shows an example IPMB architecture in a Clavister chassis. The abbreviation SFB refers to the Switch Fabric Blade.



Ethernet Interfaces

The SFB has two types of Ethernet interfaces:

- A Base Ethernet interface, which consists of a single Broadcom BCM56304 1 Gigabit Ethernet switch.
- A Fabric Ethernet interface, which consists of a single Broadcom BCM56800 Ethernet switch. The switch has twenty ports that can be configured for 1Gbps or 10 Gb operation.

An overview of the topology and the connections of the Ethernet interfaces is provided below.

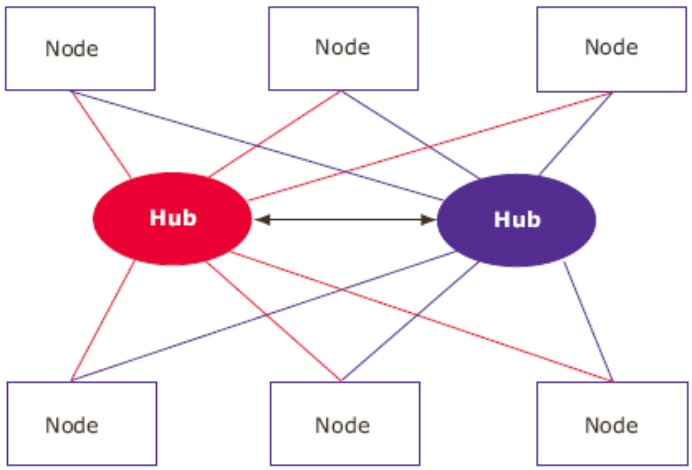
Base Ethernet Interface

The SFB provides a Base Ethernet interface that supports the 10/100/1000Base-T Ethernet standard on the backplane. Uplinks for the Base Ethernet are also available through the front panel.

The SFB provides a Base Ethernet interface that supports the 10/100/1000Base-T Ethernet

standard on the backplane. Uplinks for the Base Ethernet are also available through the front panel.

When two SFBs are installed in the hub slots on an ATCA platform, the topology of the Base Ethernet forms a dual star. The diagram below illustrates this topology, showing the relationship between the node and the hub modules.



Each Ethernet backplane link consists of four differential signal pairs. Each SFB is installed in a hub slot and acts as a switch that connects to every node (non-SFB) module to form the star topology.

This redundant topology provides an alternate path if one Ethernet path fails, or if one SFB fails. The SFBs connect to each other through a direct Ethernet connection (Base channel 2), which the high-level chassis-management software uses. Traffic through the SFB-to-SFB link is controlled by the Base Ethernet switch, which can also pass user traffic.

Fabric Ethernet Interface

The SFB can support either 1 Gbps or 10 Gb Ethernet links for the Fabric interconnect, depending on the node module connections. The full Fabric interface channel is implemented using a 10 Gb XAUI configuration as described in the PICMG 3.1 specification.

The Fabric Ethernet interface also implements a dual star topology similar to the topology of the Base Ethernet interface. However, with the Fabric Ethernet interface there is no direct switch-to-switch link.

Network Timing Subsystem

The network timing subsystem (NTS) consists of a circuit block and software. The circuit block provides a centralized timing source for the chassis based on the AdvancedTCA synchronization clock Interface specification. The NTS selects between several internal and external timing references, and provides a phase-locked, highly stable system clock to the platform modules managed by the SFB's Shelf Manager and the NTS software.

COM Express and Drive

The SFB includes sites for a COM Express module and a SAS or SATA drive. The supported COM Express module and the drive, which acts as a peripheral to the COM Express, are available as a build option for the SFB.

When the COM Express module and the SAS/SATA drive are implemented on the SFB, they offer processing and storage capabilities well-suited for radio network controller (RNC), media

gateway (MGW), information management system (IMS), and Internet protocol television (IPTV) applications.

Installed Software

The SFB's Linux-based software includes many features and several management interfaces, including Ethernet switching protocol support, a PICMG-compliant Shelf Manager, a command line interface (CLI), application programming interfaces (APIs), and an SNMP agent.

External Connectivity

The SFB provides connectivity for communicating with external equipment through three interfaces:

- **Front panel interfaces** — Front panel interfaces are accessed through each SFB's faceplate. The connection types are labeled on the faceplate.
- **Backplane interfaces** — The backplane is the primary electrical interconnection between the modules in a platform. The Zone 2 connectors of the backplane provide connectivity for the Base, the Fabric, the update channel interfaces, and synchronization clocks.
- **Rear transition module interfaces** — When installed in a Clavister chassis, each SFB has an associated rear transition module (RTM) site, which can provide external access to SFB's signals and connections.

Chapter 2: Subsystems and Components

The Switch Fabric Blade consists of a series of subsystems that work together to provide the network elements required for third generation wireless and wire-line infrastructures. This chapter covers the details of the subsystems that make up the SFB.

Local Management Processor

The local management processor (LMP) is used to manage the Ethernet switches, to manage the network timing subsystem (NTS), and to provide access to the hardware management subsystem.

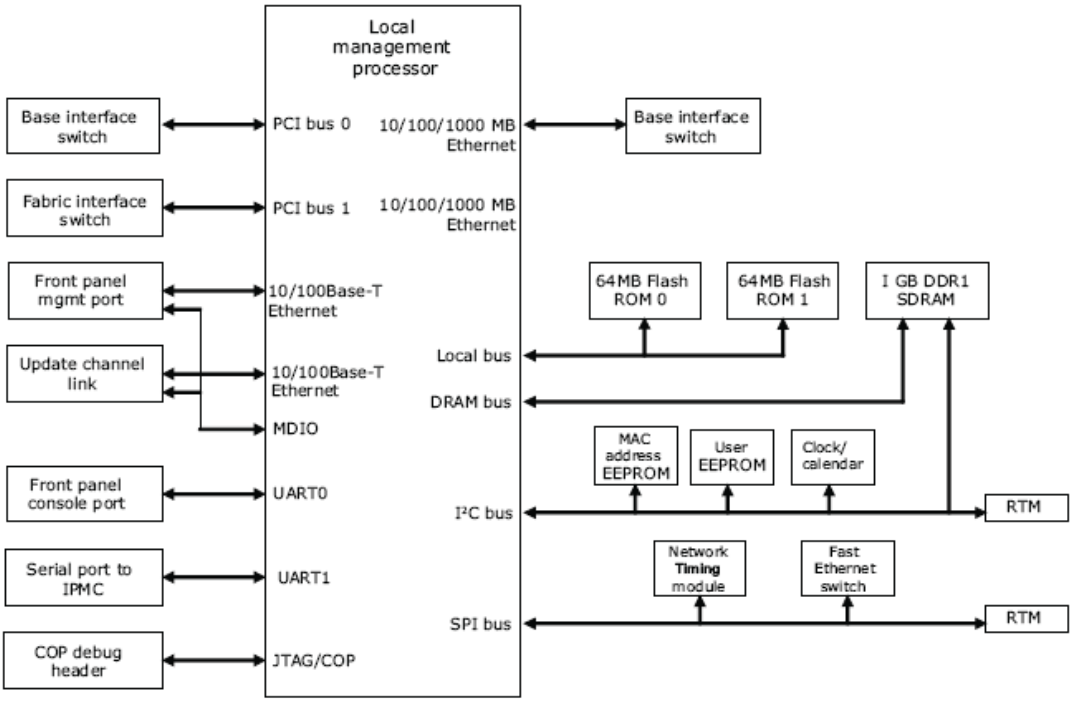
The LMP provides Ethernet ports, serial communications, an inter-integrated circuit (I2C) controller, a serial peripheral interface (SPI), memory, and peripheral component interconnect (PCI) bus interfaces. Two 1 Gigabit Ethernet ports provide high-speed interfaces to the Base switch and the Fabric switch. A 10/100 Mb Ethernet channel from the LMP is routed to the front panel and provides an interface for switch management. A 10/100 Mb Ethernet channel is routed across the backplane to the redundant SFB. The I2C controller and the SPI buses control on-board peripheral devices. The memory buses provide interfaces to the local synchronous dynamic random access memory (SDRAM), and the boot flash memories. The PCI buses connect to the management ports of the Base and the Fabric Ethernet switches.

The LMP includes the following components:

- A PowerQUICC III processor chip with 833 MHz core frequency.
- A 200-pin SODIMM memory connector for connecting to a memory module with a capacity of 1 Gb DDR SDRAM.
- A redundant pair of 64 MB Spansion flash memory components (128MB total) for PowerQUICC III configuration, Linux operating system boot image, and file system.
- Clock generation circuitry.
- Power good generation circuitry.
- Two 10/100 Fast Ethernet interfaces to a front panel connector and the update channel.
- A serial interface to a front panel connector.
- A serial bus interface to the Intelligent Platform Management Controller (IPMC).
- An I2C bus interface to the serial EEPROMs, the clock/calendar, and the rear transition module (RTM)

- A SPI bus to the NTS, the Fast Ethernet switch, and the RTM.
- Two peripheral component interconnect (PCI) bus configuration interfaces to the Base and the Fabric Ethernet switches.
- 1 Gb Ethernet interface to the Base switch.
- 1 Gb Ethernet interface to the Fabric switch.

The diagram below shows the relationship between the LMP and the SFB.



The PowerQUICC III Processor

The LMP is based on a PowerQUICC III processor. The PowerQUICC III LMP is packaged in a 783-pin ball grid array and is highly integrated with an embedded e500 core; integrated instruction and data caches, a system interface unit, and an integrated reduced instruction set computer-based (RISC) communications processor.

The PowerQUICC III LMP has the following characteristics:

- **Processor:** MPC8541
- **Core Freq. (MHz):** 833
- **PCI Bus Freq. (MHz):** 66
- **SYSCLK Freq. (MHz):** 66
- **VDD/Core:** 3.3V/1.2V
- **I Cache (Kbytes):** 32
- **D Cache (Kbytes):** 32
- **Processor Version:** 0x0080

The PowerQUICC III processor chip incorporates the following elements:

- CPU with e5000 core that implements Book E 32-bit architecture.
- 256KB on-chip memory.
- DDR memory controller.
- Programmable interrupt controller (PIC) compliant with OpenPIC architecture.
- Local Bus Controller (LBC), which connects to the two 64MB flash components.
- Two PCI bus controllers used to configure the Base and the Fabric switches.
- I2C controller providing access to the SDRAM module, two serial EEPROMs, the clock/calendar, and devices on the RTM.
- Two UART interfaces for:
    1. Serial console port routed through front panel.
    2. Serial port providing link between the PowerQUICC III and the on-board IPMC.
- Two 10/100Base-T Ethernet interfaces (Fast Ethernet) used for:
    1. Management and maintenance purposes.
    2. Connecting to the redundant SFB through the update channel.
- Two 10/100/1000 interfaces configured as ten-bit interfaces (TBI) and linked to a serializer/deserializer (SerDes). The interfaces are connected to the Base Ethernet switch and Fabric Ethernet switch as alternate ports to the PCI bus.
- Serial management data input/output (MDIO) port for managing the physical layer devices (PHYs) on the 10/100Base-T interfaces.
- SPI controller providing a four-wire serial bus for accessing the NTS, the 10/100Base-T Ethernet switch, and devices on the RTM.
- EEPROM used to store media access control (MAC) addresses for Ethernet ports.
- EEPROM used for storing user-defined information.
- Clock/calendar to keep a running time and date. Includes lithium battery to maintain timekeeping when power is not available to the SFB.
- General purpose input/output (I/O) signals to and from various functions on the SFB for control and status.
- Synchronous dynamic random access memory (SDRAM) interface for the small outline dual in-line memory module (SODIMM).
- Two Spansion flash memory devices used as a redundant set of 64MB flash stores.
- Reads I/O bus information on the SFB version and revision history.
- Integrated reset logic.

Base Ethernet Switch Subsystem

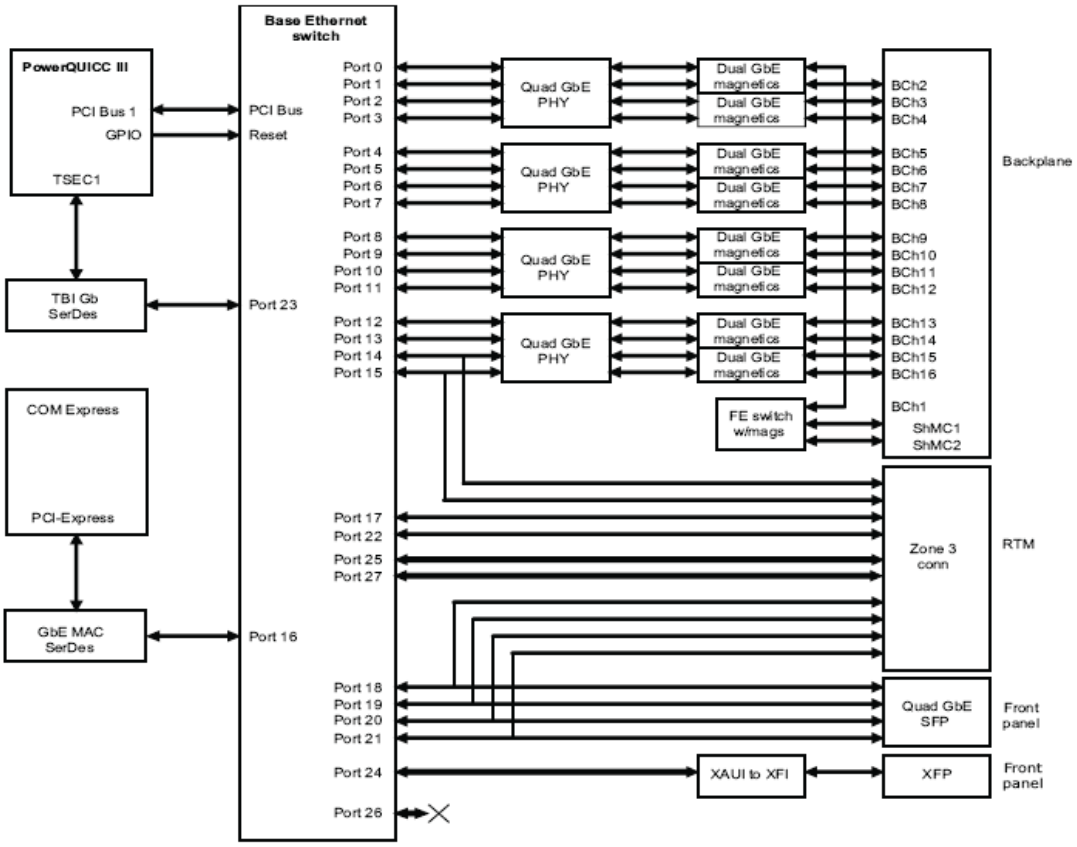
One of the primary functions of the SFB is to provide the Ethernet switch function for ATCA

platforms. This is implemented using a Broadcom 28-port Gigabit switch. Twenty-four of the ports are 1 Gb and four of the ports are 10 Gb.

The key features of the 28-port Gigabit Ethernet switch include the following:

- Twenty-four 10/100/1000Mb ports.
- Four 10 Gb ports.
- Ethernet switch/router with integrated MACs, packet buffer engine and switching engine.
- Supports line-rate Layer 2 switching and Layer 2 multicast for all packet sizes and conditions.
- Line-rate switching for all packet sizes and conditions.
- Eight layer Quality of Service (QoS) on a per-port basis.
- Supports multiple (256) Spanning Trees (IEEE 802.1D).
- Supports Flow Control (IEEE 802.3x).
- VLAN Support based on MAC, port, subnet, and protocol (IEEE 802.1Q).
- Supports link aggregation (up to 8 ports - IEEE 802.3ad).
- IPSec engine supports DES, 3DES, HMAC-SHA-1, and HMAC-MD5, and is compatible with IPSec, SSL, TLS, 802.1x, and 802.11i.
- Head-of-line blocking prevention.
- Per-port packet rate (storm) control.
- Supports port mirroring.
- PCI interface to PowerQUICC III host processor.
- SGMII interface support to external 1000 Mb PHYs.
- Integrated SerDes.

The diagram below shows how the Base Ethernet switch communicates with the other components on the SFB.



Configuration

The switch has built in media access controllers (MACs) for each port that interface to the external physical layer devices (PHYs).

The 1 Gigabit Ethernet ports can each be configured to operate in:

- Serial Gigabit Media Independent Interface (SGMII) mode, which has Gigabit Ethernet ports going to the backplane, front panel RJ-45 ports, and RTM.
- Serializer/deserializer (SerDes) modes, which support Gigabit Ethernet ports going to optical SFP modules, 10/100/1000 Mbit ports going to copper SFP modules, or internal links.
- One port to the COM Express site as 1 Gb SerDes.

The 10 Gigabit Ethernet ports are configured to operate in 10 Gigabit attachment unit interface (XAUI) mode, with ports routed to Zone 3 and the front panel XFP ports.

Base Ethernet Port Mapping

Port #	PHY/Port	MDIO interface address	Port destination	IPMI hardware address	Logical slot
0	0 / 0	0x00	FE Switch (Shelf Manager)	N/A	N/A
1	0 / 1	0x01	Base channel 2 (link to redundant SCM)	0x41 / 42	1 / 2
2	0 / 2	0x02	Base channel 3	0x43	3
3	0 / 3	0x03	Base channel 4	0x44	4
4	1 / 2	0x06	Base channel 5	0x45	5
5	1 / 3	0x07	Base channel 6	0x46	6
6	1 / 0	0x04	Base channel 7	0x47	7
7	1 / 1	0x05	Base channel 8	0x48	8
8	2 / 2	0x0A	Base channel 9	0x49	9
9	2 / 3	0x0B	Base channel 10	0x4A	10
10	2 / 0	0x08	Base channel 11	0x4B	11
11	2 / 1	0x09	Base channel 12	0x4C	12
12	3 / 2	0x0E	Base channel 13	0x4D	13
13	3 / 3	0x0F	Base channel 14	0x4E	14
14	SFP interface to RTM	0x0C or N/A	SFP3 on the RTM	N/A	N/A
15	SFP interface to RTM	0x0D or N/A	SFP4 on the RTM	N/A	N/A
16	SerDes	N/A	COM Express	N/A	N/A
17	SFP interface to RTM	N/A	SFP1 on the RTM	N/A	N/A
18	SFP interface	N/A	Front panel 1/6: 1G optical or 10/100 Mbit copper	N/A	N/A
19	SFP interface	N/A	Front panel 1/7: 1G optical or 10/100 Mbit copper	N/A	N/A
20	SFP interface	N/A	Front panel 1/8: 1G optical or 10/100 Mbit copper	N/A	N/A
21	SFP interface	N/A	Front panel 1/9: 1G optical or 10/100 Mbit copper	N/A	N/A
22	SFP interface to RTM	N/A	SFP2 on the RTM	N/A	N/A
23	SerDes	N/A	PowerQUICC III	N/A	N/A
X0	XFP interface	0x10	Front panel 1/5: XFP	N/A	N/A
X1	XFP interface	TBD	RTM XAUI	N/A	N/A
X2	XFP interface	N/A	Not used	N/A	N/A
X3	XFP interface	TBD	RTM XAUI	N/A	N/A

10/100/1000Base-T Gigabit Ethernet Ports

Sixteen Gigabit Ethernet PHYs provide 10/100/1000Base-T connectivity to the backplane and the Fast Ethernet (FE) switch. All of the 1000Base-T ports use a Gigabit PHY channel along with Gigabit Ethernet transformers to couple to the backplane.

- 12 ports to backplane node slots, depending on the board configuration.
- One port to an FE switch, allowing two FE connections to an external Shelf Manager (if present) through Base channel 1.
- One port to the Base interface of a redundant SFB (if present) through Base channel 2.

SerDes Gigabit Ethernet Ports

Six ports are configured as serializer/deserializer (SerDes) ports. Two of the SerDes ports go to the on-board LMP and the COM Express site. Four of the SerDes ports are directed to SFP sockets on the SFB front panel.

10 Gigabit Ethernet Ports

Three 10 Gb Base Ethernet ports provide external access on the front and the rear faceplates.

- 1 XFP port on the front panel, XFI signaling converted from XAUI.
- 2 XAUI signaling to RTM.

The XFP 10 Gb connector uses XFI signaling, composed of single transmit and receive 10 Gb differential pairs. A XAUI-to-XFI interface chip converts the signaling from the XAUI signals (connected to the switch) to XFI signals (connected to the XFP port) at the front panel.

Fast Ethernet Switch

A 5-port 10/100Base-T Ethernet switch connects one Base interface port to one or two off-board Shelf Managers through Base channel 1.

The Fast Ethernet (FE) switch is managed by the PowerQUICC III using the SPI port.

Port #	Port destination	Port states
1	Base interface	Port 1 should always be enabled.
2	Base channel 1a	Ports 2 and 3 may be enabled or disabled (together), depending on configuration for Base connectivity to an off-board Shelf Manager.
3	Base channel 1b	
4	Not used	Ports 4 and 5 are not used, and should be disabled.
5	Not used	

Port Status LEDs

The following table describes the LEDs specific to the Base Ethernet.

Port	LED and possible states
1 Gb optical or 10/100 Mbit copper SFP ports Each Base Ethernet SFP port on the front panel or on the RTM has two LEDs	1 green link status LED ▪ Green – link established ▪ Blinking green – link activity ▪ Off – link fail or port is disabled
	1 green port status LED ▪ Green – port is enabled ▪ Off – port is disabled
10 Gb XFP ports Each Base Ethernet XFP port on the front panel or on the RTM has two associated LEDs.	1 green link/activity status LED ▪ Green – link established ▪ Blinking green – link activity ▪ Off – link fail or port is disabled
	1 green port status LED ▪ Green – port is enabled ▪ Off – port is disabled

Fabric Ethernet Switch Subsystem

The SFB provides a 10 Gb Fabric Ethernet connection to 12 or 14 node slots (depending on the board configuration) based on AdvancedTCA 3.1. Four ports of the Fabric interface are also available on the front panel and three ports are routed to the RTM through the Zone 3 area.

The Fabric interface is implemented using a Broadcom 20-port 10 Gb Ethernet switch. The backplane ports use XAUI, made up of four lanes (Tx and Rx differential pairs) operating at 3.125 Gb. The front panel uses XFI on an XFP connector, converted from the XAUI signaling on the switch ports by a XAUI-to-XFI transceiver.

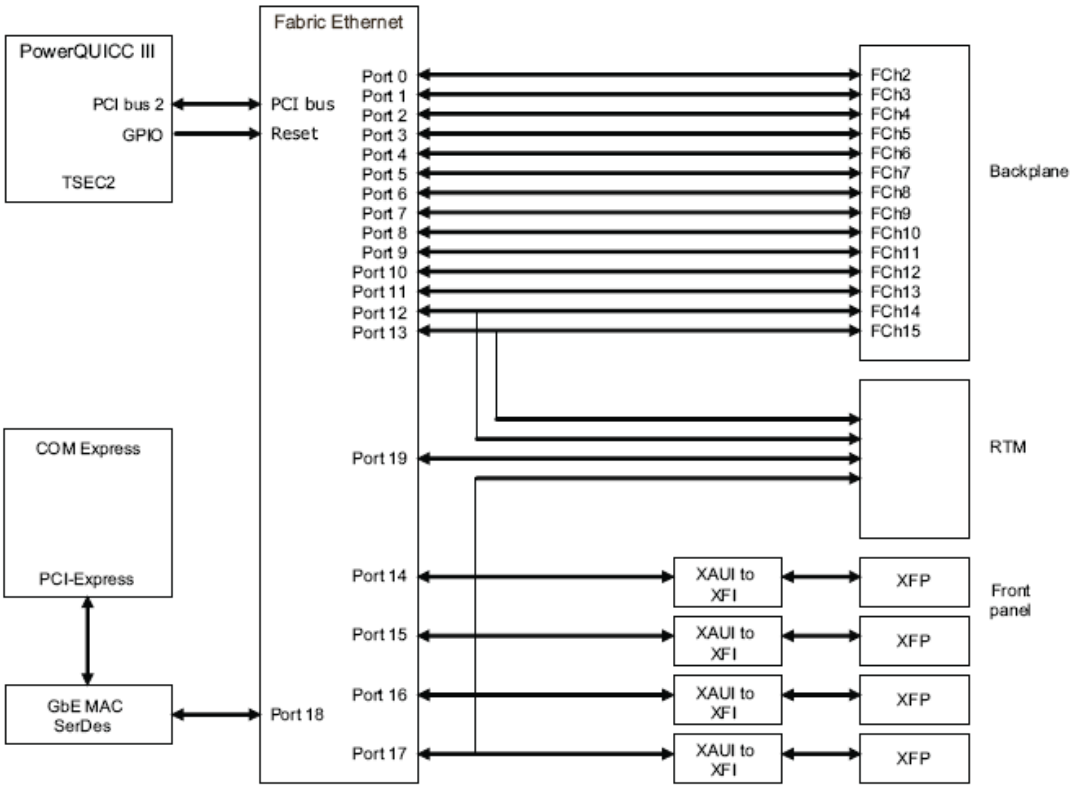
The Fabric Interface connects:

- Node slots as 10 Gb XAUI or 1 Gb SerDes.
- Front panel XFP connector ports as 10 Gb XFI.

- 1 or 3 ports (depending on the board configuration) to the RTM as 10 Gb XAUI or 1 Gb SerDes.
- One port to the COM Express site as 1 Gb SerDes.
- The LMP on PCI Bus 2.

Note that there is no direct Base-to-Fabric connection on an SFB and no Fabric-to-Fabric connection between the two SFBs.

Below is a block diagram showing how the Fabric Ethernet switch communicates with the rest of the SFB.



Fabric Port Mapping

Port #	PHY/Port	MDIO interface address	Port destination	IPMI hardware address	Logical slot
0	XFP interface	N/A	Fabric channel 2	0x43	3
1	XFP interface	N/A	Fabric channel 3	0x44	4
2	XFP interface	N/A	Fabric channel 4	0x45	5
3	XFP interface	N/A	Fabric channel 5	0x46	6
4	XFP interface	N/A	Fabric channel 6	0x47	7
5	XFP interface	N/A	Fabric channel 7	0x48	8
6	XFP interface	N/A	Fabric channel 8	0x49	9
7	XFP interface	N/A	Fabric channel 9	0x4A	10
8	XFP interface	N/A	Fabric channel 10	0x4B	11
9	XFP interface	N/A	Fabric channel 11	0x4C	12
10	XFP interface	N/A	Fabric channel 12	0x4D	13
11	XFP interface	N/A	Fabric channel 13	0x4E	14
12	XFP interface	defined by RTM	RTM	N/A	N/A
13	XFP interface	defined by RTM	RTM	N/A	N/A
14	XFP interface	0x10	Front panel XFP 1/1	N/A	N/A
15	XFP interface	0x11	Front panel XFP 1/2	N/A	N/A
16	XFP interface	0x12	Front panel XFP 1/3	N/A	N/A
17	XFP interface	0x13	Front panel XFP 1/4	N/A	N/A
18	SerDes	N/A	COM Express	N/A	N/A
19	XAUI interface	N/A	RTM	N/A	N/A

Configuration

The Fabric interface on the SFB is a managed 10 Gb Ethernet switch with the following features:

- Switch configuration via the PowerQUICC III PCI bus.
- Transmit (Tx) port disable based on E-Key port state.

The Fabric Ethernet configuration port is connected to the PowerQUICC III through PCI bus 2, a 32-bits wide bus operating at 66MHz.

10 Gigabit Ethernet ports

For the standard SFB configuration, twenty 10 Gb Fabric Interface ports are available on the Fabric Ethernet:

- Twelve ports provide direct XAUI node-slot backplane connections and can be configured as 10 Gb XAUI or 1 Gb SerDes.
- Four XFP ports on the front panel. Each XFP 10 Gb connector uses XFI signaling, composed of single transmit and receive 10 Gb differential pairs. A XAUI to XFI interface chip converts the signaling from the XAUI signals at each switch port to the XFI signals on the XFP connector.
- Three ports routed to the RTM, which can be configured as 10 Gb XAUI or 1 Gb SerDes.
- One port is configured for 1Gb operation and provides a connection to the COM Express module site.

XFP port LEDs

This table describes the LEDs specific to the Fabric Ethernet. The LEDs are controlled by the Fabric Ethernet switch through a serial LED bus. Each of the Fabric Ethernet XFP ports on the front panel or the RTM has two LEDs:

1. 1 green link/activity status LED:
    - **Green** – link established

- **Blinking green** – transmit/receive activity.
- **Off** – no link or port disabled.

2. 1 green port status LED:
   - **Green** – link established.
   - **Off** – port is disabled

LED Controller Interface Subsystem

Ethernet port status LED indicators are controlled from the Base Ethernet and the Fabric Ethernet switches through serial LED data streams consisting of clock and data signals. One stream comes from the Fabric Ethernet switch and one comes from the Base Ethernet switch. Each bit position in the bit-stream indicates the on/off value of one port status LED. The streams are routed to a complex programmable logic device (CPLD), which acts as a hardware LED controller-driver. The stream is then routed to the Zone 3 connector to support a LED controller on an optional rear I/O module (e.g. RTM).

A three-wire serial peripheral interface port (SPI) is available on the LED control CPLD. The SPI supports read operations on the CPLD Revision ID register. Write operations are not supported.

Hardware Management Subsystem

The function and operation of the SFB hardware management subsystem is controlled by the Intelligent Platform Management Controller (IPMC). The hardware management subsystem is the collection of IPMCs and sensors on the installed modules and field replaceable units (FRUs), and the communication between these devices and the Shelf Manager. The SFB's IPMC manages the commands and the data portion of this subsystem.

The IPMC has the following features:

- Dual I2C interface to backplane intelligent platform management bus (IPMB) with programmable pull-ups.
- Serial interfaces to LMP and COM Express site.
- I2C interfaces to I/O expander, field-programmable gate array (FPGA), temperature sensors, voltage sensors, and RTM.
- Remote executable flash and micro-controller software update support.
- Electronic keying (E-Keying) support for Base interface, Fabric interface, synchronization clocks, and update channels.
- Remote micro-controller software update support

Non-volatile Random Access Memory

The non-volatile random access memory (NVRAM) in the IPMC on the SFB is a 64 KB device. The NVRAM stores the following types of information about the field replaceable units (FRUs):

- Serial number.
- Part number.

- Manufacturer.
- Date and time of manufacture.
- Product name.
- FRU capabilities.
- Point to point connectivity records for modules that plug into the backplane, which is used for E-Keying.

The SFB acting as the active Shelf Manager, retrieves this information from the IPMC and stores the information. To provide redundancy, the active Shelf Manager also sends the FRU information to the inactive Shelf Manager, which also stores the information. This information is recorded in the system event log (SEL) to monitor system events.

IPMC Watchdog Timer

The IPMC entity includes a hardware watchdog timer. Once the watchdog is activated, the IPMC must strobe it in order to keep it from timing out. If a firmware or a hardware problem on the IPMC causes it to stop strobing the watchdog timer, the IPMC is automatically isolated from the IPMB. This isolation of the IPMC keeps the IPMI buses functional for the remaining IPMC devices. A two-pin header can be used to disable the watchdog timer during debug.

COM Express Site Subsystem

The SFB has sites for an optional COM Express module and a peripheral SAS or SATA drive, which are available as an SFB build option.

Chapter 3: Physical Interfaces

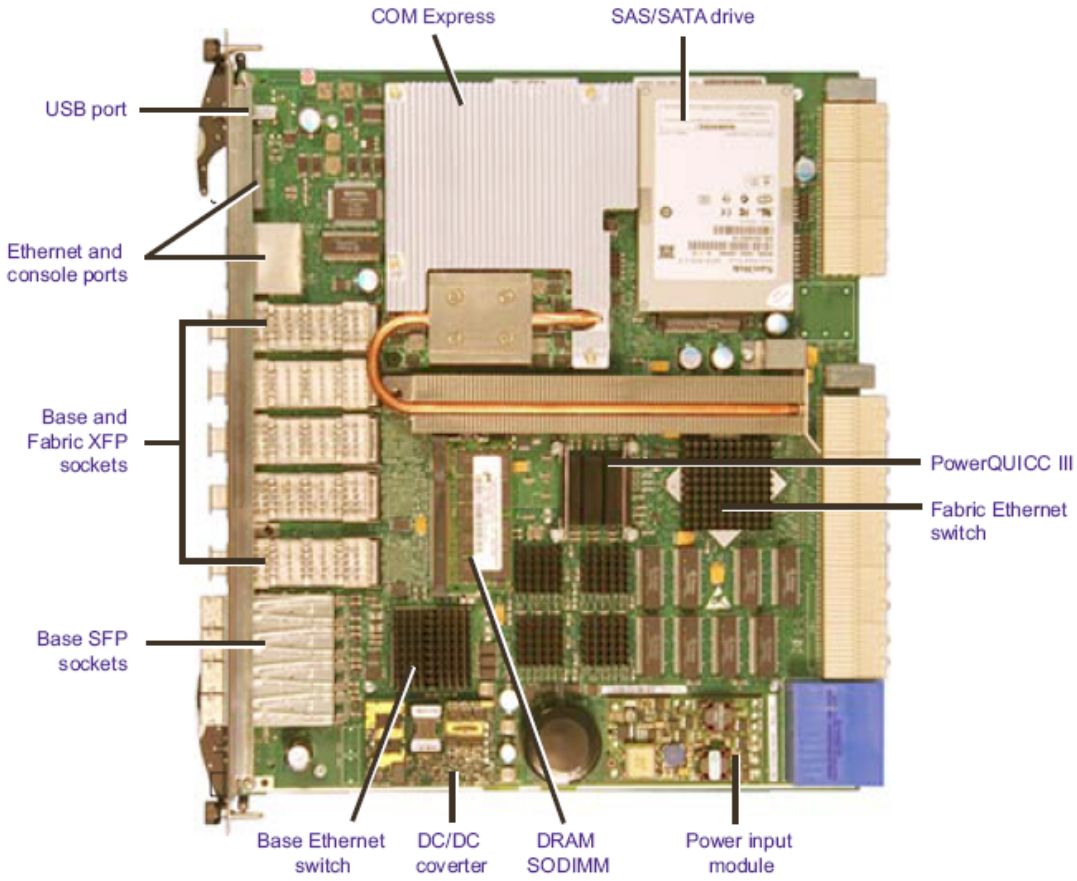


Figure 3.1. Switch Fabric Blade Base Board Layout

Mechanical Specification

The SFB dimensions conform to the PICMG AdvancedTCA 3.0 R2.0 Specification.

Thermal Design

Heat sinks are used in the SFB's thermal design. Extruded and crosscut aluminum heat sinks with thermal interface pads are used to cool the Base Ethernet switch, the Fabric Ethernet switch, the

Base Ethernet PHYs, and the PowerQUICC III processor. The heat sinks for the XFPs are a pin fin configuration and are compliant with the specifications defined in the XFP MSA.

Air Intake and Cooling

Each chassis and the configuration of modules installed in each chassis is unique. The full complement of front modules, carriers, fans, air management filler panels, and power components can greatly vary between systems and affects the amount of air intake and cooling required for your system.

Front Panel Interfaces and LEDs

Each SFB includes a sheet metal front panel that will serve as an EMI/RFI barrier and complies with PICMG 3.0 Revision 2.0. The front panel provides access to the LMP, the Base Ethernet switch and the Fabric Ethernet switch.

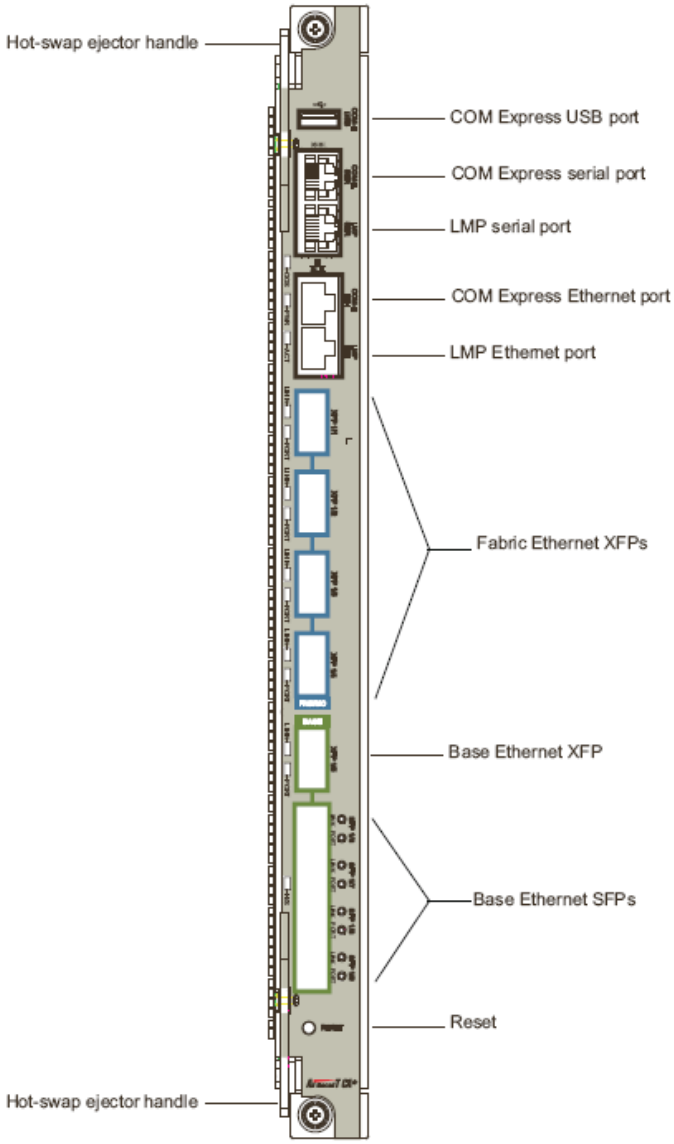


Figure 3.2. Switch Fabric Blade Interface Locations

Interface descriptions and pinouts

- **Push buttons/switches**

  The SFB includes a reset button and a hot-swap switch. The recessed reset push button only resets the payload section of the SFB when pressed. The hot-swap switch closes when the ejector handle is fully latched.

- **LMP serial console port**

  An RJ-45 connector provides serial access to the console port of the LMP (labeled LMP SER). This may be used as a console port for setup and management of applications running on the PowerQUICC III LMP.

Note
Use the LMP serial console port for temporary connections only. Use a shielded cable with the port.

- **LMP Ethernet maintenance port**

  An RJ-45 connector is used to provide a maintenance port for the SFB. This Ethernet port interfaces to the PowerQUICC III LMP. Two integrated LEDs on each connector indicate link status and activity.

Note
Use the LMP Ethernet port for connections to indoor cables only. Use a shielded cable with the port for applications that need to fully comply with NEBS GR 1089 requirements.

- **XFP Base Ethernet and Fabric Ethernet ports**

  The XFP connectors are used to provide front panel access to the 10 Gb ports of the Base and the Fabric Ethernet switches. The front panel has four 10 Gb XFP sockets with 10 Gb XFI-to-XAUI interfaces to the Fabric Ethernet switch and one XFP socket with a 10 Gb XFI-to-XAUI interface to the Base Ethernet switch. Two LEDs next to each connector indicate link, activity, and port status.

  Each connector is coupled to a switch port through a XAUI-to-XFI interface. The connector is inside a metal cage, which is connected to the SFB. You can install or remove XFP transceivers while the SFB is running, but they are not auto-detected. XFP module loading is optional.

  An XFP port will not work if:

  1. No module is installed.

  2. The port is administratively disabled.

- **SFP Base Ethernet interface ports**

  The SFP Base Ethernet connectors are used to provide front panel access to the Base Ethernet switch. The front panel has four SFP sockets with 1 Gb SerDes interfaces to the Base Ethernet switch when optical SFPs are installed or 10/100 Mbit SerDes interfaces to the Base Ethernet switch when copper SFPs are installed. Two LEDs next to each connector indicate link, activity, and port status.

  Each connector is directly coupled to a port of the Base Ethernet switch. You can install or remove SFP transceivers, which are auto-detected, while the SFB is running. SFP loading is optional.

An SFP port will not work if:

1. No module is installed.
2. The port is administratively disabled.

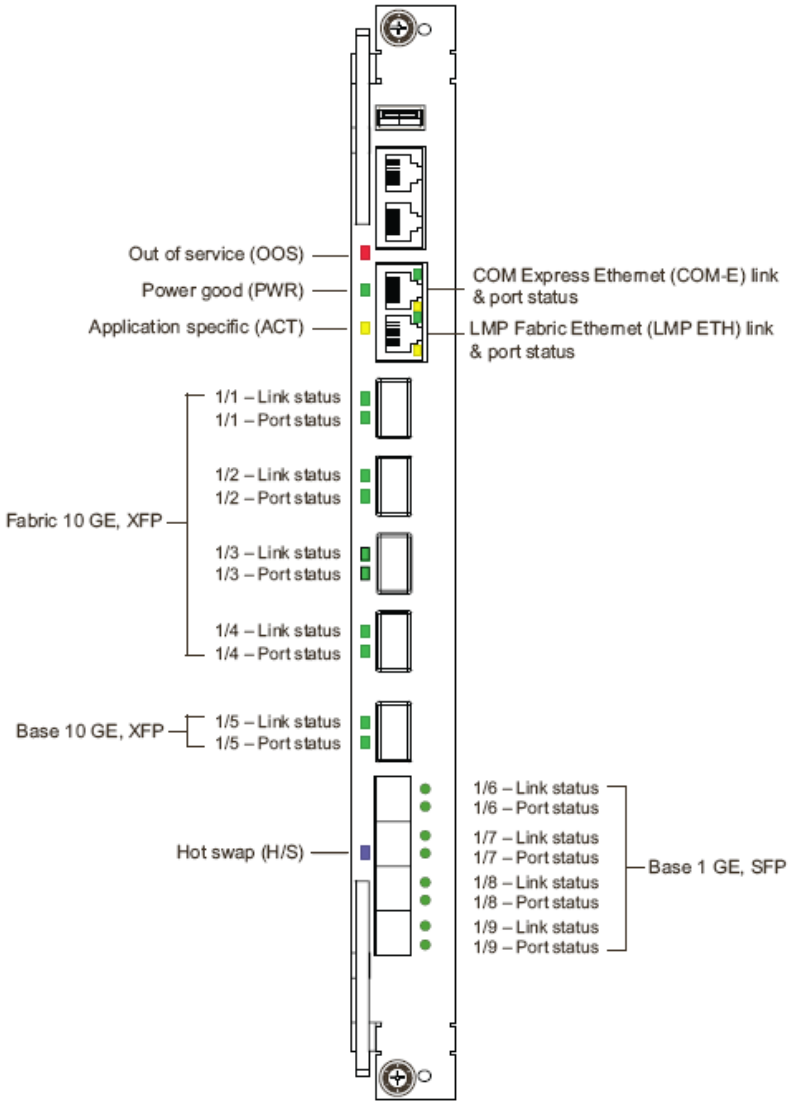


Figure 3.3. LED Positions on Front Panel

Description	States
One bi-color Out of Service (OOS) "LED1" controlled by GPO signals on the IPMC.	Amber = OOS Off = normal operation
One green Power Good "LED2" controlled by GPO signals on the IPMC.	Green = power is good
One amber Module State "LED3" controlled by GPO signals on the IPMC.	Amber flashing = booting Amber on = functioning Off = inactive
One blue Hot Swap ready "Blue LED" controlled by an IPMC output signal.	Blue = ready for hot swap
Two green COM Express and LMP Ethernet console port links LED part of the RJ-45 connector - indicates PHY port activity.	Green = link Off = no link
Two amber COM Express and LMP Ethernet console port activity LED part of the RJ-45 connector - indicates PHY port activity.	Amber blink = activity Off = no link or port disabled
Four green Fabric Ethernet XFP port status LEDs controlled by the LMP through the 10 G PHY LED signals.	Green = link is good Off = no link or port disabled
Four green Fabric Ethernet XFP link/activity LEDs controlled by the LMP through the 10 G PHY LED signals.	Green = enabled Green blink = activity Off = no link or port disabled
One green Base Ethernet XFP port status LEDs controlled by the LMP through the 10 G PHY LED signals.	Green = link is good Off = no link or port disabled
One green Base Ethernet XFP link/activity LEDs controlled by the LMP through the 10 G PHY LED signals.	Green = enabled Green blink = activity Off = no link or port disabled
One green Base Ethernet SFP link/activity LEDs part of the RJ-45 connector – indicates GE PHY port status, driven through the PHY LED signals.	Green = link is good Green blink = activity Off = no link or port disabled
Four green Base Ethernet SFP port status LEDs part of the RJ-45 connector – indicates GE PHY port status, driven through the PHY LED signals.	Green = enabled Off = port disabled

Figure 3.4. Front Panel LED States

Backplane Interfaces

The following SFB interfaces are available through the backplane:

- Twelve 10/100/1000Base-T Base interface node ports.
- One inter-LMP link to a redundant SFB through the update channel.
- One Base Ethernet switch-to-switch link port through Base channel 2.
- Dual 10/100Base-T Base interface port to one or two external Shelf Manager's through Base channel 1.
- Twelve 10 Gb XAUI Fabric interface node ports.
- Six synchronization channel clock input/outputs to the NTS.
- Master/slave link to NTS module clock through the update channel.
- One 10/100/1000Base-T link between COM Express modules through the Update channel when two redundant SFBs installed.

- Two -48V power rails.
- IPMB interface (two I2C ports) with programmable pull-ups.
- Eight hardware address lines, seven address/one parity.

Update Channels

Update channels are the backplane connections that exist between a pair of SFBs operating on a redundant basis. Application software can use update channels for redundancy interlock.

If you configure a pair of SFBs to use the update channel for redundancy support, you must install the SFBs into hub slots linked by an update channel. The update channel for SFBs exists between physical slots 1 and 2 in a Clavister 6-slot chassis and 7 and 8 in a Clavister 14-slot chassis.

Rear Transition Module

When an SFB is installed in a chassis, an associated rear transition module (RTM) can be connected to provide external access to some of the SFB's signals and electrical connections. The RTM is installed in the rear of the platform and connects to the SFB using the Zone 3 connectors.

Electrical connections between the SFB and the RTM include:

- Two to four 1 Gb SGMII/SerDes ports from the Base interface.
- Two 10 Gb XAUI ports from Base interface.
- One to three 10 Gb XAUI ports from Fabric switch.
- Four external reference clock inputs.
- Three external system clock outputs.
- I2C interface to IPMC for ID, temperature, temperature sensors, and control management.
- I2C interface to the LMP for ID.
- SPI to the LMP.
- PCI Express interface from COM Express site.
- Switched 12 V from payload supply.
- Current limited 3.3 V from IPMC supply.
- RTM hot-swap switch closure signal.
- Hot-swap RTM LED output from the IPMC.

The block diagram below shows an SPM designed for a 14-slot chassis.

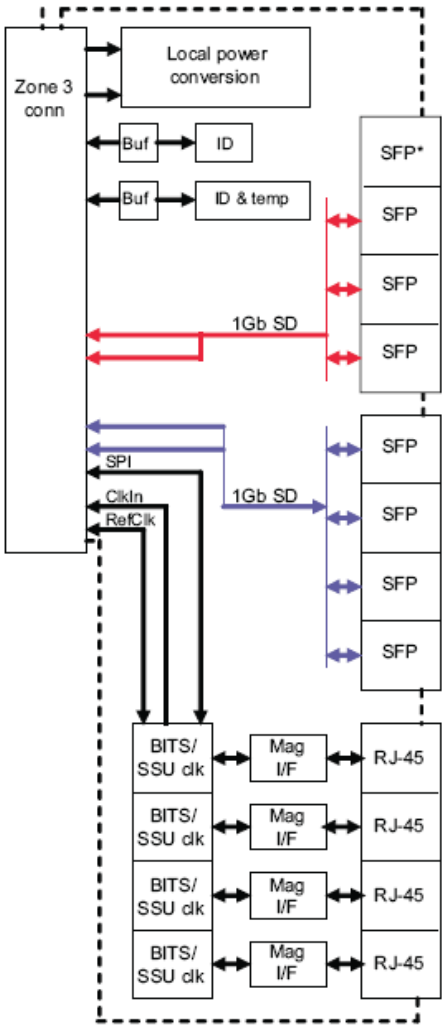


Figure 3.5. SPM Block Diagram

Chapter 4: Initial Verification and Configuration

Connecting to the SFB

Serial and Telnet connection are described below.

- **Serial Connection**

  1. Connect the serial cable to the serial port of the SFB (labeled LMP SER). The appropriate 9-pin D-shell to RJ-45 cable was included in SFB package.

  2. Connect the cable's other end to the COM1 or COM2 serial port of an external computer (or to a dumb terminal).

  3. Start a terminal-emulator application on the external computer. Specify 115200 baud, 8 data bits, no parity, one stop bit, and no hardware or software flow control. When the terminal emulator is set up and connected, you will receive a login prompt.

- **Telnet Connection**

  1. Connect an Ethernet cable to the SFB's Ethernet maintenance port (labeled LMP ETH).

  2. Connect the cable's other end to the Ethernet port of a computer that is connected to your network.

  3. Configure your computer to a subnet IP address (*10.0.0.x*, but not *10.0.0.1*) and set the netmask to *255.255.255.0*.

  4. Telnet to IP address 10.0.0.1 (port 23). When the connection is made, you will receive a login prompt.

Logging in to the SFB

The table below shows which user names you can enter to access the command line interface (CLI) or the Linux shell.

CLI	SFB login prompt	**admin** and password
CLI	Linux shell	**mcli**
Linux shell	SFB login prompt	**root** and password

Linux shell	CLI	**linux-shell**

If you have not yet set passwords, no password is required for the root and admin accounts.

Adding User Accounts and Setting Passwords

Establish passwords for the root and admin users during initial setup. You should also create user accounts for those who do not require administrative privileges. For details on account administration, see the documentation for the version of Linux that you are using.

Note
To connect to the SFB, use either Telnet or log in through the Shelf Manager.

To set up another account that goes directly to the CLI, set the login script for the account to */usr/bin/mcli*. This mimics how the admin user is set up in the */etc/passwd file*.

Once logged in, you are not prompted for a password when moving from the CLI to the Linux shell and back.

Rebooting the SFB

To reboot the SFB from the Linux shell, enter:

```
reboot
```

To physically initiate a reboot of the SFB, use a pen or narrow instrument (not a pencil) and firmly push the recessed reset button on the front panel.

Configuring IP Addresses

IP interfaces are configured by default to minimize your work in setting them up. Some of these interfaces are set up to get IP addresses from a DHCP server. Others are assigned unique static addresses by including variables for the logical chassis, physical slot, or logical slot numbers in the IP address. A few interfaces have static, literal IP addresses assigned to them that should not be changed.

We recommend that you:

- Keep the default IP addresses for certain interfaces.
- Obtain dynamic IP addresses for interfaces configured for DHCP by connecting the SFB to a network that has a DHCP server and running the DHCP client.
- Add static, literal IP addresses as alternates to some of the literal addresses already assigned (as subinterfaces).

Keeping Default IP Addresses for Certain Interfaces

Do not change the default IP addresses for these interfaces:

- **eth1** - This update channel is used to communicate with the redundant SFB to synchronize the Shelf Managers. Changing this address would reduce performance of Shelf Manager

communications with its peer.

- **lo** - This internal loopback interface is not visible to any network. Changing it may disrupt occasional communications within the SFB between applications and the local management processor (LMP).

Requesting IP Addresses Through DHCP

The *dtl0* and *dtl1* interfaces are configured to request addresses through DHCP, with Clavister-defined client IDs.

To get addresses:

1. Physically connect the SFB to a network that has a DHCP server.
2. Connect to the SFB through the serial port or with Telnet.
3. Ensure that the interfaces on the Base and Fabric switches that are expected to communicate with the DHCP server are administratively enabled.
4. Access the Linux shell.
5. Verify that the interfaces have received IP addresses by entering:

   ```
   ifconfig dtl0
   ifconfig dtl1
   ```

   Look for the "inet addr" values. If they do not show, wait five minutes and look again.

When the SFB reboots, an address is requested. The previous address is initially suggested, but if it is not free, a new address is provided by the DHCP server.

Adding Static IP Addresses for Management Ports

This procedure configures subinterfaces with alternate IP addresses for the ports:

- **eth0:1** — subinterface for the front panel Ethernet maintenance port.
- **dtl0:2** — subinterface for the Base interface connection to the local management processor (LMP).
- **dtl1:0** — subinterface for the Fabric interface connection to the LMP

To configure alternate IP addresses for the ports:

1. Connect to the SFB's serial port.
2. Access the CLI.
3. Select the Base interface switch by entering:

   ```
   base-ethernet
   ```

4. Configure the front panel port (*eth0*) with a new IP address, subnet mask, and (optional) default gateway, using this syntax:

   ```
   serviceport ip <ipaddr> <netmask> [gateway]
   ```

   This establishes the subinterface *eth0:1* with the IP address you assigned.

5. Configure *dtl0* with a new IP address, subnet mask, and (optional) default gateway, using this syntax:

```
network parms <ipaddr> <netmask> [gateway]
```

This establishes the subinterface *dtl0:2* with the IP address you assigned.

6. Exit the Base interface by entering:

```
exit
```

7. Select the Fabric interface switch by entering:

```
fabric-ethernet
```

8. Configure the Fabric interface connection to the LMP with a new IP address, subnet mask, and (optional) default gateway, using this syntax:

```
network parms <ipaddr> <netmask> [gateway]
```

This establishes the subinterface *dtl1:0* with the IP address you assigned.

9. Exit the switch portion of the CLI by typing:

```
exit
```

10. When prompted to save changes, say yes "*y*" to save the changes to the SFB's startup configuration.

Per-VLAN Interfaces

In some use cases for the SFB, customers may need to route management traffic over more than one VLAN. The illustration below shows an example of such a use case.

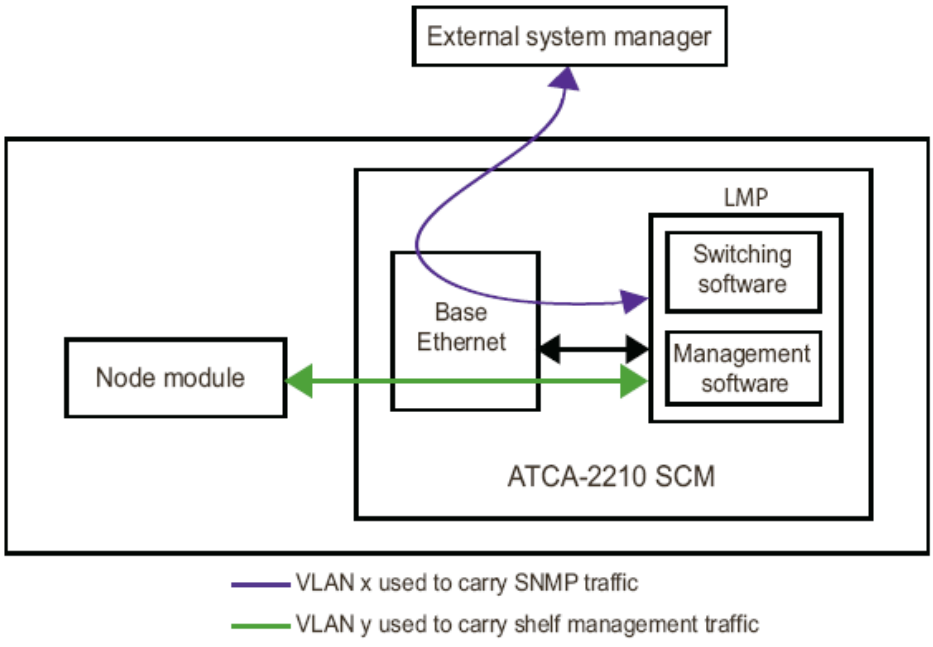


Figure 4.1. Use-case for VLAN Interfaces

In the above, the term *Node module* refers to a front module that is not a switch and control module. For example, node modules can include carrier modules, compute modules, storage modules, I/O modules, and line cards.

As shown above, chassis management traffic can be carried over a VLAN that exists only inside the chassis, while higher level management traffic, such as SNMP, can be carried over a VLAN set up for an entire system. This system could consist of multiple chassis.

The SFB supports this use case with per-VLAN interfaces. Per-VLAN interfaces are created using the 802.1Q VLAN driver and configured with the vconfig application. The syntax for adding and removing per-VLAN interfaces is:

```
vconfig add [interface-name] [vlan_id]
```

```
vconfig rem [interface-name] [vlan_id]
```

The following example shows a per-VLAN interface (*dtl0.5*) being created for a VLAN (VLAN 5). The IPv4 address *192.168.5.1/24* is assigned to the per-VLAN interface:

```
vconfig add dtl0 5
```

```
iconfig dtl0.5 192.168.5.1 netmask 255.255.255.0
```

When setting up your own per-VLAN interfaces, be sure to verify the following once the per-VLAN interface has been created and configured:

- The appropriate Base or Fabric interfaces have been configured to participate in the VLAN.
- The traffic being sent from the per-VLAN interface is reaching the desired destination or destinations.

Configuring SNMP Trap Destinations and Security Access

To configure the SNMP notification (trap and inform) destinations and community names:

1. Connect to the SFB.
2. Access the Linux prompt.
3. Make a backup copy of the configuration file:

   ```
   cp /etc/snmp/snmpd.conf /etc/snmp/snmpd.conf.orig
   ```

4. Edit the configuration file */etc/snmp/snmpd.conf*.
5. Optionally Configure a trap destination by uncommenting a template line and filling in values as follows:

   a. Locate the line that begins with "*#trap2sink*".

   b. Fill in the host IP address and community name in the correct positions as follows:

```
trap2sink <ip_addr> <community_name> 162
```

c. Uncomment the line by removing the "#" character, and add more lines as necessary.

6. Optionally configure an inform destination by uncommenting a line and filling in values as follows:

   a. Locate the line that begins with "*#informsink*".

   b. Fill in the host IP address and community name in the correct positions as follows:

```
informsink <ip_addr> <community_name> 162
```

   c. Uncomment the line and add more lines as necessary.

7. Configure community names by uncommenting lines and filling in values as follows:

   a. Locate the lines that begin with "*#com2sec*". The lines that specify the context name as "*-Cn fi*" configure access to the Fabric interface switch objects. The other lines configure access to all other objects, including Base interface switch objects. The items ending in "*RO*" or "*RW*" are the security names.

Warning
Do not change the context name "fi", or you will lose access to many Fabric interface switch objects.

   b. Change the community names. This step is optional, but highly recommended.

   c. Uncomment these lines and add more lines as appropriate.

8. Save the file. The saved changes take place when you enable or restart the SNMP agent. Changes will also take effect if the agent reads its own configuration file. Changes to the file are saved persistently, so they will be restored after the SFB is rebooted.

Note
If you have installed the Net-SNMP man pages onto a system, you can find more details on the snmpd.conf file options using the command:

```
man snmpd.conf
```

Enabling the SNMP Agent

This procedure assumes that you have already performed the steps described above in *Configuring SNMP Trap Destinations and Security Access*.

To enable the SNMP agent:

1. Connect to the SFB.

2. Access the CLI.

3. Access blade-management mode by entering:

```
blade-mgmt
```

4. Enable the master agent by entering:

```
service snmpd
```

5. Exit blade-management mode by entering:

```
exit
```

6. Make the change to the agent's enable status persistent across reboots by entering:

```
copy system:running-config nvram:startup-config
```

Enabling the DHCP Server

The SFB's DHCP server is set up to work with a redundant SFB to provide primary and secondary DHCP servers. Configuration files are supplied in */etc* as *pri.dhcpd.conf.example*, *sec.dhcpd.conf.example*, and *common.dhcpd.conf.example*, but they must be modified to fit your network.

If you want to use the SFB's DHCP server to provide IP addresses, configure and enable the DHCP server as follows:

1. Connect to the SFB.
2. Access the Linux prompt.
3. Change to the directory containing the DHCP server configuration files:

```
cd /etc
```

4. Copy the example configuration files or your replacement files to the required file names as follows:

```
cp pri.dhcpd.conf.example pri.dhcpd.conf
```

```
cp sec.dhcpd.conf.example sec.dhcpd.conf
```

```
cp common.dhcpd.conf.example common.dhcpd.conf
```

5. Modify the interface configuration for the SFB interface that will serve addresses. The DHCP server configuration uses *dlt0* by default (as specified in the */etc/default/dhcp* file), but *dtl0* uses a dynamic IP address by default.
6. Verify that the interface that will serve addresses is administratively enabled through the Base or Fabric Ethernet CLI.
7. Modify the configuration files as necessary for your network. Make the address and the peer address match the IP addresses that the two SFBs will use to communicate with each other

in both the primary and secondary files. The *eth1* update channel IP address can be used for this purpose.

8. If this SFB will host the secondary DHCP server, change the configuration as follows:

   a. Edit the DHCP server initialization script */etc/rc.d/init.d/dhcpd*.

   b. Locate the *FAILOVER_PRIMARY* line that appears after the commented *Read Me* text. The line should look like this:

   ```
   FAILOVER_PRIMARY=1
   ```

   The default value is *1*, indicating the primary, so no changes are needed on the SFB hosting the primary DHCP server.

   c. Change the *FAILOVER_PRIMARY* value to 0 to indicate the secondary. The line should then look like this:

   ```
   FAILOVER_PRIMARY=0
   ```

   d. Save the file.

9. Create a directory for the lease database file:

   ```
   mkdir -p /rsys/onboot.data/bindmount/var/state/dhcp
   ```

10. Make the directory persistent:

    ```
    touch /rsys/onboot.data/bindmount/var/state/dhcp/.bindmount
    ```

    The directory appears in the file system as */var/state/dhcp*, and is not present until the SFB reboots.

11. Create a symbolic link from the */etc/rc.d/rc2.d* directory to the startup script:

    ```
    ln -s ../init.d/dhcpd /etc/rc.d/rc2.d/<link_name>
    ```

    For example:

    ```
    ln -s ../init.d/dhcpd /etc/rc.d/rc2.d/S53dhcpd
    ```

    This causes the DHCP server to be started automatically at SFB bootup. The value of <link_name> determines the order of startup relative to other scripts in the rc2.d directory and should place this service after all other scripts upon which this service depends.

12. Enable the DHCP server immediately by running the startup script:

    ```
    /etc/rc.d/init.d/dhcpd start
    ```

Enabling and Disabling the Telnet Server and TFTP Server

If you want to restrict access to the SFB, you can disable the Telnet server and the TFTP server, which are enabled by default. These services are controlled by the */etc/inetd.conf* file.

Warning

Do not disable the Telnet server until you confirm access to the SFB through other means. For example, the SSH service is enabled by default, but a user account with a

password must be set up before SSH can be used.

To enable or disable the Telnet server or TFTP server:

1. Connect to the SFB.
2. Access the Linux prompt.
3. Change to the directory containing the DHCP server configuration files:

```
cd /etc
```

4. Edit the *inetd.conf* file as appropriate:
   a. To disable the Telnet server, comment out the line beginning with *telnet*. You can comment out the line by inserting a "#" character at the beginning of the line.
   b. To disable the TFTP server, comment out the line beginning with *tftp*.
   c. To enable the Telnet server, uncomment the line beginning with *telnet* (by removing the "#" character).
   d. To enable the TFTP server, uncomment the line beginning with *tftp*.

   This changes the enable states of the services the next time the SFB boots up.
5. Enable or disable the services immediately by restarting *inetd*:

```
/etc/init.d/inetd restart
```

Enabling and Disabling Syslog and Other Services

If you want the SFB to log messages to a file, enable the syslog service. You may want to enable other services (such as NTP) that are disabled by default, or to re-enable a service after it has been disabled.

To enable a service:

1. Prepare a configuration file for the service with appropriate contents (for example, syslog.conf).
2. Copy the configuration file to the correct location (*/etc*) on the SFB.
3. Create a symbolic link from the /etc/rc.d/rc2.d directory to the service's startup script:

```
ln -s ../init.d/<startup_script> /etc/rc.d/rc2.d/<link_name>
```

   For example:

```
ln -s ../init.d/syslog /etc/rc.d/rc2.d/S12syslog
```

   This causes the service to be started automatically at SFB bootup. The value of <*link_name*> determines the order of startup relative to other scripts in the *rc2.d* directory and should place this service after all other scripts upon which this service depends.
4. Enable the service immediately by running the startup script:

```
/etc/rc.d/init.d/<startup_script> start
```

For example:

```
/etc/rc.d/init.d/syslog start
```

To disable a service:

1. Change to the */etc/rc.d/rc2.d* directory:

```
cd /etc/rc.d/rc2.d
```

2. Identify the symbolic link to the appropriate startup script. For example, listing the details of the syslog link might show:

```
lrwxrwxrwx 1 root root 16 May 29 08:57 S12syslog ->
../init.d/syslog
```

3. Disable the service immediately by running the startup script with the *stop* option:

```
/etc/rc.d/init.d/<startup_script> stop
```

For example:

```
/etc/rc.d/init.d/syslog stop
```

4. Keep the service from restarting when the SFB reboots by removing the symbolic link:

```
rm <link_name>
```

For example:

```
rm S12syslog
```

The service can be re-enabled later by adding the link again as describe above.

Copying a Configuration from Existing CLI Configuration Files

You can change the configuration of the SFB using CLI configuration files created on another SFB or created on this SFB but stored remotely. The files should be placed on an accessible TFTP server (for example, in the */tftpboot* directory). This procedure copies the files to the SFB and makes them the current and persistent configurations.

To change to configuration settings created by CLI commands:

1. Connect to the SFB.
2. Access the CLI.
3. Retrieve the Base Ethernet switch configuration file and make it the running configuration:

```
base-ethernet
```

```
copy tftp://<IP address>/<file-1> nvram:startup-config
```

Executing the command returns you to the master CLI.

4. Retrieve the Fabric Ethernet switch configuration file and make it the running configuration:

```
fabric-ethernet
```

```
copy tftp://<IP address>/<file-2> nvram:startup-config
```

Executing the command returns you to the master CLI.

5. Retrieve the blade-management configuration file and make it the running configuration (from the master CLI):

```
copy tftp://<IP address>/<file-3> nvram:startup-config
```

```
copy nvram:startup-config system:running-config
```

The running configuration and the persistent startup configuration has changed to reflect the content of the configuration files.

Copying CLI Configuration Files to a TFTP Server

You can copy the CLI configuration files from the SFB to a TFTP server. This procedure also saves the current running configuration to be the SFB's persistent startup configuration, if it wasn't already.

Most TFTP servers require that the target files being copied already exist on the target TFTP server. The target files must also have "global write" permissions. To avoid having to create the files in advance, start the TFTP server with the *-c* option. If you have the Linux man pages installed on a system, see the *tftpd* man page for details.

To save the current configuration to be the startup configuration and to copy the configuration files to a TFTP server:

1. Connect to the SFB.
2. Access the CLI.
3. Save the Base Ethernet switch configuration file to persistent storage and copy it to the server:

```
base-ethernet
```

```
copy system:running-config nvram:startup-config
```

```
copy nvram:startup-config tftp://<IP address>/<file-1>
```

```
exit
```

4. Save the Fabric Ethernet switch configuration file and copy it to the server:

```
fabric-ethernet
```

```
copy system:running-config nvram:startup-config
```

```
copy nvram:startup-config tftp://<IP address>/<file-2>
```

```
exit
```

5. Save the blade-management configuration file and copy it to the server (from the master CLI):

```
copy tftp://<IP address>/<file-3> nvram:startup-config
```

```
copy system:running-config nvram:startup-config
```

```
copy nvram:startup-config tftp://<IP address>/<file-3>
```

The configuration now resides on the TFTP server, and will be restored from persistent storage when the SFB is rebooted.

Copying Any File to the SFB

You can copy any file to the SFB using TFTP. The files should be placed on an accessible TFTP server (for example, in the */tftpboot* directory).

To copy a file to the SFB:

1. Change to the directory where the files should be placed on the SFB.
2. Connect to the TFTP server:

```
tftp <IP address of TFTP server>
```

3. Optionally for binary files, specify binary mode:

```
bin
```

4. Retrieve the file:

```
get <filename>
```

   Repeat this step for each additional file.

5. Exit TFTP:

```
quit
```

Chapter 5: Software Features

Shelf Manager

The SFB provides a PICMG-compliant *Shelf Manager* and the IPMI infrastructure for managing the chassis and FRU data. The Shelf Manager keeps track of alarms and provides E-Key services to modules to authorize them to power up (if appropriate) when inserted into the chassis. The Shelf Manager also corrects many error conditions and controls overheating situations by increasing fan speeds.

Ethernet Switching

The SFB provides Ethernet switching software that runs on both the Base and Fabric switches. The software contains support for many Layer 2 switching protocols such as virtual LANs, spanning tree, and Class of Service queuing. The SFB provides a CLI and some SNMP MIB support to manage the switch.

IP Routing

The SFB includes IPv4 routing features as a software option. These unicast and multicast routing features are supported on the Fabric interface only. The IPv4 implements these features using the Linux stack running on the LMP. With the IP routing feature set enabled, the SFB supports:

- OSPF (open shortest path first) version 2.
- RIP (router information protocol) version 1 and 2.
- DVMRP (distance vector multicast routing protocol).
- VRRP (virtual router redundancy protocol).

The routing interfaces created by the licensed software option and the routing table entries from Linux should not be modified. You manage the Layer 3 configuration on the SFB using CLI commands.

Linux and Boot Loader

The installed SFB software includes a Linux kernel along with user space applications and utilities.

The SFB uses a boot utility to load its software into RAM so it can run in a way that resembles diskless Linux.

The software includes a universal boot loader which initializes the SFB hardware and the Linux operating system. When the software image is upgraded, the boot loader should also be upgraded to ensure it includes the most current changes. When the boot loader is reflashed, the image and the environment variables are reset to the factory defaults.

The boot loader also provides an environment for debugging, changing the flash contents, and scripting. The primary and secondary boot devices each contain a copy of the boot loader image and its environment variables. When the boot loader is powered on, the IPMC starts a watchdog timer called the corrupt flash detection (CFD) watchdog timer. If the boot loader does not issue a "Stop Watchdog timer" command to the IPMC, the CFD timer will expire, and the loader will switch to the alternate boot loader flash bank and reset the LMP. If the CFD timer expires, the alternate bank will always be used.

Linux Services

All standard services of this Linux version are included on the SFB. While Clavister provides additional management interfaces and documentation for some services, the following are included without additional support:

- NFS client.
- FTP client.
- TFTP server and client.
- Watchdog daemon.
- Bootp.

Effects of Rebooting the SFB

- Base and Fabric interface connections from node modules may change over to the redundant SFB and possibly lose traffic.
- Spanning tree topology changes occur, possibly resulting in lost traffic.
- If the rebooted SFB hosts the active Shelf Manager or master network timing subsystem (NTS), they will fail over.
- The SFB terminates all of its Telnet sessions.
- Configuration settings revert to those saved as startup configurations (configuration changes not saved are lost).
- The file system is recreated, so files and directories revert to those saved persistently (changes not made persistent are lost).

Rebooting the SFB causes the following to occur:

- Base and Fabric interface connections from node modules may change over to the redundant SFB and possibly lose traffic.
- Spanning tree topology changes occur, possibly resulting in lost traffic.
- If the rebooted SFB hosts the active Shelf Manager or master network timing subsystem (NTS), they will fail over.

- The SFB terminates all of its Telnet sessions.
- Configuration settings revert to those saved as startup configurations (configuration changes not saved are lost).
- The file system is recreated, so files and directories revert to those saved persistently (changes not made persistent are lost).

Linux Prompt

The default Linux shell prompt can be very informative and follows this format:

```
<userName>@<hostName>@<LCAB>-<LCHAS>-<SLOT>:<pwd><symbol>
```

where:

- <*userName*> is the name of the user currently logged in.
- <*hostName*> is the system name.
- <*LCAB*> is the logical cabinet number stored in the FRU Information.
- <*LCHAS*> is the logical chassis (chassis address) stored in the FRU Information.
- <*SLOT*> is the physical slot number in the chassis.
- <*pwd*> is the present working directory.
- <*symbol*> is *#* for the root user or *$* for any other user.

If the FRU Information including the logical cabinet cannot be read at startup, it is considered an error condition, and the prompt uses this format:

```
<userName>@<hostName>@ DEBUGMODE :<pwd><symbol>
```

The prompt is defined in */etc/profile*.

Flash Memory Usage

The diagram below shows the layout of the flash memory devices used in the SFB. Each flash memory device consists of 64MB flash parts. The part of the flash memory selected by the IPMC as the boot device is located in the memory map at address FC000000h. Flash part 0 is the primary boot device and is located at address E0000000h. Flash part 1 is the secondary boot device and is located at address E4000000h.

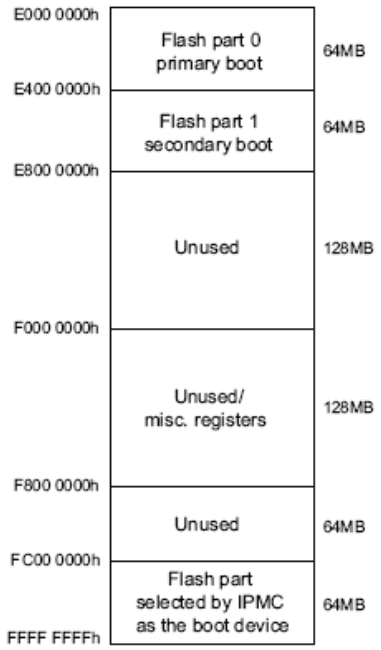


Figure 5.1. Flash Memory Usage

The diagram below shows the layout of the flash memory device used in the SFB. The JFFS is used to store configuration, log, and other non-volatile files. The JFFS image comes from the active part of the memory. The address of the JFFS image is FC000000.

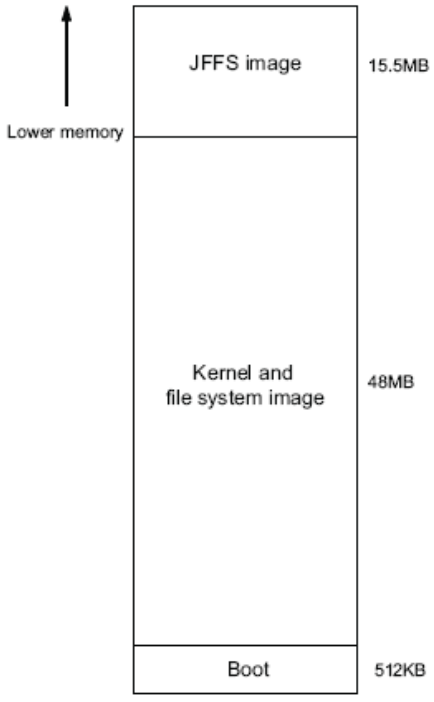


Figure 5.2. Flash Memory Device Map

The diagram below shows the layout of memory technology device (MTD) partitions used in the SFB. These partitions are used to load items from the flash and also to program the flash

memory.

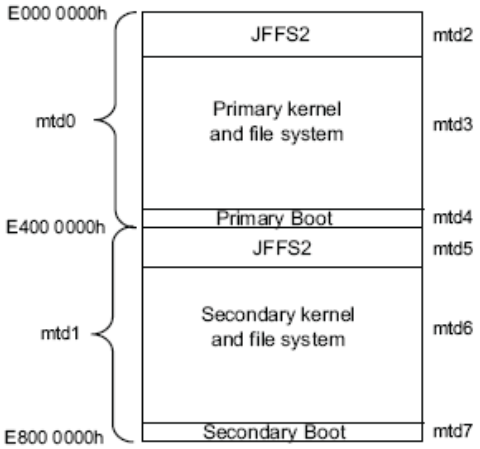


Figure 5.3. Flash Memory Device Map

RAM and File System Usage

The SFB uses a boot utility to load its software into RAM so it can run in a way that resembles diskless Linux.

The SFB runs its software from the RAM disk and uses a JFFS2 (journaling flash file system version 2) for persistent file storage. At SFB bootup, the boot utility copies the kernel and initial RAM disk (initrd) from flash memory into RAM. Then Linux converts the initial RAM disk into a normal RAM disk mounted as the root file system.

During the boot process, certain files in the JFFS are linked (bind mounted) to the RAM disk, causing the files in the JFFS to appear in RAM. Certain other files are copied (overlayed) to RAM. The linking and copying of files to RAM occurs before the initialization scripts are run.

File Persistence

Because the SFB file system is recreated in RAM at each reboot, changes made to files must be saved persistently to the JFFS or be lost upon reboot. Changes to a Clavister-defined set of configuration files can be preserved by deliberately saving them persistently through the CLI. Other files must be included in a persistency directory.

The persistency directories include files that should replace existing files or be added to the RAM file system upon bootup. The types of persistency directories are:

- **Bind Mount**

  The files are linked to the RAM disk, and any changes made to these files in RAM are persistent in the JFFS and relinked in RAM upon reboot. Use the bind mount directory for files that will change over time and must be persistent.

  The bind mount directory is */rsys/onboot.data/bindmount*. The contents of the directory are linked to the RAM file system at the root level (/). For example, */rsys/onboot.data/bindmount/sbin/make-client-id* is linked to */sbin/make-client-id*.

- **Overlay**

  The files are copied to the RAM disk, and any changes made to these files in RAM are lost upon reboot. Use the overlay directory for files such as executables that are normally not

changed. Overlaying is typically used to replace existing files.

The overlay directory is */rsys/onboot.data/overlay*. The directory's contents are recreated at the root level (/) in RAM. For example, */rsys/onboot.data/overlay/usr/bin/filename.txt* replaces the */usr/bin/filename.txt* file in RAM.

The JFFS partition is limited to 15.5 MB. Bind mounting and overlaying should be used conservatively, because the partition must hold the factory-installed persistent files, your persistent files, configuration files, and any changes made to the persistent files in RAM. In most cases, individual files should be made persistent instead of whole directories.

The factory default configuration includes a number of files and directories that are bind mounted and overlayed. You can view the current overlay and bind mount lists of persistent files which are generated each time the SFB boots, in */etc/version*.

All of the */etc* directory, which includes all configuration files generated by the CLI, is bind mounted for persistence. In some portions of the CLI, before you request that the configuration be saved or copied persistently, the configuration change exists only within the application. When you request that the current configuration be made persistent, the configuration file is written to the RAM disk, which is also saved to the JFFS because of the bind mount settings.

Adding a Persistent Directory

This procedure explains how to bind mount a directory to the RAM file system. This procedure assumes that the original directory exists in RAM on the SFB, but directories can be moved directly to the persistency directories from another system.

To make a new or existing directory and its contents persistent:

1. Copy the directory and its contents into the bind mount tree:

   ```
   cp -a <path><dir> /rsys/onboot.data/bindmount<path><dir>
   ```

   For example, when <path>=/var/lib/ and <dir>=misc:

   ```
   cp -a /var/lib/misc /rsys/onboot.data/bindmount/var/lib/misc
   ```

2. Specify that the directory should be made persistent rather than the individual files contained within it by creating a *.bindmount* file for the directory:

   ```
   touch /rsys/onboot.data/bindmount<path><dir>/.bindmount
   ```

   For example, when <path> is */var/lib/* and <dir> is *misc*:

   ```
   touch /rsys/onboot.data/bindmount/var/lib/misc/.bindmount
   ```

3. Reboot the SFB to make the persistency configuration take effect:

   ```
   reboot
   ```

After reboot, the directory and its contents immediately appear in RAM. After this point, any changes within a bind mounted directory are saved persistently.

Tip

If you accidentally made an empty directory persistent and overwrote a directory in RAM that had valuable contents, you can retrieve the contents by unmounting the directory

(with the umount command). Then you can repeat the above procedure. If you have rebooted the SFB, your contents are lost.

Adding a Persistent File

This procedure explains how to bind mount a file to the RAM file system. Overlaying a file is very similar, but instead uses the overlay directory (*/rsys/onboot.data/overlay*). This procedure assumes that the original file exists in RAM on the SFB, but files can be moved directly to the persistency directories from another system.

To make a file persistent:

1. Copy the file into the bind mount tree:

```
cp -a <path><file> /rsys/onboot.data/bindmount<path><file>
```

For example, when <*path*> is */sbin/* and <*file*> is *make-client-id*:

```
cp -a /sbin/make-client-id
                /rsys/onboot.data/bindmount/sbin/make-client-id
```

2. Reboot the SFB to make the persistency configuration take effect:

```
reboot
```

After reboot, the file appears in RAM and any changes are saved persistently.

Notes:

- The procedure for overlaying a file is very similar, but the overlay directory (*/rsys/onboot.data/overlay*) is used instead. In addition, any changes to an overlayed file are not saved persistently after reboot.
- Modifications made to the JFFS2 partition are currently not preserved across software upgrades. Track any changes made directly to the file system (bypassing the CLI), so they can be reapplied following an upgrade.

IPv4 Interface and Ethernet Switch Port Identifiers

This section describes the IPv4 interfaces on the local management processor (LMP) and the Ethernet switch port interfaces on both the Base and Fabric switches. An SNMP ifIndex is listed for each interface, and a CLI slot/port number is listed for each Ethernet switch port interface.

Below is a list of the LMP IPv4 interface identifiers and descriptions.

LMP IPv4 interface	SNMP *ifIndex*	Description
eth2	1	Ethernet connection to the Base switch port 2/2.
lo	2	Software loopback interface.
eth0	3	Ethernet connection to the SCM front panel maintenance port labeled LMP ETH.
eth1	4	Update channel between the two slots designated, used by the Shelf Manager to synchronize with its peer.
dtl0	5	Pseudo-Ethernet connection to the Base switch, implemented over PCI.
dtl1	6	Pseudo-Ethernet connection to the Fabric switch, implemented over PCI.

Below is a list of the Base Ethernet switch port identifiers and descriptions.

Base switch port	SNMP *ifIndex*	CLI slot/port	Description
To front slots	1000 + <slotnum> For example, 1003 for slot 3.	0/<slotnum> For example, 0/3 for slot 3.	Backplane Base Ethernet connections to front slots, where <slotnum> is the physical chassis slot number.
To front panel ports	1021	1/5	10 Gbps Base Ethernet connection to front panel port 1/5.
	1022	1/6	1 Gbps Base Ethernet connection to front panel port 1/6.
	1023	1/7	1 Gbps Base Ethernet connection to front panel port 1/7.
	1024	1/8	1 Gbps Base Ethernet connection to front panel port 1/8.
	1025	1/9	1 Gbps Base Ethernet connection to front panel port 1/9.
To other peer.	1033	2/1	Backplane Base Ethernet connection to the other, redundant peer (AdvancedTCA 3.0 Base channel 2).
To LMP	1034	2/2	Base Ethernet connection to the LMP eth2 interface.
To COM Express	1035	2/3	Base Ethernet connection to the COM Express module.
To RTM	1053	3/5	Base Ethernet connections to the RTM through the zone 3 connector.
	1054	3/6	
	1055	3/7	
	1056	3/8	
To LMP	1065	N/A	Base pseudo-Ethernet connection to the LMP dtl0 interface, implemented over PCI.

Base switch port	SNMP *ifIndex*	CLI slot/port	Description
Link aggregation connections	1066	4/1	Base Ethernet link aggregation (LAG) logical port-channels. These exist only if created by users and are numbered in the order in which they were created.
	1067	4/2	
	1068	4/3	
	1069	4/4	
	1070	4/5	
	1071	4/6	
	1072	4/7	
	1073	4/8	

Below is a list of the Fabric interface Ethernet interface identifiers and descriptions.

Fabric Switch Port	SNMP ifIndex	CLI Slot/Port	Description
To front slots	2000 + <slotnum> For example, 2005 for slot 5.	0/<slotnum> For example, 0/5 for slot 5.	Backplane Fabric Ethernet connections to front slots, where <slotnum> is the physical chassis slot number.
To front panel ports	2017	1/1	10 Gbps Fabric Ethernet connection to front panel port 1/1.
	2018	1/2	10 Gbps Fabric Ethernet connection to front panel port 1/2.
	2019	1/3	10 Gbps Fabric Ethernet connection to front panel port 1/3.
	2020	1/4	10 Gbps Fabric Ethernet connection to front panel port 1/4.
To COM Express	2034	2/3	Fabric Ethernet connection to the COM Express module.
To RTM	2049	3/1	Fabric Ethernet connections to the RTM.
	2050	3/2	
	2051	3/3	
	2052	3/4	
To LMP	2065	N/A	Fabric pseudo-Ethernet connection to the LMP dtl1 interface, implemented over PCI.
Link aggregation connections	2066	4/1	Fabric Ethernet link aggregation logical port-channels. These exist only if created by users and are numbered in the order in which they were created.
	2067	4/2	
	2068	4/3	
	2069	4/4	
	2070	4/5	
	2071	4/6	
	2072	4/7	
	2073	4/8	
VLAN routing interfaces	N/A	5/1 to 5/28	Fabric Ethernet VLAN router interfaces. These exist only for VLANs in which routing is enabled. The interfaces are numbered in consecutive order as the routing is enabled for each VLAN.

Pre-defined IP Addressing Scheme

The SFB supports a number of IP interfaces that are used as management interfaces to the module. The interfaces are configured by default to minimize your work in setting them up.

This section describes the default IP addresses, the pre-defined address assignment scheme, and the options for customizing address assignment if necessary.

Important

Some of the default addresses and settings may not be appropriate for the use of the SFB in your network. You may need to modify and enable interface settings in the templates and the scripts to give the SFB the proper network access.

Default IPv4 Addresses

The tables below summarizes the IPv4 interfaces and their default IP addresses.

Interface name	Default IPv4 address/ subnet mask	Description	Interface and IP address usage
lo	127.0.0.1/8	Software loopback interface on the LMP.	IPv4 datagrams sent to this interface by the LMP are received immediately by the LMP. Do not change this address.
eth0	10.0.0.1/24	Ethernet maintenance port labeled LMP ETH on the front panel.	This interface can be used as a maintenance port for the entire ATCA shelf. From an attached remote console, a user can update the firmware and software, and also perform configuration and service of other modules that have IPv4 connectivity This private IPv4 address allows for easy connectivity from a service terminal connected to it, provided that the terminal is also on the 10.0.0.0/24 subnet. If IPv4 Masquerade is enabled, this allows the terminal to easily communicate with other modules beyond the switch fabric, since it translates the private addresses on the 10.0.0.0/24 network to the public or private address assigned on its dtl0 interface.
eth0:0	10.1.<*LCHAS*>.<*SLOT*>/16 (where <*LCHAS*> is the logical chassis and <*SLOT*> is the physical slot)	Subinterface.	This subinterface allows users and validators to use a shared LAN segment to maintain concurrent IPv4 connectivity via eth0:0 subinterfaces. The subinterface also enables multiple peers to be networked together via their eth0 physical interfaces without reconfiguring the eth0 IP addresses.
eth0:1	None, interface not created	Subinterface.	This subinterface is configured through the Base interface CLI using the `serviceport ip` command.
eth1	10.0.1.<*LSLOT*>/24 (where <*LSLOT*> is the logical slot)	PICMG 3.0 update channel to the redundant peer connecting logical slots 1 and 2.	This interface is used by the Shelf Manager to synchronize with its peer. The interface could also be used for management. Changing this address reduces performance of Shelf Manager synchronizations, which then use a slower path.

Interface name	Default IPv4 address/ subnet mask	Description	Interface and IP address usage
dtl0	Dynamically assigned by DHCP	PCI bus connection between the Base switch and the LMP.	This is a pseudo-Ethernet interface.
dtl0:0	10.2.<LCHAS>.<SLOT>/16	Subinterface.	This subinterface can be used to configure.
dtl0:1	<ShelfMgrIPAddr>/24, which is 192.168.16.17/24 by default	Used by the LMP to communicate with the Shelf Manager through the Base interface.	This subinterface is dedicated to providing access to the Shelf Manager server. This subinterface exists only on the peer with the active Shelf Manager, and is enabled and disabled by the Shelf Manager server. This setup allows the system manager to always reach the Shelf Manager, regardless of which is active at a given time. The Shelf Manager IP address is stored in the Shelf FRU Device and can be configured via the HPI example application, *hpiapp*. Updating the Shelf Manager IP address immediately updates the address of dtl0:1, bringing the interface down and then up with the new address.
dtl0:2	None, interface not created	Subinterface.	This subinterface is configured through the Base interface CLI using the `network parms` command.
dtl1	Dynamically assigned by DHCP	PCI bus connection between the Fabric switch and the LMP.	This is a pseudo-Ethernet interface.
dtl1:0	None, interface not created	Subinterface.	This subinterface is configured through the Fabric interface CLI using the `network parms` command.

Static IP Address Assignment Configuration File Setup

The static IP address assignments are determined by configuration files.

The */etc/sysconfig/network-devices* directory contains the configuration files that determine the IP assignments at SFB bootup. Files called *template.ifconfig.<interfaceName>* are interpreted and used to create *ifconfig.<interfaceName>* files. The interpretation process replaces variables with appropriate values. Interfaces that receive literal IP addresses or that use DHCP have just an *ifconfig.<interfaceName>* file and no template file, because there is nothing to interpret.

The IP address configuration files are saved persistently because the /etc directory is bind mounted as a factory default setting.

Variables Used to Assign IP Addresses

Below is explained the use of specific variables to assign static IP addresses.

- **Variable <LCHAS> - logical chassis**

  Usage - The logical chassis ID, or chassis address, can identify the physical location of the chassis within a central office. The ID can also be used to generate unique IP addresses and prompts to help users distinguish between shelves.

  How it is set and obtained - Clavister assigns the default value of 255 to the chassis. During the commissioning process, a chassis should be assigned an address that is unique within a particular central office, or at least within the domain in which the SFB communicates. This information is stored in the FRU Device on the chassis, and is read during host processor initialization. If the address does not conform to the Clavister-defined format, the SFB cannot read it, and the default value is used.

- **Variable <SLOT> - physical slot in chassis**

  Usage - The physical slot location uniquely identifies this SFB's location in the chassis.

  How it is set and obtained - The slot number is read during host processor initialization.

- **Variable <LSLOT> - logical slot**

Usage - The logical slot is an easy way for the SFB to distinguish itself from the other SFB.

How it is set and obtained - The logical slot identifies the position of the SFB in the chassis in relation to the other SFB. The lower-numbered physical slot is logical slot 1. The higher numbered slot is logical slot 2.

Note

The factory default logical chassis address is 255. If two shelves are connected which both have factory default settings, there will be multiple SFBs with the same eth0:0 IP address (for example, 10.1.255.7 for the slot 7 SFB and 10.1.255.8 for the slot 8 SFB). To avoid this, add a chassis to the network and commission it with a new chassis address before adding another chassis.

DHCP Client

Some interfaces are set up to use the DHCP client (*/sbin/dhclient*) to request an IP address from a DHCP server on the network. The chassis address and the physical slot number are used to derive the DHCP client identifier string sent in DHCP client requests.

The client ID is formatted as a colon-delimited hexadecimal string containing the logical chassis and physical slot. The ID always begins with a null character (for example, *00:34:2d:38* for logical chassis 4 and physical slot 8).

To modify the format of the client ID, edit the */etc/sysconfig/network-devices/make-interfaces* script. The client ID appears as the variable CLIENT_ID in */etc/template.dhclient.conf* and is replaced by the actual value in the *dhclient.conf file*.

As the SFB boots up, the DHCP client works in the background requesting IP addresses. By default, the requests from the DHCP client are no more than 5 seconds apart for a total duration of 300 seconds or until an address is received. If no DHCP server has responded to the requests after 300 seconds, a timeout of 5 minutes occurs before the DHCP client starts the request sequence again.

You can configure the values for the maximum request interval, the timeout period, and the length of time before retrying the request sequence in the */etc/template.dhclient.conf file*.

If you have Linux man pages installed on a system, refer to the *dhclient.conf* man page for more information.

Customizing IP Address Assignments

- Addresses to avoid changing:

  It is recommended to avoid changing the following addresses:

  - *lo*
  - *eth1*

  In addition, it may not be necessary to change the other statically configured IP addresses.

-
- Options for customization

  Users who wish to further change the default configuration of the IPv4 interfaces have these choices:

  - Change the */etc/sysconfig/network-devices/template.ifconfig.<interface>* template files to

modify the variables used in assigning addresses.

- Change the */etc/sysconfig/network-devices/ifconfig.<interface>* files to change the static address assignments.
- Change the external DHCP server configuration to change the dynamic address assignments based on the client identifier.
- Change the run level startup script (*/etc/rc.d/init.d/network*) to modify the default behavior for configuring the static and dynamic address assignments.

Users who wish to change the DHCP client behavior have these choices:

- Change the */etc/template.dhclient.conf* files to change the protocol timers and options used by the DHCP client.
- Change the run level startup script (*/etc/rc.d/init.d/network*) to modify the default behavior for launching the DHCP client application.
- Change the DHCP client application itself and the scripts that invoke it.

DHCP Server

The SFB supports the standard Linux DHCP server, which can supply IP addresses for blades and other network equipment on the same network as the SFB.

Telnet and SSH

The SFB supports the standard Linux Telnet server and client to enable connections to and from the SFB. The Telnet server daemon is enabled by default, responding to requests on TCP port 23 and bringing you to the SFB login prompt.

The SFB also supports secure shell (SSH) client and server, along with the rest of the full OpenSSH distribution. SSH allows secure connections to and from the SFB, and requires a password. SSH protocol version 2.0 is supported. The SSH server is enabled by default, responding to requests on TCP port 22.

NTP server

The SFB supports the Linux standard NTP server, which allows it to provide time of day services to other blades or systems. The NTP server is disabled by default.

Module State Management

The SFB provides control over the hot-swap state of any chassis FRU through the platform-management CLI.

Flash and File System Management

The SFB provides CLI commands and an API to help you manage configuration files and other files on the SFB. These help you to:

- Copy Clavister-defined configuration files to new locations, on or off the SFB.
- Retrieve configuration files from another system to the SFB.

- Save the current configuration persistently.
- Reset the configuration to the one in persistent storage (the startup configuration).

The Clavister-defined configuration files are those operated on by the CLI. They cover all configuration performed in the CLI.

Firmware and Software Upgrades

The SFB provides Linux utilities to help you upgrade the firmware and the software. An API can also handle certain aspects of software upgrades, and a software upgrade example application is provided.

Logging Service

The SFB supports the Linux standard syslog service, which allows it to collect messages from various SFB software components and control the output. By default, the module-wide syslog service is disabled, but logging is enabled for several components. Messages from the components are ignored unless the module-wide service is enabled. The module's syslog configuration also determines the destination for the messages.

The Ethernet switching software is an example of a component that enables syslog generation by default. Generation of syslog messages is controlled separately on the Base and Fabric switches. To disable it or re-enable it, the *logging syslog* command can be used.

E-Key Authorization and Notification

The SFB's Shelf Manager provides E-Key authorization for blades to enable and disable the backplane ports between blades and modules (such as AMCs). The E-Key notification handler communicates with the Shelf Manager and notifies interested applications when ports change E-Key states.

For example, the Ethernet switching software needs to know the E-Key state of relevant ports. The Ethernet switching software registers itself with the E-Key notification handler through the internal E-Key API to receive notifications when its switch ports become enabled or disabled. Other software components may also register themselves.

In addition, the Shelf Manager tracks current authorization for the synchronization clock interface using bused E-Key commands, giving permission to blades as appropriate to drive clock signals onto the clock buses.

Chapter 6: SNMP Agent Support

The SFB can be remotely managed by an SNMP agent. The SNMP agent implementation is modular and extensible, dividing the functionality into a master agent and two subagents that represent the Base interface switch and the Fabric interface switch.

The master agent receives requests from the SNMP manager and completes the requested action, coordinating with its subagents when necessary. Notifications generated by the master agent and the subagents are relayed by the master agent to registered destinations. The master agent communicates with the subagents using the AgentX protocol.

Subagents are provided for Ethernet switch management (*switchdrvr*) and for SNMP notification logging (*snmptrapd*).

The SFB implements its SNMP agent by incorporating Net-SNMP open-source software, version 5.3.1.

The SNMP agent currently supports SNMPv2c. SNMPv1 and SNMPv3 are currently implemented.

Base or Fabric Ethernet Switch Selection

The SNMP objects in the *BRIDGE-MIB*, *P-BRIDGE-MIB*, and *Q-BRIDGE-MIB* modules are instantiated separately for the Fabric interface and Base interface switches. For the Fabric interface switch, use community names that map to the "fi" context name to access these objects. To access all other objects, use community names that map to the null context name.

Spanning tree notifications (*newRoot* and *topologyChange*) include the variable dot1dBaseBridgeAddress.0 to indicate the appropriate switch. The variable consists of the 6-byte MAC address used in the switch's bridge priority value. The dot1dBaseBridgeAddress.0 value for the switches is shown in the output of the Ethernet CLI *show spanning-tree brief* command for either the Base or the Fabric interfaces. The last six bytes of the Bridge Priority field determine the value.

MIB Module Support

The table lists the MIB modules supported by the SNMP agent. For each MIB module, the table also references the associated Request for Comments (RFC) document (if applicable), identifies whether the MIB module is supported by the master agent or one of the subagents, and provides the support details for the MIB module.

MIB	RFC document	Supported by	Support details
BRIDGE-MIB	RFC 1493	*switchdrvr* subagent	exceptions
EtherLike-MIB	RFC 2665	*switchdrvr* subagent	exceptions
IEEE8023-LAG-MIB	N/A, IEEE Web site	master agent and *switchdrvr* subagent	exceptions
IF-MIB	RFC 2863	master agent and *switchdrvr* subagent	exceptions
IP-MIB	RFC 2011	master agent	no major exceptions
P-BRIDGE-MIB	RFC 2674	*switchdrvr* subagent	exceptions
Q-BRIDGE-MIB	RFC 2674	*switchdrvr* subagent	exceptions
RFC1213-MIB	RFC 1213	master agent	exceptions
SNMPv2-MIB	RFC 3418	master agent	exceptions
SNMP-FRAMEWORK-MIB	RFC 3411	master agent	no major exceptions
SNMP-MPD-MIB	RFC 3412	master agent	no major exceptions
SNMP-TARGET-MIB	RFC 3413	master agent	no major exceptions
SNMP-NOTIFICATION-MIB	RFC 3413	master agent	no major exceptions
SNMP-USER-BASED-SM-MIB	RFC 3414	master agent	no major exceptions
SNMP-VIEW-BASED-ACM-MIB	RFC 3415	master agent	no major exceptions
TCP-MIB	RFC 2012	master agent	no major exceptions
UDP-MIB	RFC 2013	master agent	no major exceptions

Exceptions

The following is a list of exceptions:

- *BRIDGE-MIB* support

  dot1dStaticTable is not populated.

- *EtherLike-MIB* support

  dot3CollTable is not populated.

- *IEEE8023-LAG-MIB* support

  dot3adAggPortDebugTable is not populated.

- *IF-MIB* support

  ifTestTable is not populated.

- *P-BRIDGE-MIB* support

  The following tables are not populated:

  dot1dUserPriorityRegenTable
  dot1dPortOutboundAccessPriorityTable

- *Q-BRIDGE-MIB* support

  The following are not populated:

  dot1qForwardAllTable
  dot1qForwardUnregisteredTable
  dot1qPortVlanStatisticsTable dot1qPortVlanHCStatisticsTable
  dot1qLearningConstraintsTable

  The following scalars are not populated:

  dot1qConstraintSetDefault
  dot1qContraintTypeDefault

- *RFC1213-MIB* support

  The following objects are not populated:

  egp scalars
  egpNeighTable

  The following tables are supported as originally specified in RFC1213-MIB:

  atTable
  ipRouteTable

  The remaining scalars and tables are supported as re-specified in SNMPv2-MIB, IF-MIB, IP-MIB, TCP-MIB, and UDP-MIB.

- *SNMPv2-MIB* support

  warmStart notification is not supported.

Notification Support

The following table lists the notifications generated by the SNMP agent, the MIB modules in which they are specified, and the part of the SNMP agent that generates them.

Notifications	MIBs	Generated by
authenticationFailure	SNMPv2-MIB	master agent
coldStart	SNMPv2-MIB	master agent
linkDown	IF-MIB	master agent and *switchdrvr* subagent
linkUp	IF-MIB	master agent and *switchdrvr* subagent
newRoot	BRIDGE-MIB	*switchdrvr* subagent
nsNotifyRestart	NET-SNMP-AGENT-MIB	master agent
nsNotifyShutdown	NET-SNMP-AGENT-MIB	master agent
topologyChange	BRIDGE-MIB	*switchdrvr* subagent

MIB Modules Supplied

MIB modules are supplied as text files installed on the SFB's file system and the files include the MIB module name and end with the filetype *.txt*. The modules include:

- BRIDGE-MIB
- EtherLike-MIB
- IANAifType-MIB
- IEEE8023-LAG-MIB
- IF-MIB
- IP-MIB
- NET-SNMP-AGENT
- NET-SNMP-MIB
- NET-SNMP-TC

- P-BRIDGE-MIB
- Q-BRIDGE-MIB
- RFC1155-SMI
- RFC1213-MIB
- RMON-MIB
- SNMP-FRAMEWORK-MIB
- SNMP-MPD-MIB
- SNMP-NOTIFICATION-MIB
- SNMP-TARGET-MIB
- SNMP-USER-BASED-SM-MIB
- SNMP-VIEW-BASED-ACM-MIB
- SNMPv2-CONF
- SNMPv2-MIB
- SNMPv2-SMI
- SNMPv2-TC
- SNMPv2-TM
- TCP-MIB
- UDP-MIB

The text files are located in the */usr/share/snmp/mibs* directory on the SFB's file system. The directory also contains a README.txt file and a zipped archive of all the MIB files. The README.txt file explains how the MIB module files were created. The zipped archive of MIB files facilitates the transfer of files to other hosts.

MIB Browser Utility Applications

The SFB provides MIB browser utility applications from the Net-SNMP package. These applications can simulate an SNMP manager's interaction with the master agent and are installed directly on the SFB in the */usr/bin* directory. If you have the NET-SNMP 5.3.1 man page installed on a system, execute *man snmpcmd* for more information.

The applications include:

- *snmptbulkget*
- *snmpbulkwalk*
- *snmpdelta*
- *snmpget*
- *snmpgetnext*
- *snmpset*

- *snmpstatus*
- *snmptable*
- *snmptest*
- *snmptranslate*
- *snmpwalk*

Chapter 7: The CLI

The Switch Fabric Blade's Command Line Interface (CLI) allows the administrator to interactively configure and monitor a wide range of services. This chapter describes the CLI and provides basic information about using it.

Accessing the Master CLI

To access the CLI:

- Connect to the SFB.
- Login to the CLI.

Once logged in to the CLI, you should see the *ATCA-2210#* prompt.

Displaying available commands and options

To access online help while typing commands, enter a question mark (?). The CLI returns a list of the commands or options available in your current mode. To get a list of available commands, enter a question mark (?):

```
ATCA-2210# ?
base-ethernet   : Start base ethernet CLI
fabric-ethernet : Start fabric ethernet CLI
blade-mgmt      : Blade management configuration and status
linux-shell     : Start the Linux shell
show            : Show configuration and status
platform-mgmt   : Platform management configuration and status
copy            : Copy information to and from the blade
erase           : Erase startup configuration
exit            : Logout from CLI session
logout          : Logout from CLI session
help            : Operation instructions
```

To get help on a specific command, type as much as you know of the command, followed by a question mark. Many commands require multiple arguments, each one shown as the next option when you type a question mark. After entering an option, type a question mark again to see if there are more options. The "<CR>" notation indicates that a carriage return is an option. This means that you have typed all required arguments, and you can press Enter to execute the command.

Command Modes

The top-level (master) CLI is divided into several command modes that give access to different subsets of commands. Some of the modes are documented in detail in this manual, while other modes are documented in manuals that describe those portions of the software in more detail. As indicated from the list of commands above, the master CLI has the following main command modes:

The master CLI also contains some commands that can be executed directly, without entering a mode.

Many of these modes contain additional modes within them. The additional modes are described with the command reference information for each command mode.

To exit any of the modes or the master CLI, enter:

```
exit
```

Mode	Access Method	Prompt	Mode Description
Base Ethernet	base-ethernet	ATCA-2210-base#	Configure the Base Ethernet switch.
Fabric Ethernet	fabric-ethernet	ATCA-2210-fabric#	Configure the Fabric Ethernet switch, including optional IP routing features.
Blade Management	blade-management		Configure the services the module provides, including Linux services, SNMP, and network timing.
Linux Shell	linux-shell	Linux prompt	Access the Linux shell, which is not part of the CLI. When you exit the shell, you return to the CLI.
Platform Mgmt	platform-mgmt	platform-mgmt#	Access shelf management, alarm & FRU information.

"*No*" Commands

Many configuration commands have a "no" form that is used to disable, delete, or reset a configuration back to the factory defaults. For example, the following command administratively disables a port:

```
shutdown
```

The "no" form of this command administratively enables a port:

```
no shutdown
```

"*Show*" Commands

A "show" command displays information about resources. You can use the applicable "show" command to show information for individual resources. The "show" commands are executed outside of configuration modes, usually directly within a main command mode such as Base Ethernet.

Command Line Completion

Pressing the tab key completes a partially typed command keyword. For example, typing a partial command *con* and pressing the tab key completes the keyword *configure*.

Typing a question mark (?) lists valid entries after a command keyword. For example, typing the keyword *configure* and then a question mark brings up a list of valid options that follow

.

Editing and history keys

The editing key sequences are similar to those used by EMACS text editors. Any character typed is inserted into the command line at the current cursor position, and all characters to the right of the typed character shift to the right. The history key sequence is similar to that used by the UNIX C shell. You can obtain a list of key sequences by entering *help* at the main CLI prompt.

Saving configurations persistently

Some CLI configuration changes are kept within the application until you save them to persistent storage. If you do not save, your unsaved changes remain in effect until the module is rebooted, which restores the startup configuration. These methods for saving CLI configurations are currently required for the various CLI command modes:

- **Base Ethernet**
  Two alternatives:
  - Save when exiting the mode, as prompted.
  - From within the mode, use this command:

    ```
    copy system:running-config nvram:startup-config
    ```

- **Fabric Ethernet**
  Same as above.
- **Blade Management**
  From the master CLI, use the command:

  ```
  copy system:running-config nvram:startup-config
  ```

- **Platform Management**
  Not necessary. The configuration is saved persistently when you execute commands.

Master CLI Commands

The master CLI commands are as follows:

- ***show version***

  Shows the software versions of the master CLI and various CLI components.

  ```
  show version
  ```

- ***show running-config***

  Shows the current configuration, displayed as a series of commands necessary to reproduce the current configuration. This command currently shows only the blade-management mode configuration.

  ```
  show running-config
  ```

- ***copy tftp***

Copies the specified file from a TFTP server and makes it the startup configuration in flash memory or the running configuration. This command currently replaces only the blade-management mode configuration.

```
copy tftp:<url> { nvram:startup-config | system:running-config }
```

Options:

nvram:startup-config - Loads the configuration file so it becomes the persistent startup configuration.

system:running-config - Loads the configuration file so it becomes the running configuration.

- **copy system:running-config**

  Saves the current running configuration to persistent storage on the SFB or to another system. If saved to persistent storage, the saved configuration is loaded again when the module is rebooted. This command currently saves only the blade-management mode configuration.

  ```
  copy system:running-config {tftp:<URL> | nvram:startup-config}
  ```

  Options:

  tftp:<URL> - Saves the configuration to another system.

  nvram:startup-config - Saves the configuration to the SFB's non-volatile memory. The saved configuration is reloaded when the SFB is rebooted.

- ***copy nvram:startup-config***

  Copies the persistently saved startup configuration to the specified URL, or reloads it so it becomes the running configuration. This command currently copies only the blade-management mode configuration.

  ```
  copy nvram:startup-config {tftp:<URL> | system:running-config}
  ```

  Options:

  tftp:<URL> - Saves the configuration to another system.

  system:running-config - Reloads the configuration to RAM, making it the current configuration.

- ***erase nvram:startup-config***

  Erases the saved CLI configuration files from persistent storage. This returns aspects of the module that are controlled through the CLI to the factory defaults.

  ```
  erase nvram:startup-config
  ```

  This command currently only affects the blade-management mode configuration.

- ***exit***

  Leaves the current CLI command mode, and goes to the next higher level. If you are already in the master CLI (at the top level), this is the same as the *logout* command, which exits the CLI entirely and logs out from the module.

```
exit
```

- ***logout***

  Exits the CLI and logs out from the module. This command works only from the master CLI (top level).

```
logout
```

- ***help***

  Shows the editing and history keys.

```
help
```

Chapter 8: Blade Management Commands

This section provides reference material for the commands within the blade-management CLI mode. These commands help you enable, disable, configure, and show status of various services that run on the SFB.

Accessing Blade-management Command Modes

To enter blade-management commands:

- Log in to the SCM CLI.
- Access blade-management mode by entering:

```
blade-mgmt
```

Configuration

To access configuration mode from blade-management mode, enter:

```
config
```

The prompt changes each time you change modes.

SNMP commands

The SNMP commands are:

- **service snmpd**

  Enables the SNMP master agent. The *restart* option causes the agent to reset itself by stopping, deleting any log files, and restarting with the current configuration files. The *reconfigure* option forces the agent to re-read the configuration files and change its operation based on their contents, without fully re-initializing.

  Syntax:

```
service snmpd [ restart | reconfigure ]
```

- **no service snmpd**

Disables the SNMP master agent.

Syntax:

```
no service snmpd
```

Note that to check the SNMP agent's status, use the *show running-config* command from the master CLI and check the output for the service *snmpd start* command.

Chapter 9: Maintenance and Troubleshooting

Use this chapter as a general reference when performing maintenance and troubleshooting on the SFB.

Maintenance

If an SFB needs to be installed, removed, or replaced, make sure to follow the ESD precautions

Many of the maintenance procedures can be performed using the SFB's interface management software. The software can be accessed by an external computer using the serial and the Ethernet maintenance ports on the front panel. Use the serial port to perform diagnostic and verification procedures. Use the Ethernet maintenance port to connect to network and to perform configuration procedure.

Obey ESD Precautions

WARNING! Obey the standard Clavister electrostatic discharge (ESD) procedures described in the installation manual when you install or remove the product. Electrostatic discharge can cause permanent damage to static-sensitive components in this product. Important ESD procedures include:

- Keep the product in its ESD shielding bag until a step tells you to remove it.
- Put on a grounded wrist strap before you move near or touch the product.
- Install the product only in a grounded work area.

Installing the SFB

For information on installing or replacing an SFB, see the product's installation guide.

Removing the SFB

To remove the SFB from the chassis:

1. Disconnect all cables from the SCM front panel.
2. Loosen the SCM screws from the shelf.
3. Release the ejector latch in the position noted and then stop:
    - If the SFB is oriented vertically, release the lower ejector latch.

- If the SFB is oriented horizontally, release the right ejector latch.

4. The H/S blue light starts flashing. When the H/S blue light turns solid blue you are ready to remove the SFB.

Note

When only one SFB is present in the chassis and its shelf manager function is enabled, the H/S blue light will never turn solid blue.

5. Release the ejector latch in the position noted:
   a. If the SFB is oriented vertically, release the top ejector latch.
   b. If the SFB is oriented horizontally, release the left ejector latch.
6. Simultaneously pull both extractor latches to release the module from the slot.
7. Pull the SFB out of the slot.
8. Place the SFB on a flat, static-free surface.

Resetting the SFB

If you need to reset the SFB you can push the reset button on the front panel to invoke a payload reset. You can also reset the SFB by issuing payload reset commands using the shelf management software. A payload reset acts as a "global reset". The entire SFB is reset. All circuitry, with the exception of IPMI circuitry, is reset to a known initial state, clearing all remaining bits and initializing dynamic random access memory.

Troubleshooting

When you encounter a situation in which the SFB, or the platform it is installed in, does not perform as you expect, look for symptoms that might clarify the cause. Performing the following actions can aid you in diagnosing symptoms:

- Check the state of the LEDs on all the modules in the platform, especially the power entry modules (PEMs) and rear transition modules (RTMs).
- Check the shelf-management events logged in the system event log (SEL), which is accessible through the system manager. The Shelf Management Software Reference provides details on how to use the SEL.
- Information about the Base Ethernet configuration, which can be generated using the *show* commands in the base-interface management CLI.
- Information about the Fabric Ethernet configuration, which can be generated using *show* commands in the fabric-interface management CLI.
- Verify the IP address and the subnet mask assignment.
- Verify the Fabric Ethernet interface is communicating properly.
- Verify SCMs are of the same type and are using the same software version, when installing two on the same platform for redundancy purposes.

A Checklist of Symptoms

Use the list below for diagnosing your situation. Look for symptoms that apply, then follow the recommended action (or actions) for that symptom. When an action reveals the cause of the problem, resolve the problem as indicated.

- **The power LED on the SCM or another module is not lit.**

  1. Verify that the module is fully inserted.
  2. Inspect the module connector pins for damage. If connector pins show no sign of damage, you can try the following:

     - Carefully insert the module into a different slot.
     - Carefully insert a different module in the original slot.

     Warning
     Do not force the modules into the slots.

     - To avoid damage to connectors, make sure the rear slot is either empty or contains a module that is compatible with the module being inserted.
     - If the module does not slide easily, make sure you are inserting it into the correct slot and that it is aligned properly.

  3. Check to see if the module's Shelf Manager is disabled and whether the module is waiting for an external Shelf Manager to enable it.

- **Communication cannot be established between Ethernet nodes within the platform.**

  Connect to the SFB through the serial (LMP SER) or the Ethernet maintenance (LMP ETH) ports. Use base-interface management CLI commands to diagnose the cause. In particular, verify that an IP address has been assigned to the node, an appropriate subnet mask is assigned to the node, the node is within a reachable VLAN, and routing instructions do not prevent the communication. If switch control CLI commands fail to work with one of the nodes, check the system event log to verify that the node's Ethernet backplane interface was activated.

- **The SCM is in debug mode according to the login banner.**

  Restore valid Shelf FRU Information using the frurw utility.

- **A module does not work correctly.**

  Check the system event log for significant events related to the module. In particular, verify that the module worked correctly when it was installed, and look for any events since then that would account for why the module stopped working correctly. If the log information does not reveal useful symptoms, power down the module, slide it out, perhaps move it to a different slot, slide it back in, and power it back up. If appropriate, install a different module of the same kind to help determine whether the module might be defective.

- **Intermittently, the SCM and other installed modules experience random data errors.**

  Verify that the platform's frame-ground connection is properly connected to a high-quality earth-ground connection. Check for electrical noise at the backplane power connections and at the power entry module power inputs. Consider the possibility of a malfunctioning module causing electrical noise on backplane connections.

- **The SCM or another module overheats**

  1. Verify no cover plates are installed. Empty slots must have air management panels rather

than cover plates installed to properly maintain airflow and emissions.

2. Use shelf management software to check temperatures at the air intake, on the module, and at the platform's air exhaust. Use the information to determine whether the overheating may be caused by warm facility air, a module failure, or a failed fan module.

3. Try moving the module to a different slot to see if that resolves the overheating.

4. Verify there is at least two inches of clearance between the side of the shelf and the side of the rack cabinet.

5. Check the air filter for obstructions and dirt.

Chapter 10: Specifications

- **Temperature (ambient)**

State	Value
Operating (with maximum one fan fault)	+5° C to +45° C 30° C/hr rate of change
Short-term Operating (maximum 96 hours operation)	–5° C to +55° C 30° C/hr rate of change
Storage	–40° C to +70° C Rates of change: 23° C to -40° C at 30° C/hr -40° C to 23° C at 13° C/min +23° C to +70° C at 30° C/hr +70° C to +23° C at 10° C/min

- **Relative humidity**

State	Value
Operating	5% to 85% RH non-condensing
Short-term operating (maximum 96 hours operation)	5% to 90% RH non-condensing at +30° C
Storage	5% to 90% RH non-condensing at +40° C
Short-term storage (maximum 96 hours operation)	5% to 95% RH non-condensing at +40° C

- **Altitude**

  (Operating for maximum 8 hours.)

  - Up to 1800 meters (5,905 feet), +55° C
  - > 1800 meters up to 4,000 meters (13,123 feet), derated linearly to +45° C

- **Shock (drop)**

State	Value
Unpacked (free fall, corners & edges)	0 to < 10kg = 100 mm drop 10 to < 25 kg = 75 mm drop
Packaged, Unpalletized (free fall, corners & edges)	0 to < 10kg = 750 mm drop 10 to < 25 kg = 600 mm drop

State	Value
Palletized	300 mm free fall drop

- **Vibration**

  (In each direction for each of three mutually perpendicular axes.)

State	Value
Operating	0.1g, 5 to 100 Hz and back, 0.1 octave/min sine sweep
Transportation (packaged)	0.5g, 5 to 50 Hz and back, 0.1 octave/min sine sweep 3.0g, 50 to 500 Hz and back, 0.25 octave/min sine sweep

- **Seismic**

State	Value
Operating	Per Zone 4 test method, GR-63-CORE

Safety

The safety specifications are measured with ambient temperature approximately 25° C and relative humidity between 30% and 50%. Testing has been performed in partnership with a nationally recognized testing laboratory (NRTL) accredited to provide the required certifications.

Characteristic	Certification	Standard and test criteria
US	Accessory Listing	UL 60950-1 "Safety for Information Technology Equipment"
Canada	Approval	CSA 22.2 #60950-1 "Safety for Information Technology Equipment"
EU	Conformance with the Low Voltage Directive	EN 60950-1 "Safety for Information Technology Equipment"
Other	CB Report	IEC 60950-1 "Safety for Information Technology Equipment"

Mechanical

Characteristic	Value
Dimensions	322.25 mm x 280.0 mm +0, -0.3 mm (12.687" x 11.023" +0.0, -0.012")
Board thickness	2.0 mm ± 0.2 mm (0.079" ± 0.007")

Electromagnetic compatibility (EMC)

The product has been tested and found to comply with the requirements detailed in the following standards when installed in a representative chassis:

- ICES-003 Class A.
- FCC Part 15 Class A.
- EN 55022:1998 Class A.
- EN 300 386 V1.3.3.
- EN 55024:1998 + A1 + A2.

Warning

This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy, and if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

The interface ports are designed for use as intra-building interfaces only (Type 2 or Type 4 ports as described in GR-1089-CORE, Issue 4) and require isolation from the exposed Outside Plant (OSP) cabling. The addition of Primary Protectors is not sufficient protection in order to connect these interfaces metallically to OSP wiring.

Power

SFB Power Emissions

Characteristic	State	Standard and Criteria
Nominal operating voltage	-48V	ETS 300 132-2 Static voltage levels

SFB Power Consumption Summary

Supply	Amps	Watts
Total converted from 12V		86.9
DC/DC Conversion Loss (85%)		15.3
COM Express 12V	4.47	26.0
RTM 12V	2.8	33.6
Total 12V brick power output		161.8
Brick Conversion Loss (93%)		12.2
IPMC and RTM 3.3V & 2.5V		3.0
Total PIM output		177.0
PIM Loss (97%)		5.5%
Total -48V Power Required	-	182.5

Reliability

The reported failure rates for the SFB do not represent catastrophic failure. Catastrophic failure rates will vary based on application environment and features critical to the intended function.

- Failure rate (Fit): *8849.558 failures in 109 @ 55 C°.*
- Mean time between failures (MTBF): *113,000 hours @ 35 C°.*

Clavister AB
Sjögatan 6J
SE-89160 Örnsköldsvik
SWEDEN

Phone: +46-660-299200
Fax: +46-660-12250

www.clavister.com