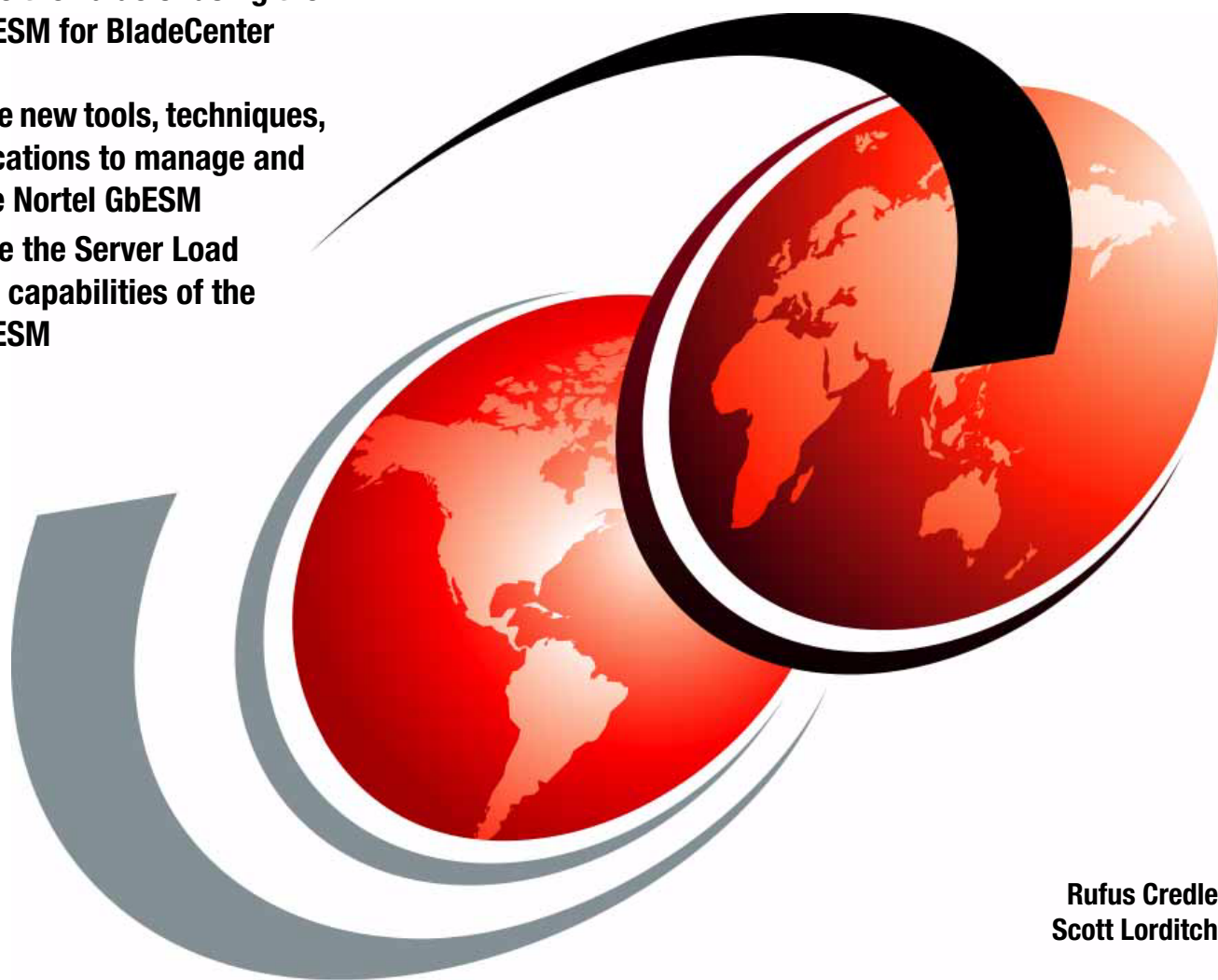


Application Switching with Nortel Networks Layer 2-7 Gigabit Ethernet Switch Module for IBM BladeCenter

Experience the value of using the Nortel GbESM for BladeCenter

Experience new tools, techniques, and applications to manage and deploy the Nortel GbESM

Experience the Server Load Balancing capabilities of the Nortel GbESM



Rufus Credle
Scott Lorditch



International Technical Support Organization

**Application Switching with Nortel Networks Layer 2-7
Gigabit Ethernet Switch Module for IBM BladeCenter**

March 2006

Note: Before using this information and the product it supports, read the information in “Notices” on page vii.

First Edition (March 2006)

This edition applies to IBM @server BladeCenter, Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter, Citrix MetaFrame 4.0, WebSphere Portal 5.1, and VMware ESX 2.5.1.

This document created or updated on March 9, 2006.

© Copyright International Business Machines Corporation 2006. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	vii
Trademarks	viii
Preface	ix
The team that wrote this Redpaper	ix
Become a published author	x
Comments welcome	x
Chapter 1. Executive summary	1
1.1 Product overview	2
Chapter 2. IBM BladeCenter overview	3
2.1 The IBM BladeCenter product family	4
2.1.1 IBM BladeCenter storage solutions	5
2.1.2 IBM BladeCenter system management	5
2.2 IBM BladeCenter architecture	6
2.2.1 The midplane	6
2.2.2 Management Module Ethernet	7
2.2.3 Gigabit Ethernet path	8
2.3 IBM eServer HS20 architecture	10
2.4 Stand-alone configuration tools	12
Chapter 3. Introduction to Nortel Networks	
Layer 2-7 Gigabit Ethernet Switch Module	15
3.1 Value proposition	16
3.2 Architecture	16
3.3 GbESM: Capabilities, features, and functions	17
3.4 Switch management and operating concepts	21
3.4.1 Switch management and control	21
3.5 Ports and performance features	22
3.6 Switch and network management	22
3.7 Network cables	23
3.8 Supported network standards	23
3.9 Layer 4-7 switching functions	24
3.9.1 Key benefits of Layer 4-7 Switching	24
3.10 Comparison to L2/3 switch module	25
Chapter 4. Integrating the L2-7 Switch Module into a network	27
4.1 Nortel Networks L2/7 GbESM management connectivity	28
4.1.1 Out-of-band management	29
4.1.2 In-band management	30
4.2 Nortel Networks L2/7 GbESM user interface	32
4.2.1 IBM BladeCenter Management Module and I2C	32
4.2.2 Command-line interface	32
4.2.3 Browser Based Interface	40
4.2.4 SNMP management: IBM Director	41
4.3 Multiple Nortel Networks L2/7 GbESMs in a BladeCenter	42
4.4 Differences with L2/3 switch module	42
4.4.1 Functions unique to the L2/7 GbESM Switch Module	42

4.4.2	Functions not included on the L2-7 GbESM switch module	42
4.4.3	Functions implemented differently on the L2/3 and L2-7 switches	44
Chapter 5.	Introduction to server load balancing	47
5.1	L4-7 implementation requirements	48
5.2	Layer 4 switching: How it works	48
5.3	Layer 7 Switching: How it works	51
5.4	Server health checking	54
5.5	Advanced server load-balancing functions	56
5.5.1	Persistence	56
5.5.2	Load balancing metrics	58
5.5.3	Key configuration parameters	60
5.6	Design examples and best practices	61
5.6.1	Definitions	61
5.6.2	Minimum configuration required for SLB	61
5.6.3	Multiple services examples and considerations	62
5.6.4	SLB across multiple BladeCenter chassis	64
5.7	High availability design considerations	65
5.7.1	Introduction to Hot Standby and Trunk Failover	65
5.7.2	Introduction to NIC Teaming	66
5.7.3	Configuration of Trunk Failover	67
5.7.4	Configuration of Hot Standby	68
5.7.5	Introduction to VRRP	69
5.7.6	Some important rules for ensuring High Availability	70
5.8	Additional functions	71
5.8.1	Filters	71
5.8.2	Network Address Translation	72
5.8.3	SYN attack mitigation	72
Chapter 6.	Load balancing with WebSphere Portal	75
6.1	Introduction to WebSphere Portal	76
6.2	Value of load balancing with WebSphere Portal	77
6.3	Implementing load balancing with WebSphere Portal	78
6.3.1	Configuration examples	79
6.3.2	Time-out configuration issues	99
Chapter 7.	Load balancing with Citrix MetaFrame and Microsoft Terminal Services	101
7.1	Value of load balancing with Microsoft Terminal Services	102
7.2	Value of load balancing with Citrix MetaFrame	102
7.3	Implementing load balancing with Citrix MetaFrame	102
7.3.1	Functions that can be load balanced	103
7.3.2	GbESM Configuration for Citrix and Terminal Server	108
7.3.3	Persistence and timing considerations	109
7.3.4	Load balancing additional Citrix services	110
Chapter 8.	Load balancing with VMware	115
8.1	Value of load balancing with VMware	116
8.2	Implementing load balancing with VMware	116
8.2.1	Preparation	116
8.2.2	Sample configuration files	118
8.2.3	Using multiple VLANs	125
8.2.4	Diagnosis and troubleshooting	128

Appendix A. Filters on L2/3 and L2/7	131
Appendix B. Workaround for VMware use of VLAN tags	133
Related publications	135
IBM Redbooks	135
Other publications	135
Online resources	135
How to get IBM Redbooks	136
Help from IBM	136
Index	137

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

BladeCenter®
Domino®
Electronic Service Agent™
Enterprise Storage Server®
@server®
eServer®

eServer™
IBM®
IntelliStation®
Redbooks (logo) ™
Redbooks™
ServerGuide™

Tivoli®
TotalStorage®
WebSphere®
xSeries®

The following terms are trademarks of other companies:

Java, JVM, Streamline, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Excel, Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

VMware, the VMware "boxes" logo, GSX Server, ESX Server, Virtual SMP, VMotion and VMware ACE are trademarks (the "Marks") of VMware, Inc.

Citrix, the Citrix logo, Citrix ICA, Citrix MetaFrame, Citrix MetaFrame XP, Citrix Nfuse, Citrix Extranet, Citrix Program Neighborhood, Citrix WinFrame, and other Citrix product names referenced herein are registered trademarks or trademarks of Citrix Systems, Inc. in the United States and other jurisdictions.

Preface

This IBM® Redpaper positions the Nortel Networks Layer 2-7 GbE Switch Modules for IBM eServer BladeCenter and describes how its integrated switch options enable the consolidation of full Layer 2-7 LAN switching and routing capabilities.

This Redpaper serves as an update to the IBM eServer BladeCenter® Layer 2-7 Network Switching, REDP-3755-00. Here, we provide more discussion on the Nortel Networks Layer 2-7 GbE Switch Modules for IBM eServer BladeCenter set of features and services. In particular, we discuss the L2-7 GbESM being a fully-functioning content and load balancing switch with capabilities equivalent to products offered as free-standing appliances. Load balancing using the Nortel GbESM was demonstrated (but not limited to) utilizing the following applications and virtual machines: Citrix MetaFrame/Terminal Services, IBM WebSphere® Application Server/IBM WebSphere Portal, and VMware ESX. However, note these applications were used as examples.

In this Redpaper, we discuss tools, techniques, and applications that help with the management and deployment of the Nortel GbESM in an IBM BladeCenter. We also discuss the management paths and rules for connecting to and accessing the Nortel GbESM.

The team that wrote this Redpaper

This Redpaper was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Rufus Credle is a Certified Consulting I/T Specialist and certified Professional Server Specialist at the ITSO, Raleigh Center. He conducts residencies and develops IBM Redbooks™ and Redpapers that discuss network operating systems, ERP solutions, voice technology, high availability and clustering solutions, Web application servers, pervasive computing, and IBM and OEM e-business applications, all running on IBM xSeries® and IBM BladeCenter technology. Rufus's various positions during his IBM career have included assignments in administration and asset management, systems engineering, sales and marketing, and IT services. He holds a BS degree in business management from Saint Augustine's College. Rufus has been employed at IBM for 25 years.

Scott Lorditch is a Sales Network Architect for the Blade Switching Server business unit of Nortel Networks. He develops designs and proposals for customers and potential customers of the Nortel Networks GbESM products for the IBM BladeCenter, including overall network architecture assessments. He also has developed several training and lab sessions for IBM technical and sales personnel and has provided field feedback to the product team. His background before working for Nortel includes almost 20 years working on networking, including electronic securities transfer projects for a major bank based in New York City, as Senior Network Architect for a multi-national soft drink company, and as Product Manager for managed hosting services for a large telecommunications provider. He holds a BS in Operations Research with specialization in Computer Science from Cornell University.

Thanks to the following people for their contributions to this project:

Tamikia Barrow, Jeanne Tucker, Margaret Ticknor
International Technical Support Organization, Raleigh Center

Jere Dancy, Dipak Shah, Smitha Velagapudi, Hunter Tweed, Sunil Bhatnagar - WebSphere Portal Level-2 Support
IBM Research Triangle Park

Rob Boretti, Collaboration Center Analyst - WebSphere Edge / IHS
IBM Research Triangle Park

Fred Rabert
Nortel Networks

Rob Harper, WW Business Development Executive-IBM
Citrix Systems, Inc.

Mike Ballengee
Citrix Systems, Inc.

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our papers to be as helpful as possible. Send us your comments about this Redpaper or other Redbooks in one of the following ways:

- ▶ Use the online **Contact us** review redbook form found at:

ibm.com/redbooks

- ▶ Send your comments in an email to:

redbook@us.ibm.com

- ▶ Mail your comments to:

IBM Corporation, International Technical Support Organization
Dept. HQ7 Building 662
P.O. Box 12195
Research Triangle Park, NC 27709-2195



Executive summary

This chapter provides a short overview of the IBM and Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter.

1.1 Product overview

IBM and Nortel Networks are committed to collaborate on the design and development of server and networking technology to address client requirements by establishing a joint development center. The Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter (Nortel GbESM) represents a new height in this alliance. This BladeCenter switch module offers BladeCenter customers Nortel's latest L2-7 GbE switching technology integrated into the BladeCenter chassis. It further enhances the BladeCenter value proposition by seamlessly interfacing to a client's existing data network using six external multimode fiber or copper GbE interfaces.

When installed in the BladeCenter chassis, the Nortel GbESM provides both full L2 switching, L3 routing, application health checking, network and application load balancing, and embedded security. The Nortel GbESM provides significant added value not found in commodity switching solutions. This value includes:

- ▶ VLAN tagging: 802.1Q
- ▶ Link Aggregation and LACP: 802.3ad and 802.3-2002
- ▶ Spanning Tree: 802.1D, 802.1w, 802.1s
- ▶ Routing Information Protocol: RFC1058 and RFC2453
- ▶ Open Shortest Path First (OSPF): RFC1257, RFC2328, and others
- ▶ Virtual Router Redundancy Protocol (VRRP): RFC 3768
- ▶ Remote Management Protocol: Mini-RMON MIB (RFC 1757)
- ▶ Internet Engineering Task Force (IETF) standard SNMP management: (MIB) (RFC 1493) and MIB-II (RFC 1213)

Each Nortel GbESM provides one Gigabit/sec Ethernet (GbE) connectivity to each of the 14 blade slots and four GbE uplink interfaces external to the IBM BladeCenter. The client can install as few as one Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter or as many as four in one BladeCenter chassis. With four Nortel GbESMs installed, the client can obtain 16 GbE uplink interfaces, as well as 56 GbE internal switching capability. The flexibility of the Nortel GbESM allows customers to address a variety of performance and redundancy needs.

The Nortel and IBM agreement to form a joint development center will equip Nortel as it becomes an On Demand company able to generate customized products for its network equipment marketplace. This ensures that client needs of high availability, scalability, security, and manageability will be addressed. Combined with the integration of IBM Tivoli®, Nortel, and Cisco management products, these architectures achieve higher value solutions with lower operational expense for clients. The Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter is an integral part of these solutions. With the Nortel GbESM, the client has the investment protection and price performance of a solution the world's leading server and networking companies stand behind.



IBM BladeCenter overview

IBM designed the IBM BladeCenter innovative modular technology, leadership density, and availability to help solve a multitude of real-world issues.

For organizations seeking server consolidation, the IBM BladeCenter centralizes servers for increased flexibility, ease of maintenance, reduced cost, and streamlined human resources. Companies that need to deploy new e-commerce and e-business applications can achieve speed while ensuring flexibility, scalability, and availability. For enterprise requirements such as file-and-print and collaboration, the IBM BladeCenter is designed to offer reliability, flexibility for growth, and cost effectiveness. In addition, clients with compute-intensive applications that need highly available clustering can use the IBM BladeCenter to help achieve high degrees of scalability and performance.

This chapter provides a high-level overview of the IBM BladeCenter product family and how it interfaces with the Ethernet Switch Module.

2.1 The IBM BladeCenter product family

The IBM BladeCenter family of products features a modular design that integrates multiple computing resources into a cost-effective, high-density enclosure for a platform that:

- ▶ Reduces installation, deployment, and redeployment time
- ▶ Reduces administrative costs with our helpful management tools
- ▶ Achieves the highest levels of availability and reliability
- ▶ Provides XpandonDemand scale-out capability
- ▶ Reduces space and cooling requirements compared to 1U solutions

To understand more about how the Nortel Networks Layer 2/3 GbE Switch Module is designed to operate in the BladeCenter chassis, we suggest that you read the sections that follow which discuss the BladeCenter architecture. If you seek to know more about the IBM BladeCenter and its components, visit:

<http://www.ibm.com/products/us/>

Figure 2-1 on page 5 shows the IBM BladeCenter chassis, HS40, HS20, JS20, and LS20:

- ▶ IBM BladeCenter chassis

The BladeCenter is a high-density blade solution that provides maximum performance, availability, and manageability for application serving, storage flexibility, and long-life investment protection.

- ▶ HS40

HS40 is a 4-way blade server for high-performance enterprise applications requiring four-processor SMP capability. The BladeCenter chassis supports up to seven 4-way servers and is ideal for Enterprise Resource Planning (ERP) and database applications.

- ▶ HS20

The IBM efficient 2-way blade server design offers high density without sacrificing server performance. Ideal for Domino®, Web server, Microsoft® Exchange, file and print, application server, and so on.

- ▶ JS20

JS20 is a 2-way blade server for applications requiring 64-bit computing. Ideal for compute-intensive applications and transactional Web serving.

- ▶ LS20

LS20 is a 2-way blade server running AMD Opteron processors. The LS20 delivers density without sacrificing processor performance or availability. For applications that are limited by memory performance, the LS20 might bring sizeable performance gains.

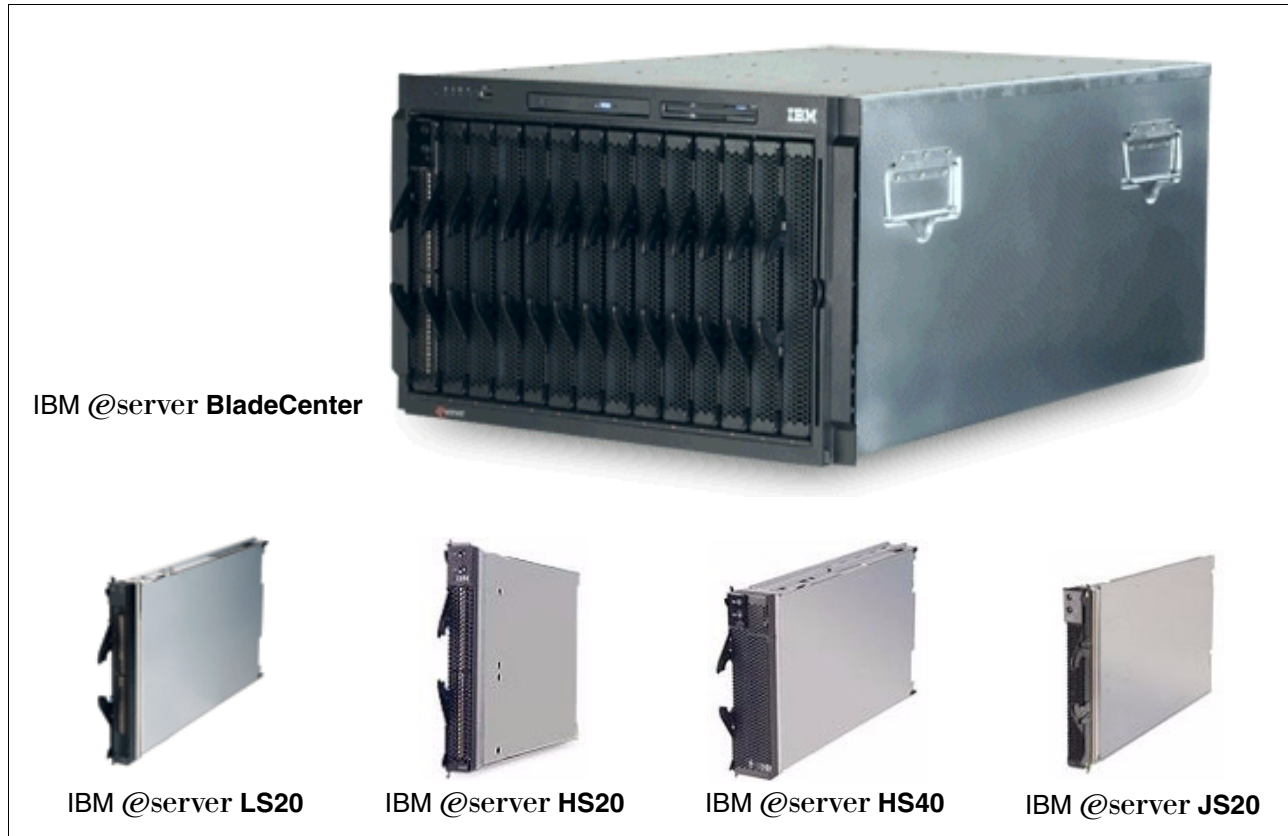


Figure 2-1 IBM @server® BladeCenter and blade modules

Blade development is ongoing for the BladeCenter platform. Therefore, we suggest that you regularly visit this Web site for the latest information about IBM BladeCenter:

<http://www.ibm.com/servers/eserver/bladecenter/index.html>

2.1.1 IBM BladeCenter storage solutions

IBM delivers a wide range of easy-to-install, high-capacity, tested storage products for the IBM BladeCenter to meet your demanding business needs. This enables you to choose from the array of IBM TotalStorage® storage solution products, which include:

- ▶ Fibre Channel products and Storage Area Networks
- ▶ Network Attached Storage
- ▶ Enterprise Storage Server®

IBM TotalStorage provides connected, protected, and complete storage solutions that are designed for your specific requirements, helping to make your storage environment easier to manage, helping to lower costs, and providing business efficiency and business continuity.

For more information about BladeCenter storage solutions, visit:

<http://www.pc.ibm.com/us/eserver/xseries/storage.html>

2.1.2 IBM BladeCenter system management

To get the most value from your IBM BladeCenter investment throughout its life cycle, you need smart, effective systems management which will keep your availability high and costs low.

Management foundation

IBM Director, our acclaimed industry standards-based workgroup software, delivers comprehensive management capability for IntelliStation®, ThinkCentre, ThinkPad, and IBM BladeCenter and xSeries hardware to help reduce costs and improve productivity. IBM Director is hardware that is designed for intelligent systems management. It offers the best tools in the industry and can save you time and money by increasing availability, tracking assets, optimizing performance, and enabling remote maintenance.

Advanced server management

This exclusive collection of software utilities provides advanced server management and maximum availability through the following components:

- ▶ Server Plus Pack
- ▶ Application Workload Manager
- ▶ Scalable Systems Manager
- ▶ Real-Time Diagnostics
- ▶ Electronic Service Agent™
- ▶ Tape Drive Management Assistant

For more information about advanced server management, see:

http://www-1.ibm.com/servers/eserver/xseries/systems_management/xseries_sm.html

Deployment and update management

IBM deployment tools help minimize the tedious work that can be involved in getting your servers and clients ready to run. These tools include:

- ▶ Remote Deployment Manager
- ▶ Software Distribution Premium Edition
- ▶ ServerGuide™
- ▶ ServerGuide Scripting Toolkit
- ▶ UpdateXpress

For more information about IBM BladeCenter deployment and update management, visit:

http://www.ibm.com/servers/eserver/xseries/systems_management/xseries_sm.html

2.2 IBM BladeCenter architecture

In this section, we look into the architectural design of the IBM BladeCenter chassis and its components.

2.2.1 The midplane

Figure 2-2 on page 7 illustrates the BladeCenter midplane. The midplane has two similar sections (upper and lower) that provide redundant functionality. The processor blades (blade servers) plug into the front of the midplane. All other major components plug into the rear of the midplane (for example, power modules, switch modules, and management modules). The processor blades have two connectors, one that is connected to the upper section and one that is connected to the lower section of the midplane. All other components plug into one section only (upper or lower). However, there is another matching component that can plug into the other midplane section for redundancy.

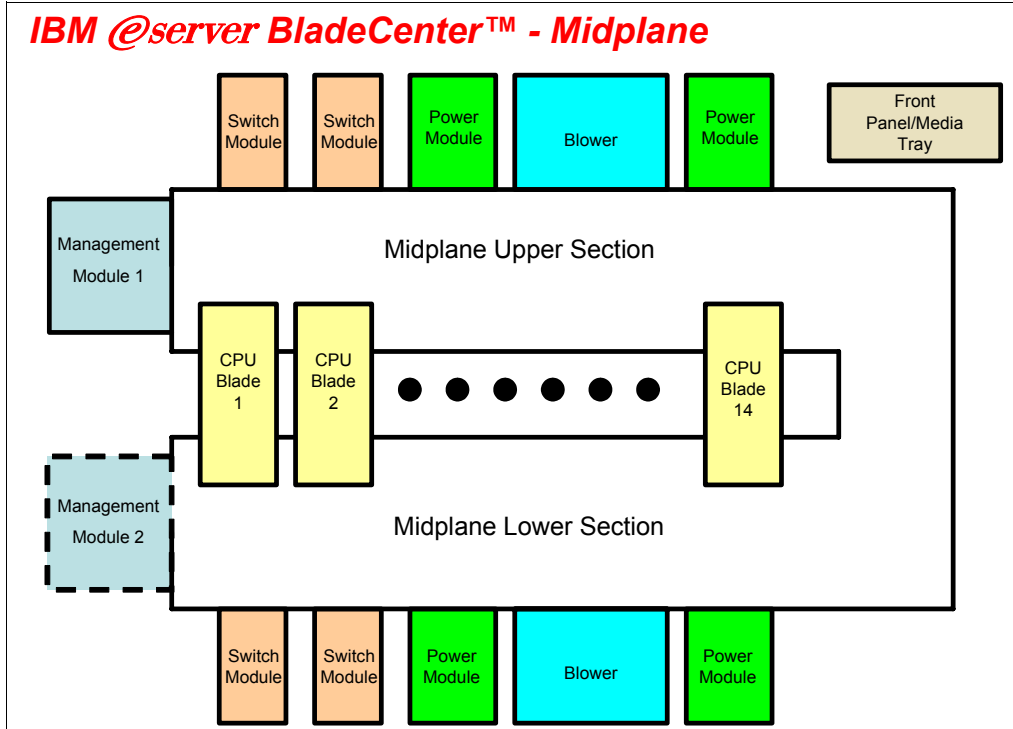


Figure 2-2 Midplane view

It should be noted that the upper and lower midplane sections in an IBM BladeCenter are independent of each other (see Figure 2-3). Having a dual midplane ensures that there is no single point of failure and the blades remain operational.

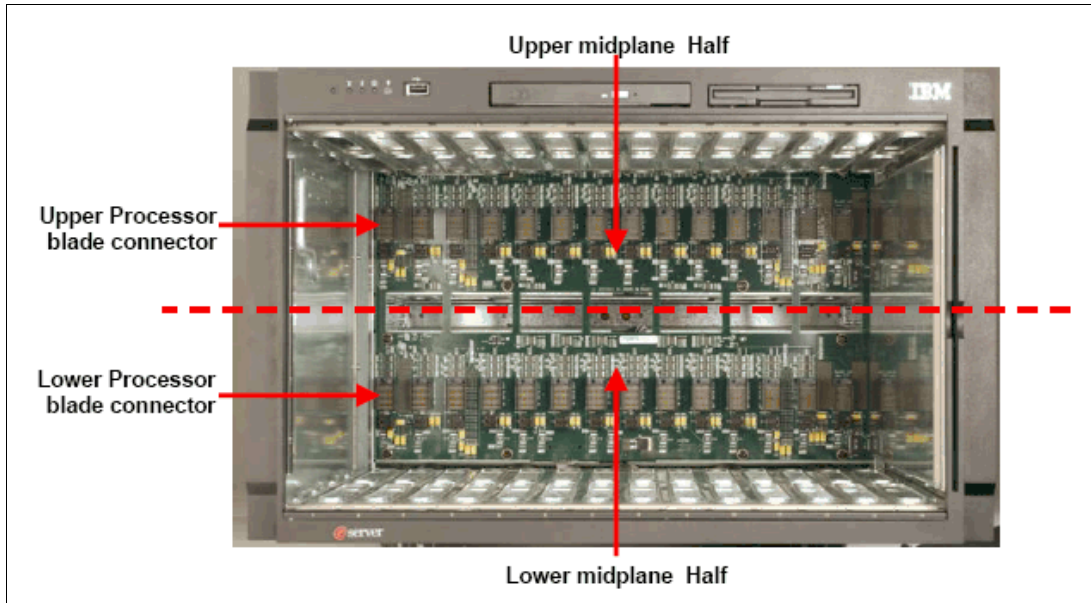


Figure 2-3 Internal picture of the upper and lower midplane of the BladeCenter chassis

2.2.2 Management Module Ethernet

Figure 2-4 illustrates the Management Module Ethernet interface. The switch modules are configured by the active Management Module through the use of a 100 Mb Ethernet interface.

Each Management Module has four 100 Mb Ethernet interfaces, one for each switch module. Each switch module has two 100 Mb Ethernet interfaces, one for each Management Module.

Note: On the Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter, the management Ethernet ports on the switch are referred to as MGT1 and MGT2. For more information beyond this generic illustration, see Chapter 3, “Introduction to Nortel Networks Layer 2-7 Gigabit Ethernet Switch Module” on page 15.

The following list clarifies the routing:

- ▶ Management Module 1 Ethernet 1 → Switch Module 1 Ethernet MGT1
- ▶ Management Module 1 Ethernet 2 → Switch Module 2 Ethernet MGT1
- ▶ Management Module 1 Ethernet 3 → Expansion Switch Module 3 Ethernet MGT1
- ▶ Management Module 1 Ethernet 4 → Expansion Switch Module 4 Ethernet MGT1
- ▶ Management Module 2 Ethernet 1 → Switch Module 1 Ethernet MGT2
- ▶ Management Module 2 Ethernet 2 → Switch Module 2 Ethernet MGT2
- ▶ Management Module 2 Ethernet 3 → Expansion Switch Module 3 Ethernet MGT2
- ▶ Management Module 2 Ethernet 4 → Expansion Switch Module 4 Ethernet MGT2

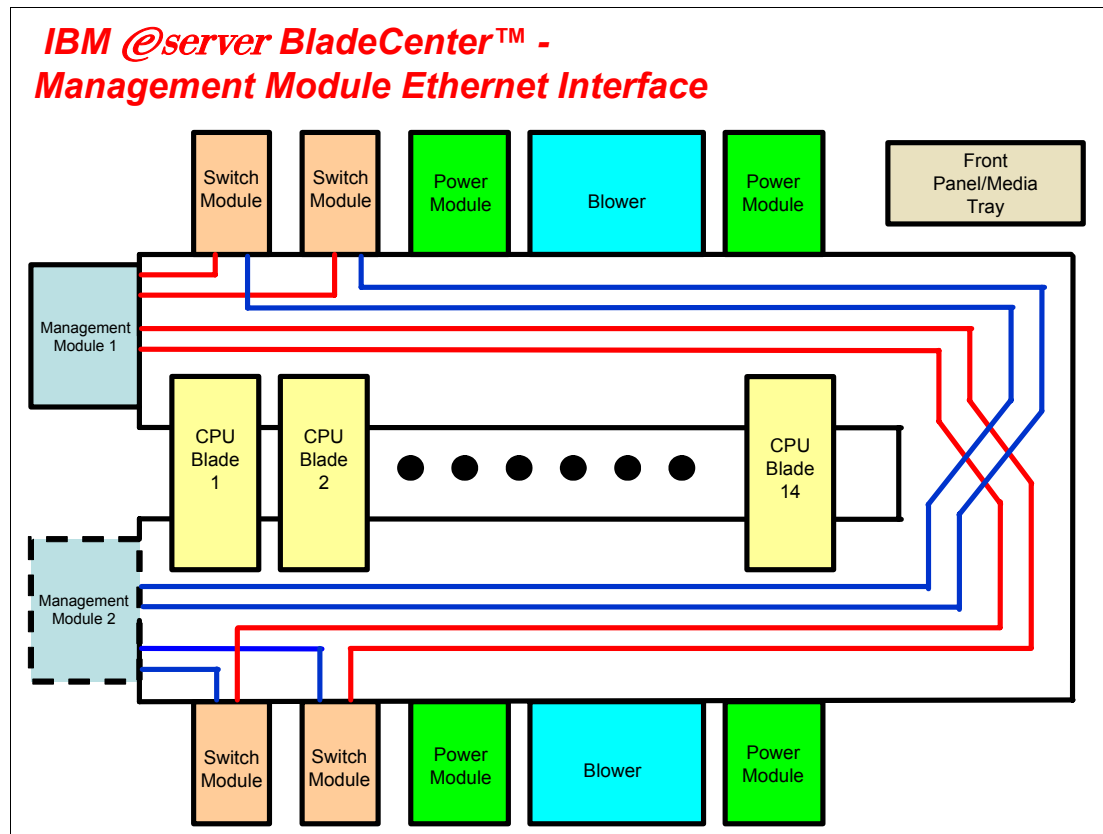


Figure 2-4 Management Module Ethernet interface

The redundant paths of the Management Module Ethernet interface are run from Management Module 2.

2.2.3 Gigabit Ethernet path

Figure 2-5 on page 10 illustrates the Gigabit Ethernet path. Each processor blade has a minimum of two and a maximum of four EtherLAN interfaces. In particular, the BladeCenter HS20 processor blade has two serializer/deserializer SERDES-based Gb Ethernet interfaces,

one for each midplane connector. With a daughter card installed, two more network interfaces can be added. Each switch module (SW Module) receives one LAN input from each processor blade, for a total of 14 inputs.

Note: On the Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter, the internal Ethernet ports on the switch are referred to as MGT1 and MGT2. For more information beyond this generic illustration, see Chapter 3, “Introduction to Nortel Networks Layer 2-7 Gigabit Ethernet Switch Module” on page 15.

The following partial listing illustrates the routing:

- ▶ Processor blade 1 LAN 1 → Switch Module 1 input INT1
- ▶ Processor blade 1 LAN 2 → Switch Module 2 input INT1
- ▶ Processor blade 1 LAN 3 → Expansion Switch Module 3 input INT1
- ▶ Processor blade 1 LAN 4 → Expansion Switch Module 4 input INT1
- ▶ Processor blade 2 LAN 1 → Switch Module 1 input INT2
- ▶ Processor blade 2 LAN 2 → Switch Module 2 input INT2
- ▶ Processor blade 2 LAN 3 → Expansion Switch Module 3 input INT2
- ▶ Processor blade 2 LAN 4 → Expansion Switch Module 4 input INT2

On processor blade, LAN 1 and LAN 2 are the on-board SERDES Gbit Ethernet interfaces, and are routed to Switch Module 1 and Switch Module 2, respectively, for every processor blade. LAN 3 and LAN 4 go to the Expansion Switch Modules 3 and 4, respectively, and are only to be used when a daughter card is installed. Unless a daughter card is installed in one or more processor blades, there is no need for Switch Modules 3 and 4. Further, the switch modules have to be compatible with the LAN interface generated by the processor blade. If a Fibre Channel daughter card is installed in a BladeCenter HS20 processor blade, Switch Modules 3 and 4 must also be Fibre Channel-based, and any daughter cards installed in the remaining BladeCenter HS20 processor blades must be Fibre Channel.

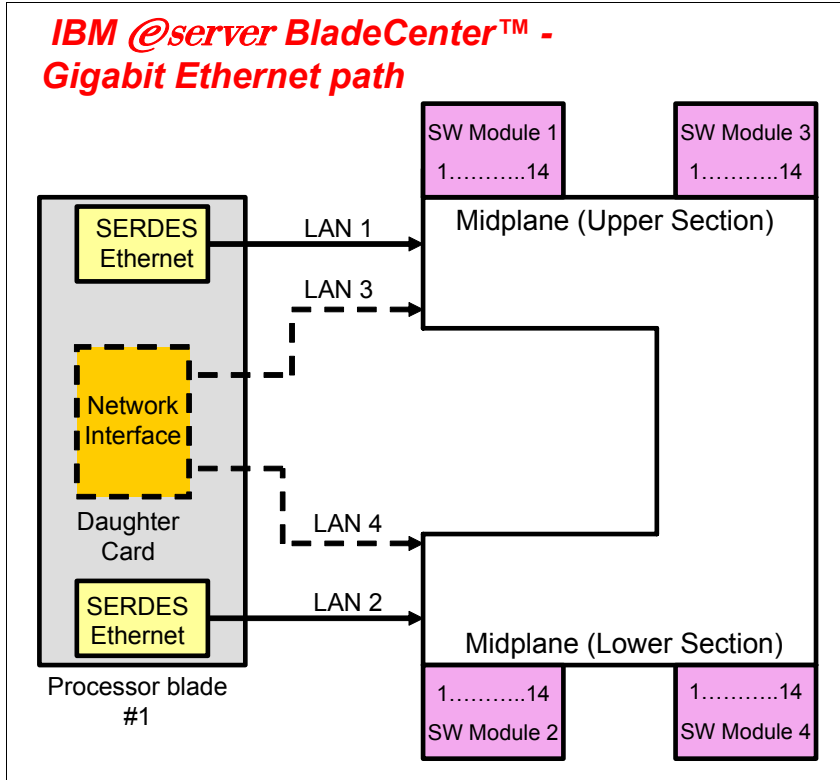


Figure 2-5 Gigabit Ethernet path

2.3 IBM eServer HS20 architecture

In this section, we discuss the architectural design of the IBM BladeCenter HS20. This is presented as just one example of the blade design for a typical dual-processor server. The BladeCenter HS20 uses the Intel® Lindenhurst chipset. See the HS20 architecture in Figure 2-6 on page 11.

8843 HS20 Block Diagram

Servicing the IBM @server HS20 (M/T 8843) and Blade Storage Expansion-II Option

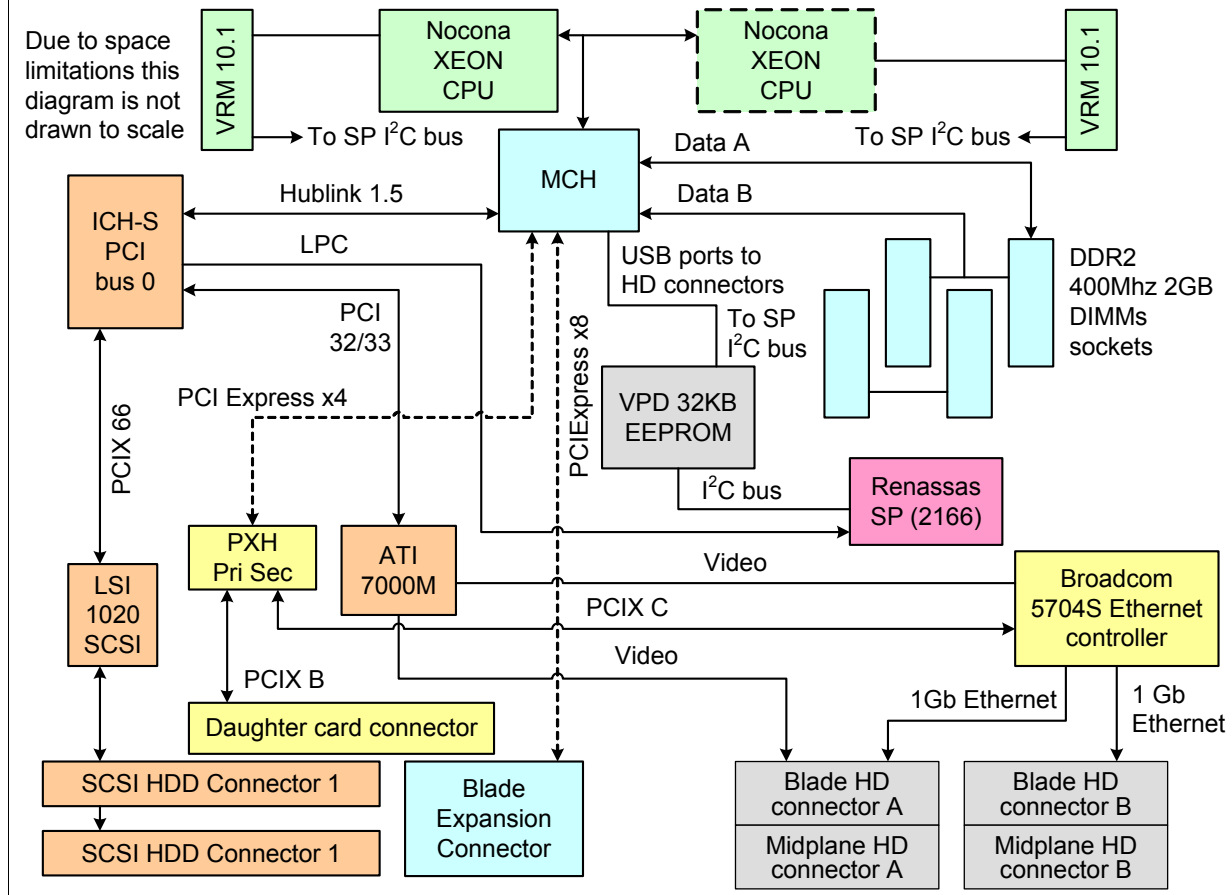


Figure 2-6 HS20 architecture

The Intel Lindenhurst chipset consists of the following components:

- ▶ Memory and I/O controller (MCH) (North Bridge)
- ▶ PXH-D
- ▶ ICH-S (South Bridge)

The Lindenhurst MCH, Memory and I/O controller provides the interface between the processors, the memory, and the PCI Express busses that interface to the other Intel chips. The Lindenhurst ICH-S (South Bridge) provides the USB interfaces, the local Service Processor interface, the POST/BIOS flash EEPROM interface, and the PCI bus interface for the ATI Radeon Mobility Video controller and LSI 1020 SCSI Host Controller. The PXH interfaces the Broadcom BCM5704S ethernet controller on its secondary bus and the daughter card on its secondary bus. I/O functions on the 8843 include Video, I²C, USB, SCSI, Gigabit Ethernet, and USB (floppy, CD-ROM (DVD), mouse, and keyboard).

The LPC bus is used to connect to the POST/BIOS EEPROM on the 8843. The size of the EEPROM is 4 MB x 8, and it contains primary BIOS, backup BIOS, and blade diagnostics.

PCI Express features include:

- ▶ PCI software compatibility
- ▶ Chip-to-chip, board-to-board implementations
- ▶ Support for end-to-end data integrity
- ▶ Advanced error reporting and handling for fault isolation and system recovery
- ▶ Low-overhead, low-latency data transfers and maximized interconnect efficiency
- ▶ High-bandwidth, low pin-count implementations for optimized performance

2.4 Stand-alone configuration tools

IBM BladeCenter hardware can be configured using standard software, such as a Web browser and a Telnet client, which are available on all the mainstream operating system platforms. This is possible by exploiting Web and American National Standards Institute (ANSI) interfaces that are embedded in both the management and the Ethernet Switch Modules.

A very comprehensive tool is accessible through the Web interface. This tool contains various configuration submenus, and one of them (I/O Module Tasks) lets you set up the Ethernet Switch Module. Basic settings (such as the Ethernet Switch Module IP address and the enablement of the external ports) are configured by exploiting the I2C bus. An advanced menu allows for the fine tuning of the module, by either opening another window of the Web browser or running a Java™ applet that allows for connectivity to an ANSI interface. (This requires that you have Java 2 V1.4 installed on the management system.) To achieve this, the 10/100 Mb internal link that connects the Management Module and the Ethernet Switch Modules through the BladeCenter backplane are exploited (notice that the internal network interface of the Management Module has a default static IP address of 192.168.70.126).

These more complete tools can also be accessed by pointing your Web browser, Telnet, or SSH client to the IP of the Ethernet Switch Module itself. (The default for a module that is plugged into Rear Bay 1 is 192.168.70.127. However, you can configure Dynamic Host Configuration Protocol (DHCP) based addressing.) Notice that this latter capability requires the management system to connect through the external ports (on the production LAN) of the Ethernet Switch Module and, therefore, might potentially raise concerns about security. That is why you have the capability to disable configuration control through the external ports in the I/O Module Tasks of the Management Module interface.

Figure 2-7 on page 13 illustrates the available stand-alone configuration tools.

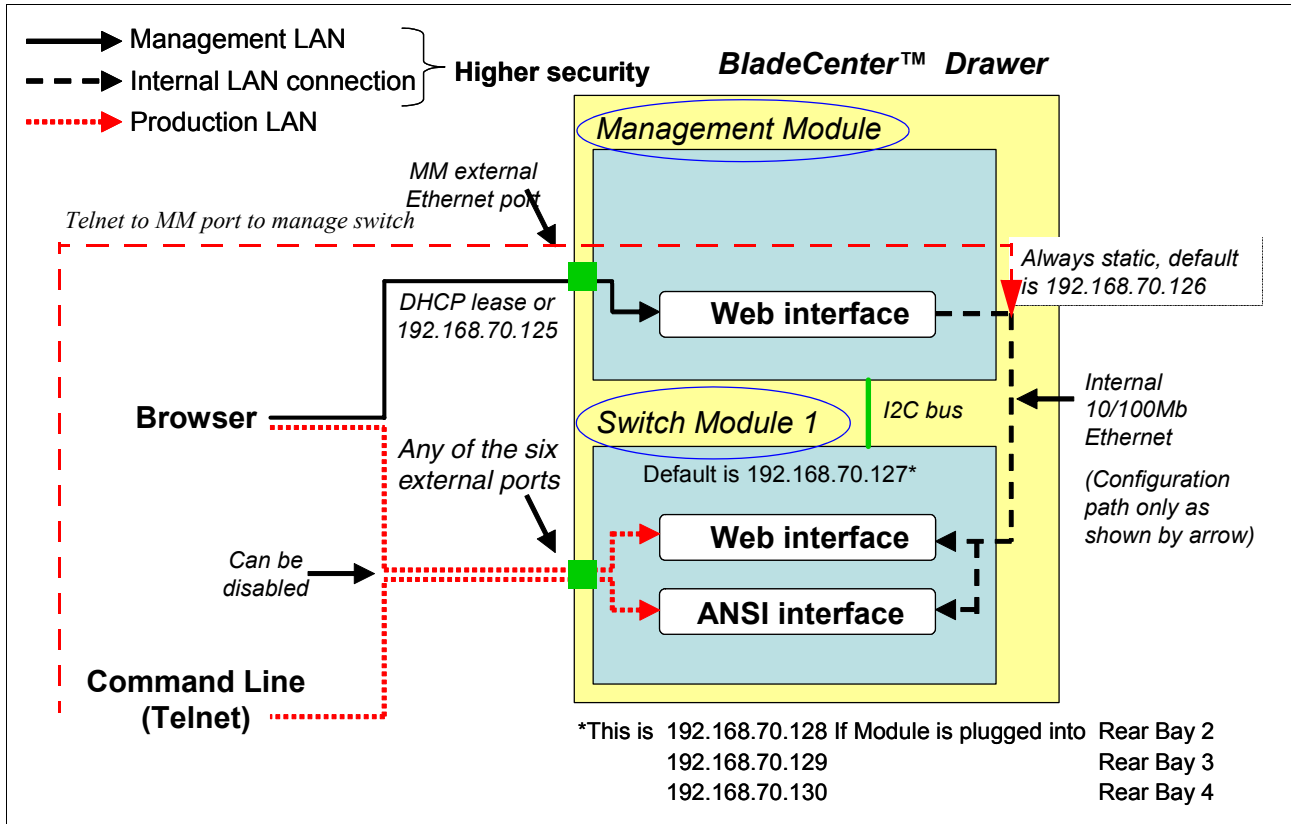


Figure 2-7 Stand-alone configuration tools



Introduction to Nortel Networks Layer 2-7 Gigabit Ethernet Switch Module

This chapter discusses the Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter set of features and services. The L2-7 GbESM is a fully functional content and load balancing switch with capabilities equivalent to products offered as free-standing appliances.

3.1 Value proposition

This section discusses the value of using the Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter for your IBM BladeCenter.

Product strength

The product delivers strengths such as:

- ▶ Provides full interoperability with Nortel, Cisco, and other standards based networking products.
- ▶ Integrates Nortel networking capabilities to reduce data center complexity and increases networking manageability and availability.
- ▶ Leverages the leadership capabilities BladeCenter Alliance Partners to provide the most technological choices.

Leadership features and functions

The leadership features and functions include:

- ▶ IBM BladeCenter delivers with the Nortel GbESM, full Layer 2 switching and Layer 3 switching (routing) functionality, as well as Layer 4 filtering and related services.
- ▶ Layer 4-7 functionality includes load balancing, Network Address Translation, SYN attack mitigation, and Layer 7 filtering (content filtering).
- ▶ The switch module runs Alteon Operating System (AOS) and appears as any other product from Nortel's Alteon product line to the data center's network management tools.

Competitive advantage

The product delivers a competitive advantage by delivering:

- ▶ Full integration of Ethernet switching, reducing infrastructure complexity
- ▶ Four external 10/100/100 Mbps copper ports
- ▶ Wire-speed performance switching at Layers 2 and 3 and Layer 4 load balancing
- ▶ Load balancing functionality similar to that offered by load balancing appliances which are offered by Nortel and several competing firms.
- ▶ Price leadership: the cost of the Nortel L2-7 GbESM is typically less than half that of a similar offering implemented as a stand alone appliance.
- ▶ Differentiation from other blade products because there is currently no equivalent switch from any of the other blade server vendors.

3.2 Architecture

Figure 3-1 shows the block diagram of the Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter.

The Nortel GbESM has two Gigabit Ethernet Aggregator (GEAs) for switching. It has 1 MB on chip cache for packet buffers and supports 20 Gigabit Ethernet ports (14 internal ports and four external ports).

The ports use four, external 1000BASE-T RJ-45 connectors.

SP0 and SP1 are Switch Processors, which perform some of the lower-level switching functions.

MP is the Management Processor, which runs the command line, SNMP, and certain high level switch functions

FP is the Frame Processor, which is a custom ASIC for performance acceleration.

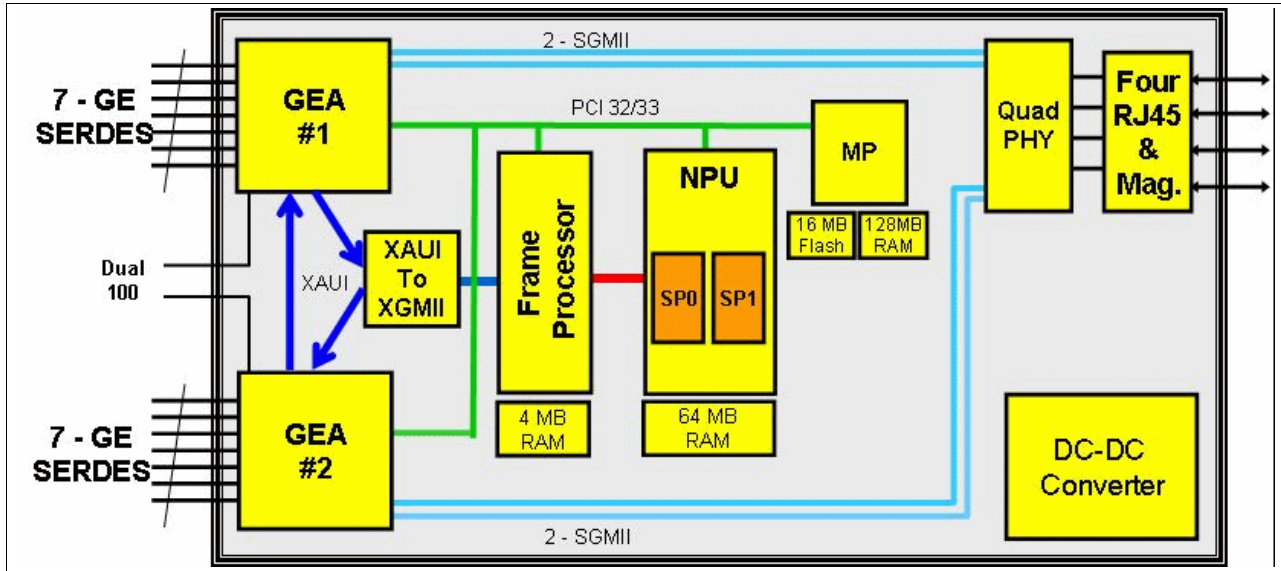


Figure 3-1 L2-7 GbE Switch Module architecture

3.3 GbESM: Capabilities, features, and functions

The IBM BladeCenter Layer 2-7 GbE Switch Module is a four-port gigabit switch module that can be installed in the IBM BladeCenter Type 8677. The IBM BladeCenter supports up to four GbESMs installed in the chassis at one time. An Ethernet daughter card option must be deployed on each server blade if the third and fourth GbESM is to be used in the chassis.

The GbESMs are hot-swappable subsystems that provide Ethernet switching capabilities within a BladeCenter chassis. The primary purpose of the switch module is to provide Ethernet connectivity among the processor blades, management modules, and the external network infrastructure.

The BladeCenter chassis requires a minimum of one switch installed in one of the slots in the rear of the chassis. The blades each ship with two 1-gigabit full-duplex links that correspond to slots 1 and 2 in the rear of the chassis. While the IBM @server® BladeCenter does function with a single switch installed, two independent switch modules installed in module slots 1 and 2 of the chassis are required for redundancy. With this configuration, each of the server blades in the BladeCenter chassis is then able to use either of its built-in network interfaces.

The GbESM has 18 ports that can be configured by the user. Ports 1 through 14 (INT1 through INT14) on the switch module are internal gigabit ports that correspond to server blades 1 through 14. Ports 17 through 20 are external 10/100/1000 Mbps copper ethernet ports for connection to the external network infrastructure. These ports are identified as EXT1 through EXT4 in the switch configuration menus and are labeled 1 through 4 (from top to bottom) on the switch module.

There are also two ports for dedicated use by the management modules. These ports, 15 and 16, are on a dedicated management VLAN (4095) and have a dedicated management

interface (Interface 128) configured. These ports appear in the Web or telnet management interfaces to the GbESM. However, the ports, VLAN, and interface cannot be configured by a user. This prevents changes to the ports that could cause access to the management modules to fail. If this happened, the switch could no longer communicate with the management modules and the management modules would no longer be able to manage the switch.

When the switch is first installed in the IBM @server BladeCenter, by default, the switch can *only* be governed through one of the Management Modules, or through the use of the external serial port and the associated special cable. The external interfaces are disabled for security reasons.

Essential information such as the machine type and serial number are located on the identification label on the side of the GbESM. You will need this information when you register the Layer 2-7 GbE Switch Module with IBM. See Figure 3-2 for the location of the identification label.

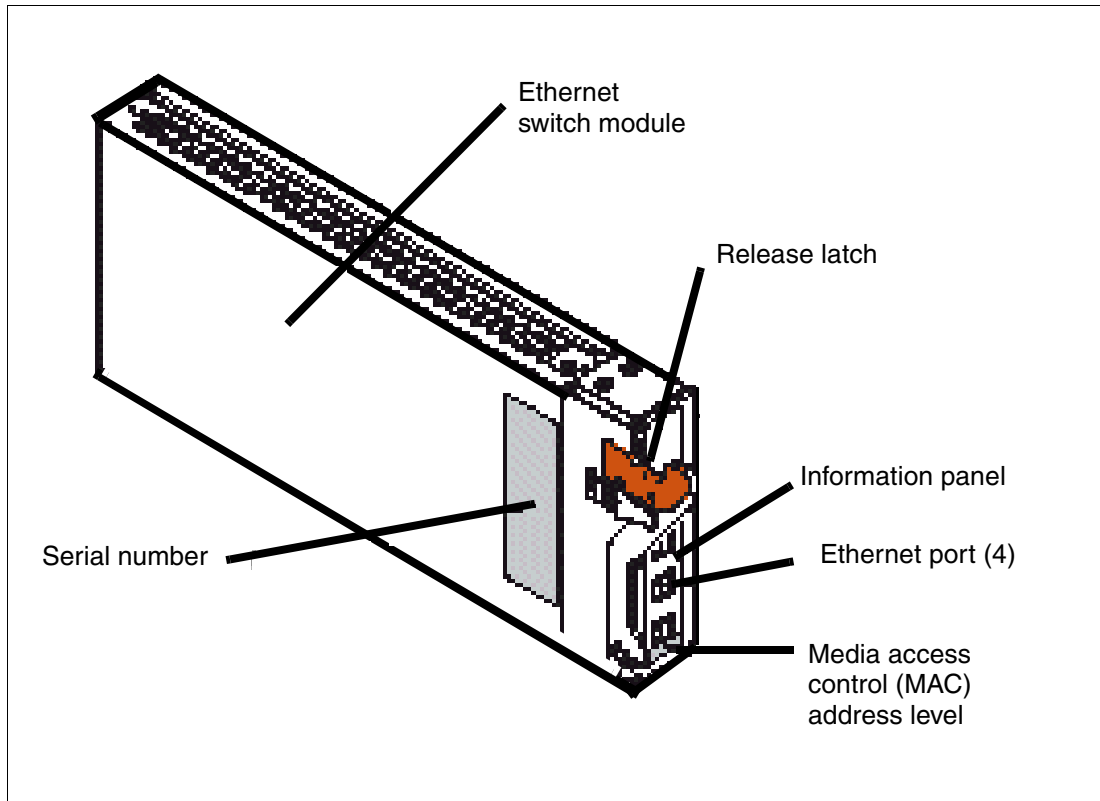


Figure 3-2 IBM BladeCenter Layer 2-7 GbE Switch Module

Note: The MAC address is also located on a separate label on the information panel under the external Ethernet port connectors.

The GbESM can be managed with Telnet or a Web interface. Both the Web and Telnet interfaces can be started by accessing the switch directly or starting a session from the management module's Web interface. Optionally, one can configure the switch to require the use of SSH for security, and disable the Telnet and Web interfaces.

Note: To manage the switch directly, without using the management module, you must meet these two conditions:

- ▶ Enable the external ports on the switch and then enable them for management.
- ▶ Because the management interface is on VLAN 4095 and cannot be moved, you must create an additional IP interface with an IP address valid on the production network.

The management interface on each switch by default is assigned a TCP/IP address that corresponds to the module slot in which the switch is installed. Table 3-1 lists the module slots and corresponding IP addresses. Figure 3-3 shows the locations for the slots in the BladeCenter chassis. The default addresses can be changed to addresses on your management network. However, the management module IP addresses and the GbESM management interface addresses must all be on the same subnet. If the management network is a different network than the production network, a router is required to access the management network. As we stated in the previous note, if you want to manage the switch directly from the production network, an interface must be created with an IP address valid on the production network.

Table 3-1 Default IP address listing

Bay Number	TCP/IP Address
1	192.168.70.127
2	192.168.70.128
3	192.168.70.129
4	192.168.70.130

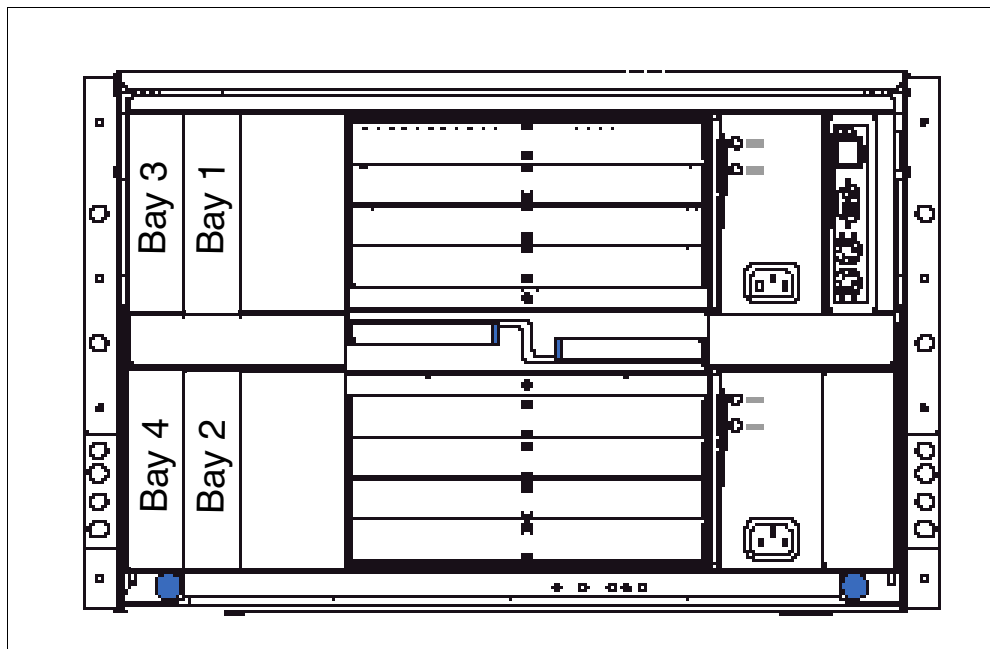


Figure 3-3 Bay locations rear of the IBM BladeCenter

Support for a GbESM installed in bays 3 and 4 requires an Ethernet daughter card option to be installed in any blade that uses those switches. The GbESM also contains an information panel with status LEDs on the rear of the switch. The status LEDs include an OK light, error

LED and link and activity lights for each of the external ports on the switch. Figure 3-4 shows the locations of the LEDs on the information panel.

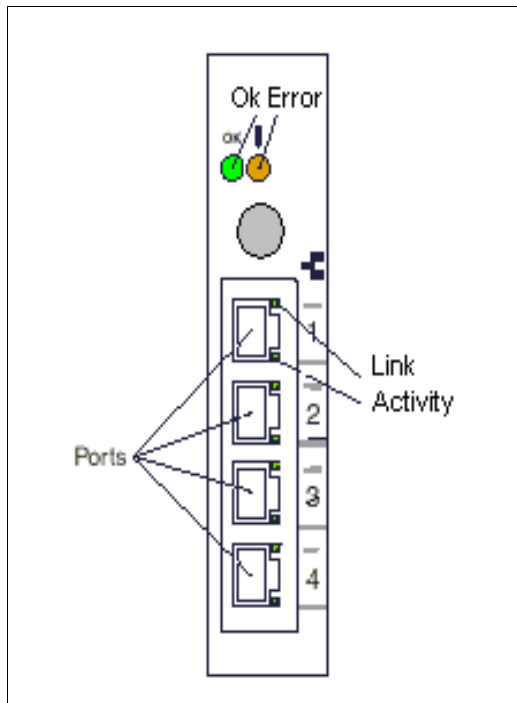


Figure 3-4 Information panel

The link and activity lights are described here starting at the top:

- ▶ OK (power-on)

This green LED is located above the four external 10/100/1000 Mbps ports on the information panel. When this LED is on, it indicates that the switch module has passed the power-on self-test (POST) and is operational.

- ▶ ! (Ethernet switch error)

This amber LED is located next to the OK LED on the information panel. The ! indicates that the switch module has a fault. If the switch module fails to POST or power on, this fault LED is lit.

- ▶ Ethernet link

This green link status LED is located at the top of each external 10/100/1000 Mbps port. When this LED is lit on a port, it indicates that there is a connection to a device on that port.

- ▶ Ethernet activity

This green activity LED is located at the bottom of each external 10/100/1000 Mbps port. When this LED blinks on a port, it indicates that data is being transmitted and received on the port.

Depending on the application, the external Ethernet interfaces can be configured to meet a variety of requirements for bandwidth or function. The IBM BladeCenter Layer 2-7 GbE Switch Module has been preconfigured with default parameter settings that can be used with most typical installations. However, all Layer 2-7 GbE Switch Modules need a few basic parameter settings initially, such as a TCP/IP address for management, security access and control parameters, and basic setup of the external ports for link aggregation.

This high performance switch is ideally suited for networking environments that require superior microprocessor performance, efficient memory management, flexibility, and reliable data storage. Performance, reliability, and expansion capabilities were key considerations in the design of the GbESM. These design features make it possible for you to customize the system hardware to meet your needs today, while providing flexible expansion capabilities for the future. If you have access to the World Wide Web, you can obtain up-to-date information about the GbESM and other IBM server products on the following Web site:

<http://www.ibm.com/eserver/xseries/>

User guides, drivers, and firmware updates can all be found at this site.

3.4 Switch management and operating concepts

This section provides a brief overview of several of the features of the GbESM and how the switch is managed. This section also covers some of the concepts of networking. The features and concepts, as well as the management of the switch, are covered in greater detail later in this Redpaper.

3.4.1 Switch management and control

The switch supports two management user interfaces:

- ▶ A Web server which supports the Browser Based Interface (BBI)
- ▶ A Telnet interface that can be invoked through the management module, or through a Telnet client program

Telnet supports the Command Line Interface (CLI), which is also supported by the serial port and by SSH if desired.

By default, the external ports on a new switch are disabled. Initial access to the switch must be through the management module to enable the external ports and configure an interface so the switch can be accessed from the network. Once the network settings are configured the switch can be accessed either through the management module or directly from the network.

Note that in order to use the Telnet option in the MM browser interface, the station where the browser is installed must have a Java Runtime Environment (JRE) installed.

After a TCP/IP address has been assigned to the GbESM, you can perform many different management and control tasks. These tasks fall in the following categories:

- ▶ Configuration of switch parameters
 - Switch TCP/IP address
 - Default gateway
 - General switch information: switch location, contact, system name
- ▶ Remote management setup
- ▶ Network monitoring
 - SNMP and traps
 - View port statistics
 - Monitor data traffic
- ▶ Switch maintenance

More information about these tasks and specific instructions on configuring the switch are given later in this Redpaper.

3.5 Ports and performance features

This section lists the specifications for the ports on the switch, as well as the performance and operational features of the GbESM.

- ▶ Ports
 - Four external copper ports for making 10/100/1000 Mbps connections to backbone, end stations, and servers. These ports are capable of autonegotiation for speed and duplex.
 - Fourteen internal full-duplex gigabit ports, one connected to each slot in the BladeCenter chassis.
 - Two internal full-duplex 10/100 Mbps ports for connection to the management modules. One port connects to each management module.
 - The four external ports can be configured for autosensing and can utilize either straight-through or crossover cables for switch-to-switch connections. The switch detects the type of cable used and sets the port accordingly. However, a cross-over cable is recommended because this cable will work with all of the modes.
- ▶ Performance and operational features of the GbESM
 - Transmission method: Store-and-forward.
 - Random-access memory (RAM) buffer: 8 MB.
 - Media access control (MAC) address learning: Automatic update, supports 28K MAC addresses.
 - Priority queues: Four priority queues per port.
 - Forwarding table age time: maximum age is 17 to 2100 seconds, default is 300 seconds.
 - 802.1D Spanning Tree support. Can be disabled on the entire switch or on a per-port basis.
 - 802.1Q Tagged virtual local area network (VLAN) support.
 - Support for 256 VLANs in total, including 128 static VLANs.
 - Link aggregation on four external ports for up to two static trunk groups or two link aggregation control protocol (LACP) 802.3ad link aggregation groups. This will be extended to a greater number of trunk groups and will support trunking of internal ports in the next software release.

3.6 Switch and network management

The GbESM supports the following network management protocols and standards. Some of these are technologies that can be used to manage and monitor the switch. Others, such as Spanning Tree, are technologies the switch uses to manage the network. The network technologies listed are covered in more detail later in this paper.

- ▶ Switch monitoring and management
 - Simple network management protocol (SNMP) version 1
 - Fully configurable either in-band or out-of-band control through SNMP-based software
 - Flash memory for software upgrades

This must be done through trivial file transfer protocol (TFTP) which requires access to an external TFTP server.

- Support for password-protected Web-based management and a telnet remote console
External security services (RADIUS, TACACS+) can be used for stronger protection.
- Built-in SNMP management:
 - Bridge management information base (MIB) (RFC 1493)
 - MIB-II (RFC 1213)
- ▶ Network management
 - 802.1P/Q MIB (RFC 2674)
 - Interface MIB (RFC 2233)
 - Mini-RMON MIB (RFC 1757) - four groups
The remote monitoring (RMON) specification defines the counters for the receive functions only. However, the switch provides counters for both receive and transmit functions.
 - Spanning Tree Protocol (STP) for creation of alternative backup paths and prevention of network loops.
 - TFTP support
 - Bootstrap protocol (BOOTP) support
 - Dynamic host configuration protocol (DHCP) client support

3.7 Network cables

The following cables and cable lengths are supported by the GbESM:

- ▶ 10BASE-T
 - UTP Category 3, 4, 5 (100 meters maximum)
 - 100-ohm STP (100 meters maximum)
- ▶ 100BASE-TX:
 - UTP Category 5 (100 meters maximum)
 - EIA/TIA-568 100-ohm STP (100 meters maximum)
- ▶ 1000BASE-T:
 - UTP Category 5e (100 meters maximum)
 - EIA/TIA-568B 100-ohm STP (100 meters maximum)

3.8 Supported network standards

The following standards are supported by the GbESM. Some of these standards are explained in greater detail later in this document.

- ▶ IEEE 802.3 10BASE-T Ethernet
- ▶ IEEE 802.3u 100BASE-TX Fast Ethernet
- ▶ IEEE 802.3z Gigabit Ethernet
- ▶ IEEE 802.1Q Tagged VLAN
- ▶ IEEE 802.3ab 1000BASE-T
- ▶ IEEE 802.3x Full-duplex Flow Control
- ▶ ANSI/IEEE 802.3 NWay auto-negotiation
- ▶ 802.3-2002 LACP
- ▶ IETF standard VRRP, with extensions

3.9 Layer 4-7 switching functions

This section provides a high level enumeration of the available Layer 4-7 Switching functions. These functions are discussed in greater depth in Chapter 5, “Introduction to server load balancing” on page 47.

Virtual server-based load balancing

This is the traditional load-balancing method. The switch is configured to act as a virtual server and is given a Virtual IP address (VIP) for each service it will distribute. Each virtual server is assigned a list of the IP addresses (or range of addresses) of the real servers in the pool where its services reside. When the user stations request connections to a service, they will communicate with a virtual server on the switch. When the switch receives the request, it binds the session to the IP address of the best available real server and remaps the fields in each frame from virtual addresses to real addresses. HTTP, IP, FTP, RTSP, IDS, and static session WAP are examples of some of the services that use virtual servers for load balancing.

Filter-based load balancing

A filter allows you to control the types of traffic permitted through the switch. Filters are configured to allow, deny, or redirect traffic according to the IP address, protocol, or Layer 4 port criteria. In filter-based load balancing, a filter is used to redirect traffic to a real server group. If the group is configured with more than one real server entry, redirected traffic is load balanced among the available real servers in the group. Firewalls, WAP with RADIUS snooping, IDS, and WAN links use redirection filters to load balance traffic.

Content-based load balancing

Content-based load balancing uses Layer 7 application data (such as URL, cookies, and Host Headers) to make intelligent load balancing decisions. URL-based load balancing, browser-smart load balancing, and cookie-based preferential load balancing are a few examples of content-based load balancing.

Network Address Translation

Network Address Translation is used to obscure the real addresses of the blade servers from the public network and to reduce the number of public routable IP addresses required. NAT is a key part of load balancing but can also be used on its own.

3.9.1 Key benefits of Layer 4-7 Switching

This is a brief summary of the benefits of using L4-7 switching:

- ▶ Increased efficiency for server utilization and network bandwidth

The Nortel GbESM is aware of shared services provided by *Virtual Service Pools* and can balance user session traffic among the available servers. Important session traffic gets through more easily, reducing user competition for connections on over-utilized servers. For even greater control, traffic is distributed according to a variety of user-selectable rules.

- ▶ Increased reliability of services to users

If any server in a server pool fails, the remaining servers continue to provide access to vital applications and data. The failed server can be brought back up without interrupting access to services.

- ▶ Increased scalability of services

As users are added and the server pool’s capabilities are saturated, new servers can be added to the pool without disrupting operations.

3.10 Comparison to L2/3 switch module

In general, the Nortel Networks L2/3 GbESM has a subset of the functions of the L2-7 GbESM. There are certain functions which the L2/3 GbESM has which are not yet implemented in the current L2-7 GbESM software. The details of this will be discussed in depth in Chapter 4, "Integrating the L2-7 Switch Module into a network" on page 27. In summary, the L2/3 GbESM and the L2-7 GbESM both include the following except where noted:

- ▶ Six external ports, either copper (10/100/1000Mbps) or Fiber (1000Mbps) on the L2/3 switch module; four copper (10/100/1000Mbps) on the L2-7 switch module
- ▶ Layer 2 forwarding (Ethernet switching)
- ▶ Layer 3 forwarding (IP routing)
- ▶ High Availability functionality at layers 2 or 3, or a combination of both
- ▶ Layer 2-4 filters which can block or allow traffic based on:
 - MAC (Ethernet) addressing - Layer 2
 - IP addresses - Layer 3
 - TCP/UDP port numbers - Layer 4
- ▶ Text command line management and configuration with Telnet or SSH
- ▶ Browser- based management and configuration

The L2-7 GbESM includes the following capabilities which are not included in the L2/3 GbESM. These functions are described in depth in Chapter 5, "Introduction to server load balancing" on page 47.

- ▶ Server Load Balancing
 - based on L4 criteria such as TCP/UDP port
 - based on L7 content inspection
- ▶ High Availability functions at Layer 4 which work in concert with Server Load Balancing
- ▶ Network Address Translation
- ▶ Global Server Load Balancing (1H2006 software release)
- ▶ Filters which can allow or drop traffic based on inspection of the payload as well as the various protocol headers (L7 content inspection)
- ▶ Denial of Service (DoS) attack mitigation.



Integrating the L2-7 Switch Module into a network

In this chapter, we discuss tools, techniques, and applications that help with the management and deployment of the Nortel GbESM in an IBM *@server* BladeCenter. We also discuss the management paths and rules for connecting to and accessing the Nortel GbESM.

4.1 Nortel Networks L2/7 GbESM management connectivity

In this section, we look at the basic management connectivity and management pathways to the Nortel GbESM, as shown in Figure 4-1.

Important: Properly managing the Nortel GbESM in the IBM BladeCenter requires proper management of the Management Module within the BladeCenter chassis. It is virtually impossible to deploy the Nortel GbESM successfully if you do not understand and properly configure certain settings in the Management Module, as well as the necessary Nortel GbESM configurations.

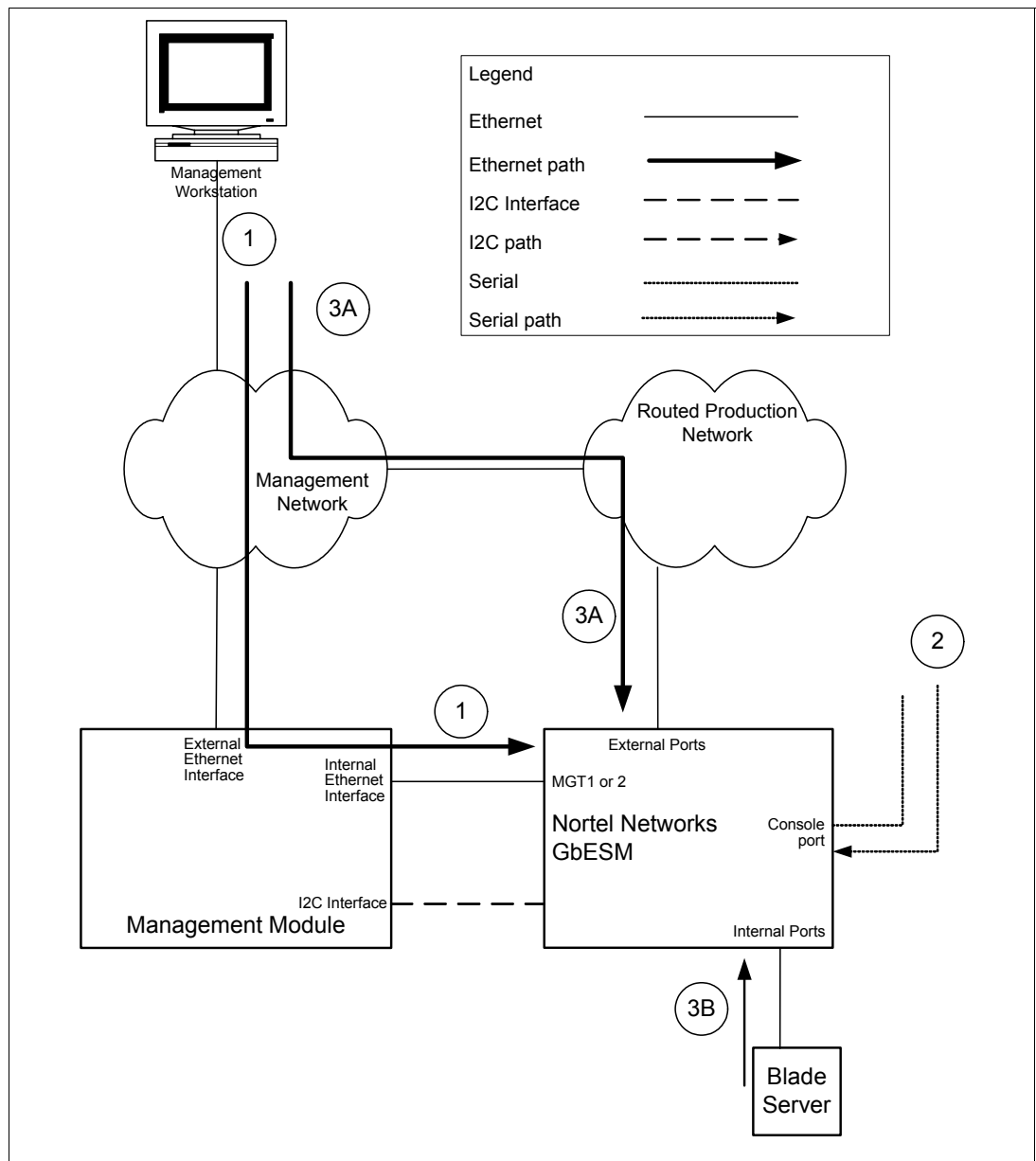


Figure 4-1 Management paths to the Nortel Networks L2/7 GbESM

4.1.1 Out-of-band management

It is common to provide a physically separate management interface for all of the devices and to carry only management traffic. This is referred to as *out-of-band management* and is sometimes a separate Ethernet connection (path 1) or a wholly different physical connection, such as the console port (path 2).

Management Module (Path 1)

The IBM BladeCenter comes with at least one Management Module. The Management Module supports an external Ethernet interface, which is used to manage the Blade servers, Ethernet switches, and the Management Module itself. Within the IBM @server BladeCenter, management traffic flows through a different bus, the I2C bus, as shown in the Figure 4-1 on page 28.

On the Nortel GbESM, the Ethernet management (MGT1 and MGT2) ports which connect the switch to the Management Module are placed in VLAN 4095. It is not possible to change this. It is also not possible to reach VLAN 4095 from any of the other internal or external ports on the switch. This is a deliberate design constraint. It is intended to enforce isolation of the Management Module network (VLAN) from any other networks (VLANs) that are configured on the switch. This implies that the Blade servers should not be on the same VLAN nor the same IP subnet as the Management Module. Placing the servers on the same subnet as the Management Module can have unexpected and undesirable results.

The first step in configuring the Nortel GbESM is to assign the IP address of the MGT ports through the Web interface of the Management Module (Figure 4-2).

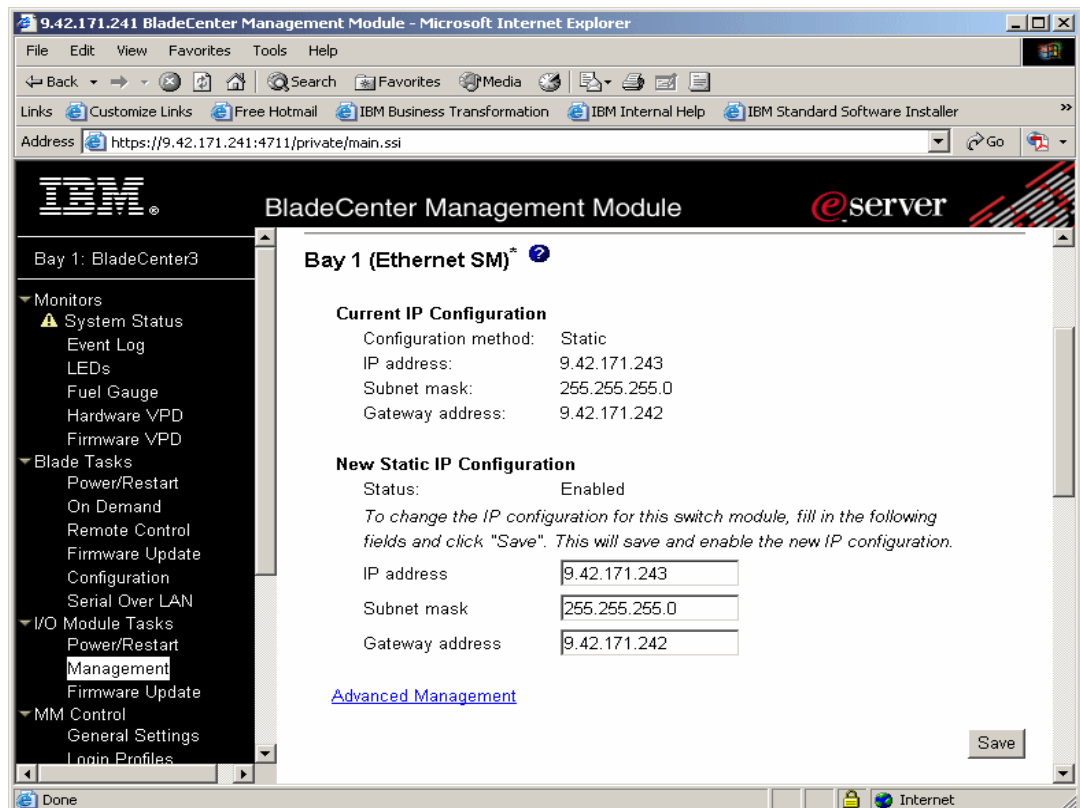


Figure 4-2 Configuring the Nortel MGT port IP address using the Management Module Web interface

Further configuration of the Nortel Networks L2/7 GbESM module is performed by using Telnet (for the Command Line Interface) or a Web browser (for the Browser Based Interface) and accessing the address of the MGT1 or MGT2 ports.

Note: It is recommended (and easier) to use a server or mobile computer that is external to the IBM @server BladeCenter chassis to perform initial configuration of the Nortel Networks L2/7 GbESM module. The server or mobile computer should be able to open the Web interface of the Management Module. It then can reach the switch when the switch has an appropriate IP address configured. This address must be within the same subnet as both the internal and external IP addresses of the Management Module.

Serial port (Path 2)

The Serial port is used for out of band management of the switch. It is useful to allow access to the CLI when all other paths are not working. It is possible to connect the serial port to a terminal server if desired; this allows out-of-band access to be easily provided to multiple devices.

The console cable that is required to use this port is included with the switch when it is shipped. The cable (L2-7 switch order number, 02R9361) has a RS232 Apple/Centronics printer plug on one end and a DB-9 plug on the other end. The DB-9 is intended to be attached to a standard serial port such as on a mobile computer or modem. Standard terminal emulation software should be used with these settings: 9600 baud; no parity; 8 data bits; 1 stop bit (9600,N,8,1).

4.1.2 In-band management

The second mode of operation that is commonly used is *in-band management*. In this case, the management traffic passes through the data traffic path (the Nortel Networks L2/7 GbESM EXTERNAL and INTERNAL ports).

External Ethernet ports (Path 3A)

The external ports can be used to provide management access to the switch from outside the IBM @server BladeCenter chassis. In order to use this path, the **External management over all ports** item in the Management Module configuration must be **enabled** (Figure 4-3 on page 31).

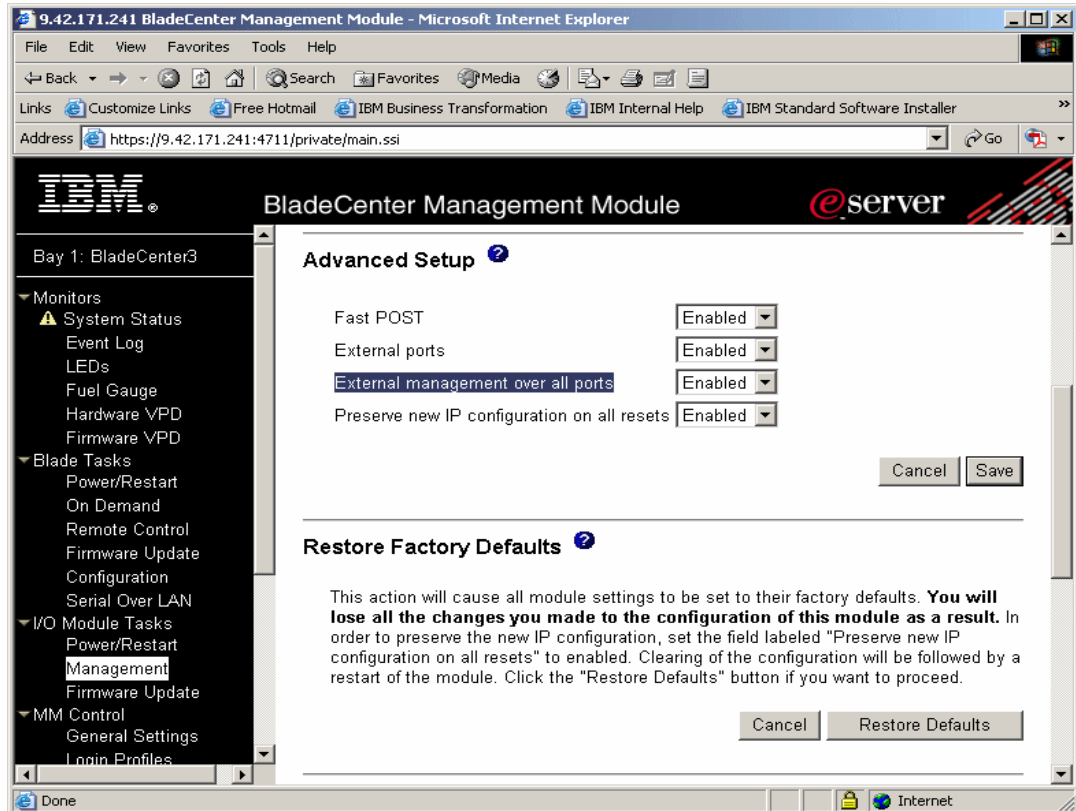


Figure 4-3 Enabling management over all ports using the Management Module Web interface

Internal Ethernet ports (Path 3B)

The internal ports can be used to provide management access to the switch from the server blades in the same chassis.

In-band management considerations

In order to use in-band management paths, you must configure at least one additional IP address on the Nortel Networks L2/7 GbESM beyond the address that is provided through the Management Module and attached to VLAN 4095. This additional IP address should be attached to one of the active VLANs configured on the switch and which includes one or more of the EXTERNAL ports.

Using the `mnet` command on the Ethernet switch, you can limit management access to the switch to management stations within a defined range of IP addresses.

Note: The `mnet` command limits all IP-based management access regardless of which path is involved. Thus, use it with care. It is possible to lock out access with the Management Module (MGT ports) using this command.

4.2 Nortel Networks L2/7 GbESM user interface

This section discusses the switch module management interface and what each task represents. To configure and manage the switch module, you can use the following interfaces:

- ▶ IBM BladeCenter Management Module and I2C
Management functions that are necessary for initial setup are provided through the Management Module Web interface. I2C is the communication that is used between the Management Module and Ethernet switch.
- ▶ Command-line interface (CLI)
You can configure and monitor the switch from the CLI, which is accessible through Telnet or SSH from a remote management station. You can also access the CLI through terminal emulation software on a management station directly connected to the switch module console port.
- ▶ Browser Based Interface (BBI)
You can use the BBI to manage and monitor the switch using a standard Web browser with HTTP. It provides a graphical means of viewing and configuring the switch's characteristics.

4.2.1 IBM BladeCenter Management Module and I2C

The Management Module Web interface is the only mechanism for performing certain management functions, including:

- ▶ Configuring the management IP address of the switch
- ▶ Enabling or disabling the external ports and management with these ports
- ▶ Configuring Power On Self Test (POST) options
- ▶ Remotely turning power to the switch on or off

All of these functions use the I2C interface when they need to communicate with the switch module. The use of the Management Module to configure Ethernet switches is documented in detail in the *Nortel Networks Layer 2/7 GbE Switch Module Installation Guide*.

4.2.2 Command-line interface

The CLI is more flexible for configuring the switch than the BBI. It is scriptable, requires less overhead to run. Because it is a Telnet session, it can be run from any operating system, whether or not it is graphical.

Main Menu commands

Figure 4-4 on page 33 shows the Main Menu window. Each of the following commands brings you to a first level submenu:

- ▶ The `stats` menu gives statistics about the switch.
- ▶ The `cfg` menu contains all of the configuration options for the switch.
- ▶ The `oper` menu contains all of the operator commands. Some of these commands can change the state of the switch, but these changes only apply until the next reboot. They are not permanent.
- ▶ The `boot` menu contains the commands to control the booting of the switch, from which image to boot, which config to boot, and the `gtimg` and `ptimg` commands for getting and putting firmware files to the switch.

- ▶ The maint menu contains all of the commands for maintenance of the switch. The commands to manipulate the ARP cache and forwarding database are here, as well as the commands to obtain dumps of the current state of the switch for technical support.

```

Command Prompt - telnet 9.42.171.243
>> Main# /
-----
[Main Menu]
info      - Information Menu
stats     - Statistics Menu
cfg       - Configuration Menu
oper      - Operations Command Menu
boot      - Boot Options Menu
maint     - Maintenance Menu
diff      - Show pending config changes [global command]
apply     - Apply pending config changes [global command]
save      - Save updated config to FLASH [global command]
revert    - Revert pending or applied changes [global command]
exit      - Exit [global command, always available]

>> Main# _

```

Figure 4-4 CLI Main Menu

Global commands

The remainder of the options on the Main Menu — diff, apply, save, revert, and exit — are all global commands that work anywhere on the switch. Figure 4-4 shows what each of the commands does. The help command is also global and lists all the global commands, as shown in Figure 4-5.

```

Command Prompt - telnet 9.42.171.243
>> Main# help
For help on a specific command, type  help <command>

Global Commands: [can be issued from any menu]
help      up          print          pwd
lines     verbose       exit          quit
diff      apply         save          revert
ping      traceroute    telnet        history
pushd     popd          who

The following are used to navigate the menu structure:
. Print current menu
.. Move up one menu level
/ Top menu if first, or command separator
? Execute command from history

>> Main#

```

Figure 4-5 CLI global command list

Navigation commands

There are several commands that are useful in moving from one part of the menu tree to another. The commands are similar to those used in a UNIX® shell:

cd	This command moves you to a given spot in the menu tree. Entering cd / always takes you back to the main menu.
pwd	This command displays the current menu path where you are in the menu tree.
up	This takes you back to the last menu that you touched.
.. or cd ..	Both of these commands take you up one level in the menu tree.
pushd and popd	These commands allow you to manage a stack of menus that you visit frequently.
history	This command displays the last several commands that you entered. You can reuse these commands by typing an exclamation point (!) followed by the number of the command as displayed.
quit or exit	Either of these commands terminates your session.

Configuration control commands

These commands control the effectiveness of changes to the switch configuration. The general rubric for configuring the switch is EASY:

- ▶ E for editing the configuration, typing in your changes
- ▶ A for the **apply** command, which makes the changes part of the running configuration
- ▶ S for the **save** command, which writes the changed configuration to flash memory
- ▶ Y for yes, which is the answer to the prompt to be sure that you really want to update flash

Additional configuration control commands include the following:

diff	This command displays the differences between the most recent edits and the running configuration.
diff flash	This command displays the differences between the running configuration and its flash copy.
revert	This command discards all changes which have not yet been applied.
revert apply	This command discards applied changes which have not yet been saved to flash.

Additional commands

These additional commands facilitate troubleshooting or are otherwise helpful:

ping	This command sends ping, Internet Control Message Protocol (ICMP) echo, requests to the specified IP address.
traceroute	This command traces the IP path to a specified IP address.
who	This command shows who is logged on to the switch and from which address.
telnet	This command opens a Telnet session to the designated IP address.
verbose	This command tailors the level of messages displayed on your session.
lines	This command controls the number of lines per screen for display purposes.

Upgrading the firmware

To upgrade the firmware on the Nortel Networks L2/7 GbESM, you must use Trivial File Transfer Protocol (TFTP). It is not possible to use the Management Module menu item for upgrading firmware at this time. However, this is a planned feature for a future software release, as is the use of full File Transfer Protocol (FTP) in addition to TFTP.

Important: Before updating the firmware, save any configuration changes to the Nortel Networks L2/7 GbESM.

1. From the Telnet session, enter `apply`, then press Enter.
2. Type `save` and press Enter.
3. Answer `y` to the prompt that asks to confirm saving to flash.
4. Answer `y` to the prompt that asks if you want to change the boot to the active config block if it appears.

Example 4-1 shows the process to load a new OS image file onto the switch. Prior to running the executables in the example, ensure the home directory of the TFTP server contains the firmware files.

Example 4-1 Command Prompt: Telnet 9.42.171.21 (Display of a firmware update using CLI)

```
>> Boot Options# /boot/gtimg
Enter name of switch software image to be replaced
["image1"|"image2"|"boot"]: image2
Enter hostname or IP address of TFTP server: 9.42.170.72
Enter name of file on TFTP server: GbESM-AOS-20.2.2.7-os.img

image2 currently contains Software Version 20.2.2.6
that was downloaded at 0:15:14 Thu Jan 1, 2070.

New download will replace image2 with the file "GbESM-AOS-20.2.2.7-os.img"
from TFTP server 9.42.170.72.

WARNING: This operation will overlay the currently booting image.
Confirm download operation [y/n]: y
Starting download...
File appears valid
Download in progress.....
.....
Image download complete (3389778 bytes)
Writing to flash...This takes about 90 seconds. Please wait
Write complete (3389778 bytes), now verifying FLASH...
Verification of new image2 in FLASH successful.
image2 now contains Software Version 20.2.2.7

Updating the Switch Image 2 Version (2002WM02007 )...
Updating the Switch Image 2 Name (AlteonOS Im2)...
Updating the Switch Image 2 Date (09/22/2005)...
>>
Jan 4 16:14:53 INFO mgmt: image2 downloaded from host 9.42.170.72, file 'GbE
SM-AOS-20.2.2.7-os.img', software version 20.2.2.7
Boot Options#
```

The firmware for the Nortel Networks L2/7 GbESM is contained in two files: one is a boot image file and the other is the OS image file. Use the following steps to upgrade the firmware on the Nortel Networks L2/7 GbESM using the Telnet session:

1. Type `/boot/gtimg`.

2. Enter where the new image file will be placed. We are upgrading the boot image file, so enter `boot`. That is the location for the boot image file.
3. Enter the IP address of the TFTP server.
4. Enter the fully qualified path name for the boot image file that is on the TFTP server.
5. The switch reports the current version of the boot kernel on the switch and asks if you wish to replace it with new file. If you wish to continue, enter `y`.
6. When the download is finished, return to Step 1 and repeat the process for the OS image file. In step 2, enter `image1` or `image2` as the location to store the new image file.
7. If the download location is the same as the location for the currently loaded OS image, the switch warns you that a failed download could result in an inoperative switch. If the download location is different from the location of the currently loaded OS image, the image file downloads. After the download is finished, the switch asks whether you want to use the old location or the new location. Example 4-1 on page 35 shows a successful download of the OS image to `image2`.
8. Type `/boot/reset` to reset the switch and reboot with the new firmware files.

Capturing the current configuration

There are a few ways to capture the current configuration in the CLI. The first is to use a TFTP server to push the configuration file from the switch to the server. However, in some text editors the resulting file is a single long line of text. We suggest using WordPad as a best practice. Although this method requires a TFTP server running in the network, it does work with any Telnet client. To capture the configuration by pushing a file to a TFTP server, follow these steps:

1. Enter `/cfg/ptcfg` at the command line.
2. Enter the IP address of the TFTP server.
3. Enter the filename to which you want to save the file.

A second way to capture the current configuration does not require a TFTP server. This method, however, requires a terminal emulator that can capture text. Example 4-2 uses a Microsoft Windows® Telnet session to capture the text. The commands on the switch are the same for any software, but the steps to set the software to capture the text might be different. If your terminal emulator does not support this, you have to use the TFTP method. By using a Windows Telnet session and issuing the `/cfg/dump` command, you can dump the full switch configuration.

Example 4-2 Example configuration file dump

```
>> Main# /cfg/dump
script start "Layer 2-7 GbE Ethernet Switch Module" 4
/**** DO NOT EDIT THIS LINE!
/* Configuration dump taken 14:03:36 Fri Jan 6, 2006
/* Version 20.2.2.6, Base MAC address 00:0e:62:38:19:00
/c/port INT1
    pvid 20
/c/port INT2
    pvid 10
/c/port INT3
    pvid 10
/c/port INT4
    pvid 99
/c/port EXT1
    tag ena
/c/port EXT2
    tag ena
```



```

/c/12/vlan 10
    ena
    name "VLAN_Green"
    def INT2 INT3 EXT1 EXT2
/c/12/vlan 20
    ena
    name "VLAN_Red"
    def INT1 INT2 EXT1 EXT2
/c/12/vlan 99
    ena
    name "MGMT"
    def INT4 EXT1 EXT2
/c/12/stg 1/clear
/c/12/stg 1/add 1 10 20 99
/c/12/lacp/port EXT1
    mode active
/c/12/lacp/port EXT2
    mode active
    adminkey 17
/c/13/if 99
    ena
    addr 10.99.0.243
    mask 255.255.255.0
    broad 10.99.0.255
    vlan 99
/c/13/gw 1
    ena
    addr 10.99.0.245
/c/13/gw 2
    ena
    addr 10.99.0.246
/
script end /**** DO NOT EDIT THIS LINE!

```

Configuring user accounts

This section describes the user accounts on the switch. The seven user accounts listed in Table 4-1 are the default accounts on the GbESM.

Table 4-1 Description of default user accounts

User account	Description/Tasks performed	Password
User	Can view switch statistics but cannot make changes.	user
SLB Operator	Can manage content servers and configure options on the SLB menus, but not filters.	slboper
Layer 4 Operator	Reserved for future use.	l4oper
Operator	The Operator manages all functions of the switch. In addition to SLB Operator functions, the Operator can reset ports or the entire switch.	oper
SLB Administrator	The SLB Administrator configures and manages content servers, and other Internet services and their loads.	slbadmin
Layer 4 Administrator	In addition to SLB Administrator functions, the Layer 4 Administrator can configure all parameters on the SLB menus, including filters and bandwidth management.	l4admin

User account	Description/Tasks performed	Password
Administrator	The super-user Administrator has complete access to all menus, information, and configuration commands on the switch.	admin

There is currently no mechanism on the GbESM for adding user IDs and passwords directly to the switch. This will be added in the next software release. Even when this feature is added, you can obtain greater security and manageability through the use of an external security server.

The GbESM supports the Remote Authentication Dial-in User Service(RADIUS) and TACSACS+ security servers. These servers can be used to authenticate and authorize remote administrators and can grant them varying levels of authority.

To use this approach, follow these steps:

1. Add to the security server the users you want added to the switch.
2. Configure the switch to use the security server.

Users only need to be defined at the server once and can then access any switch that is configured to use that server.

Security servers are not required to access the switch. Further, there is a provision for *backdoor* access, to ensure that administrators are not locked out of the switch if the security server is down or unreachable.

For more information about RADIUS, TACACS, and how to configure the switch to use them, refer to the current version of the *Alteon OS Application Guide* for the L2/7 Nortel GbESM.

When you access the switch using an external security server, the authentication methods differ slightly for the CLI and Web interfaces.

- ▶ In the case of the CLI, when a security server is in use, prompts for both USER ID and PASSWORD will be presented at sign-on.
- ▶ For the BBI, the browser's standard dialog box which contains ID and PASSWORD fields will be used.

Users have varying access levels on the switch. You will be presented with the appropriate menu for your authority level after you have successfully signed on (with the CLI). If you have **user** authority level, for example, you are presented with a much more limited menu than if you were to access the switch with **admin** authority. Figure 4-6 and Figure 4-7 on page 39 show the different main menus for **admin** and **user**. As you can see, **user** has a much more limited set of commands available than **admin**.

```

C:\WINNT\System32\cmd.exe - telnet 192.168.70.127
Serial Number:      YJ1RTK38J603
Manufacturing Date: 0334
Hardware Revision:  0
PLD Firmware Version: 3.6

Temperature Sensor 1 (Warning):  46.0 C (Warn at 75.0 C/Recover at 70.0 C)
Temperature Sensor 2 (Shutdown): 44.0 C (Warn at 90.0 C/Recover at 80.0 C)

-----
[Main Menu]
Jan 1 1:26:06 NOTICE mgmt: admin login from host 192.168.70.120      info
- Information Menu
  stats      - Statistics Menu
  cfg        - Configuration Menu
  oper       - Operations Command Menu
  boot       - Boot Options Menu
  maint      - Maintenance Menu
  diff       - Show pending config changes [global command]
  apply      - Apply pending config changes [global command]
  save       - Save updated config to FLASH [global command]
  revert     - Revert pending or applied changes [global command]
  exit       - Exit [global command, always available]

>> Main#

```

Figure 4-6 Main menu for admin

```

Telnet GSM - HyperTerminal
File Edit View Call Transfer Help
-----
Enter password:
-----
[Main Menu]
  info      - Information Menu
  stats     - Statistics Menu
  exit      - Exit [global command, always available]

>> Main>

```

Figure 4-7 Main menu for user

When you access the switch through the Web interface, you are prompted for a username and a password. For all the default users on the switch, the username and password are the same by default. If you wish to change any of the passwords, you must be logged in as admin. To change a password enter the command:

```
/cfg/sys/access/user
```

This will bring up the menu shown in Figure 4-8 on page 40. From this menu you can change the passwords of all the default users on the switch.

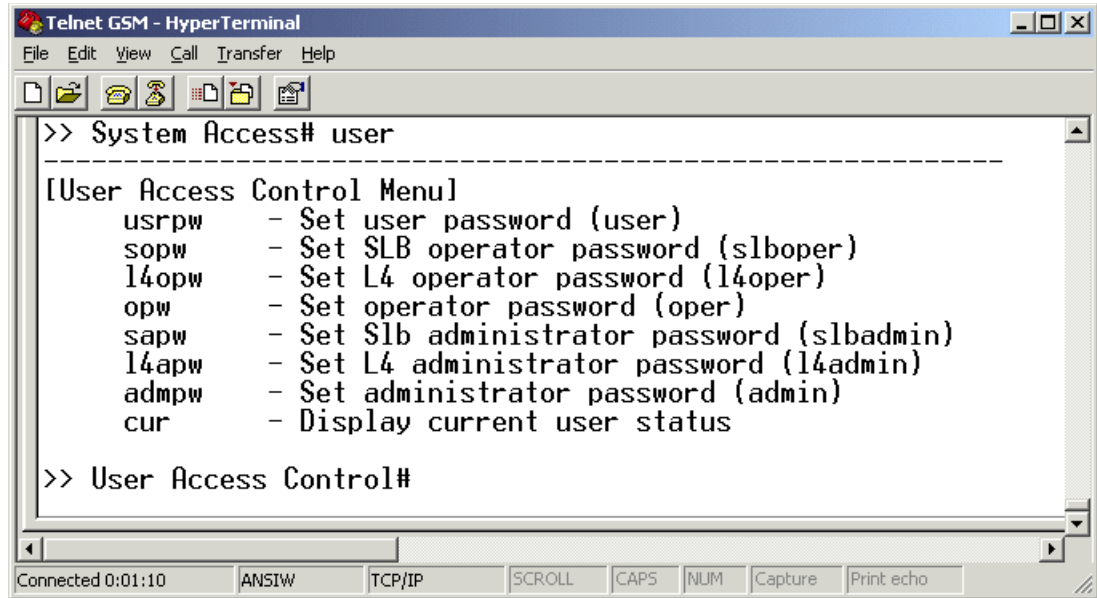


Figure 4-8 User password control

4.2.3 Browser Based Interface

We now take a brief look at the Browser-Based Interface (BBI) on the Nortel Networks L2/7 GbESM. Almost everything that can be done through the CLI can also be done in the BBI. In the remainder of this book, more emphasis is placed on configuring the switch using the CLI rather than using the BBI.

The Switch Information panel displays the MAC address of the switch as well as the firmware and hardware versions. Use the following steps to configure the system and contact information:

1. From the Nortel Networks L2/7 GbESM Web interface, click the folder icon next to Nortel Networks Layer 2/7 GbE Switch Module in the left-hand frame.
2. Click the folder icon next to System in the left-hand frame.
3. Click **CONFIGURE** at the top of the page.
4. Click the icon next to General in the drop-down list under System. On a window similar to Figure 4-9 on page 41, you see options, such as IP Address and Network Mask fields, that can be configured on this page. Other options on this page include date and time settings, syslog settings (if you have a syslog server), and SNMP settings.

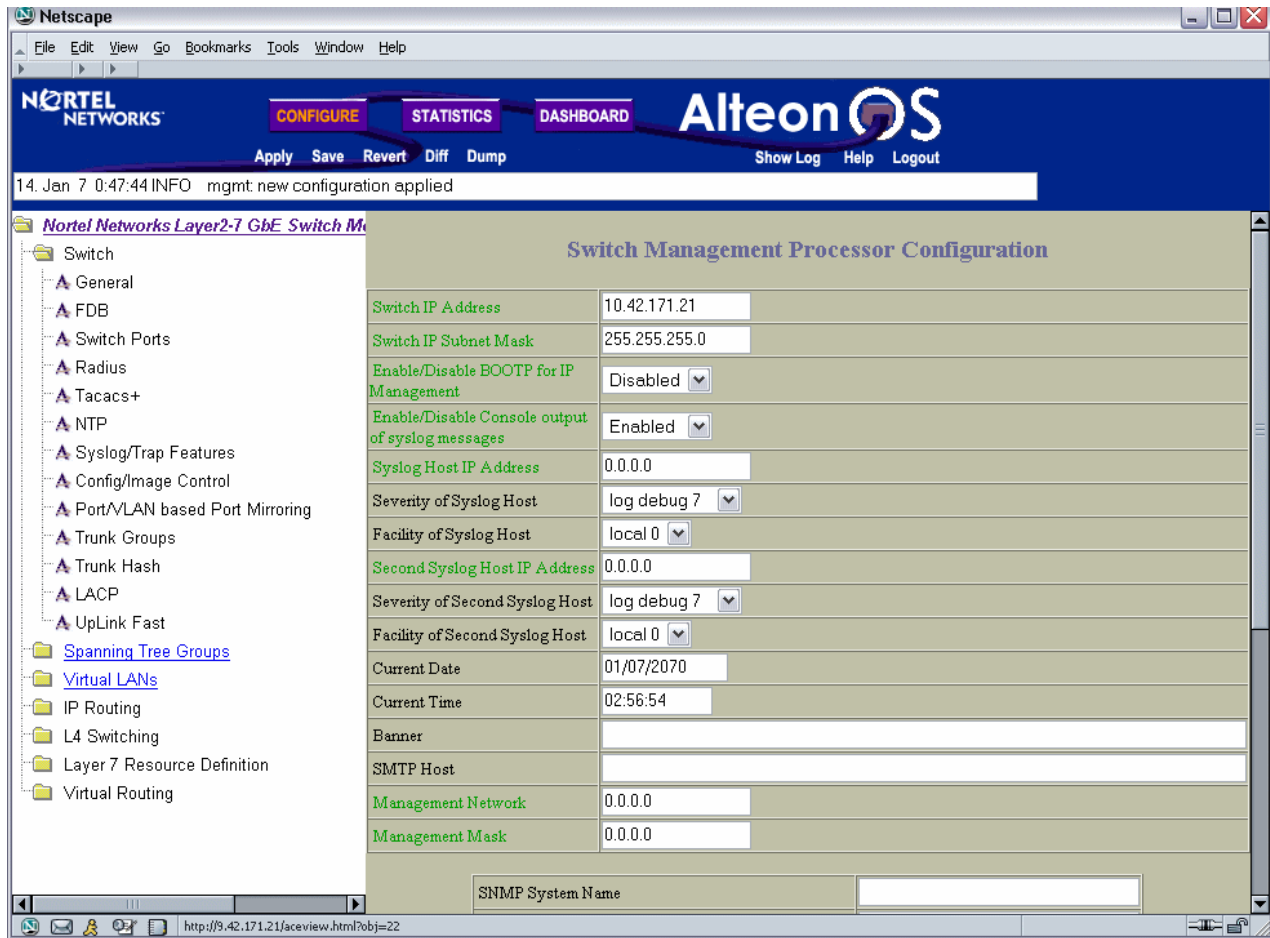


Figure 4-9 Switch information using BBI

You can browse through some of the other links in the left-hand frame to get more familiar with where the configuration options for the switch are located.

5. If you have made any changes to the switch and wish to save them, click **Apply** to apply the changes to the current running config.
6. Click **Save** to save the changes.

4.2.4 SNMP management: IBM Director

You can manage and monitor the Nortel Networks L2/7 GbESM switch module using SNMP with IBM Director. You can also use SNMP-based management systems, such as Tivoli Network Manager. The module supports the following SNMP capabilities:

- ▶ SNMP management stations can be configured to receive TRAP messages from the switch module. This is configured in the `/cfg/sys/ssnmp/` menu. Support is available for SNMPV3 as well as support for SNMP versions 1 and 2.
- ▶ SNMP Management Information Base (MIB) files are provided with every software image. These files can be imported to the MIB compiler, which is included with IBM Director and other network management products. The MIBs include Nortel proprietary extensions to the standard MIB1 and MIB2 objects. Both read and write access to these variables can be configured.

4.3 Multiple Nortel Networks L2/7 GbESMs in a BladeCenter

If there are two or more switches in a single IBM BladeCenter chassis, the management (MGTx) interfaces of all of the switches are on VLAN 4095.

The consequence of this is that all of the MGTx IP addresses configured through the Management Module Web interface should be on the same subnet as the Management Module internal and external port IP addresses (to allow for access through the Management Module). This configuration also makes it possible to Telnet from one switch module to another across the midplane of the chassis.

It is not possible to pass substantive data between switch modules across the midplane using the MGTx ports. The Nortel Networks L2/7 GbESM will not forward data between the MGTx ports and any of the internal (INTx) or external (EXTx) ports. If you want to pass data from one switch module to another, then the modules must be either cabled directly to each other or connected by way of an external switch or router.

4.4 Differences with L2/3 switch module

This section covers the functional differences between the L2/3 and L2/7 switch modules. It does not list major L4-7 functions such as Server Load Balancing not included in the L2/3 switch module. Those functions are enumerated in Chapter 3, "Introduction to Nortel Networks Layer 2-7 Gigabit Ethernet Switch Module" on page 15. Specific differences in commands are shown in Appendix A, "Filters on L2/3 and L2/7" on page 131.

4.4.1 Functions unique to the L2/7 GbESM Switch Module

The functions listed in this section are not available on the L2/3 GbESM Switch Module.

Default Gateway load balancing

On GbESM L2-7, Default Gateways 1 through 4 are used for load-balancing session requests. There is no load-balancing functionality within this feature in GbESM L2/3.

IP forwarding per interface

IP forwarding can be enabled and disabled per interface on GbESM L2-7. In GbESM L2/3, IP forwarding is a global configuration parameter only. Note that if there is no IP interface defined on a specific VLAN then there can be no forwarding (routing) onto or off of that VLAN.

Switch IP address via BOOTP

If available on your network, a BOOTP server can supply the switch with IP parameters so that you do not have to enter them manually.

4.4.2 Functions not included on the L2-7 GbESM switch module

The functions listed in this section are supported on the L2/3 switch module but are not available on the current software release on the L2-7 switch module.

Spanning Tree Protocols: 802.1w (RSTP) and 802.1s (MSTP)

Rapid Spanning Tree Protocol enhances the IEEE 802.1d Spanning Tree Protocol to provide rapid convergence on a single Spanning Tree Group. Multiple Spanning Tree Protocol extends the IEEE 802.1w Rapid Spanning Tree Protocol, to provide rapid convergence and support for multiple spanning tree groups.

In most instances it is advantageous to configure the L2-7 switch to use Layer 3 routing to support topologies with multiple uplink paths rather than using Spanning Tree.

802.1x Port Authentication

The 802.1x standard describes port-based network access control using Extensible Authentication Protocol over LAN (EAPoL). EAPoL provides a means of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics and of preventing access to that port in cases of authentication and authorization failures.

RIPv2

The current software release for the L2-7 GbESM does not support RIPv2; only RIPv1 is supported. This function will be added to the L2-7 GbESM in a future software release. In the interim, it is possible to use OSPF or static routes on the L2-7 GbESM.

ECMP (Equal Cost Multi-Path)

The current software release for the L2-7 GbESM does not support ECMP on learned or static routes to the same destination. This function will be added to the L2-7 GbESM in a future software release. In the interim, it is possible to use multiple default gateways on the GbESM. Traffic will be balanced across the available gateways if the appropriate option is configured.

HTTPS

The BBI cannot be accessed using HTTPS on the current software release on the L2-7 switch. This function will be added in a future software release. Secure access to the L2-7 switch using the Command Line Interface (CLI) with SSH.

SNMPv3

The current software release of the L2-7 GbESM does not support SNMPv3; only SNMPv1 is supported at this time. SNMPv3 support will be available in a future software release.

IGMP filtering

The current software release of the L2-7 GbESM does not support IGMP filtering. IGMP including snooping is supported.

802.1p Priority Queuing and QoS with DSCP

These traffic management functions are not included in the current software on the L2-7 GbESM. They will be added to a future software release.

Miscellaneous items

Other functions that are not supported are listed here.

FTP of firmware and configuration files

The L2-7 GbESM does not currently support use of FTP for uploading or downloading these files; only TFTP is supported.

Scheduled Reboot

Pre-scheduled, unattended reboots are not available on the L2-7 GbESM.

Operator commands

The commands that allow an operator-level user to query the NTP server, change passwords, and review unapplied configuration changes are not in the current L2-7 GbESM software. Administrator level users can perform all of these functions.

Time Zone

The `/cfg/timezone` command is not included in the current software for the L2-7 GbESM. It will be added in a future release.

4.4.3 Functions implemented differently on the L2/3 and L2-7 switches

This section shows those functions that are supported on both the L2/3 and L2-7 GbESM switch modules but that are implemented or configured differently.

Filters

Filters, also known as *Access Control Lists*, use different command syntax on the two switch modules. On both platforms, it is possible to filter on source and destination MAC address; source and destination IP address (with masking); source and destination port (TCP and UDP); protocol (TCP, UDP, ICMP, and others). An example of the configuration of the same filter on the two platforms is included in Appendix A, "Filters on L2/3 and L2/7" on page 131.

The key differences are:

- ▶ On the L2-7 GbESM, filters are defined in the `/cfg/slb/filt` menu. Filters are individually numbered and applied to individual ports where they are to be used. Filters for a particular port are executed in numerical order until the filter criteria on a particular filter are matched. When that happens, the action configured for that filter is executed and higher numbered filters on the port are not tested.
- ▶ On the L2/3 GbESM, filters are defined in the `/cfg/acl` menu. Filters called *acls* are grouped into groups and blocks which are numbered. Individual filters and blocks can be added to groups. Filters, filter groups, and filter blocks can all be assigned to ports. Filters within a block are executed in hardware concurrently. Filters, groups, and blocks on specific ports are executed in numerical order. This arrangement is more complex but allows a set of filters which together perform a specific function to be easily added to multiple ports.
- ▶ On the L2/3 GbESM, Quality of Service (QoS) functions are implemented using filters; this functionality is not currently available on the L2-7 GbESM.
- ▶ On the L2-7 GbESM, Network Address Translation (NAT) is implemented using filters; this functionality is not available on the L2/3 GbESM.
- ▶ The L2-7 GbESM includes the ability to filter at Layer 7, the ability to filter on the payload of a packet as opposed to lower-level filters that only inspect the headers.
- ▶ The Layer 2-7 GbESM includes the ability to perform application redirection which is configured using filters. This function is not supported on the L2/3 GbESM.

Miscellaneous items

This section describes other differences between the L2/3 and L2-7 GbESM.

IGMP snooping

The implementation of IGMP snooping on the L2/3 GbESM has richer functionality than that on the L2-7 GbESM.

Management Network Configuration Commands

On the L2-7 GbESM only a single IP address and mask can be specified in the `mnet` command. If a more elaborate management network specification is required then standard filters should be used to specify the source addresses to be allowed or denied and the destination addresses should be those of the switch itself (those specified in `/cfg/l3/if`).

Trunk hashing

On the L2-7 GbESM, Layer 2 and Layer 3 trunk hashing parameters can be configured independently, unlike the L2/3 GbESM where they are configured together.

Jumbo Frame support

On the L2-7 GbESM, Jumbo Frame support can be enabled or disabled on a per-VLAN basis. On the L2/3 GbESM, Jumbo Frame support is always enabled for all VLANs and cannot be disabled.

Static ARP entries

On the L2-7 GbESM, static ARP entries are configured using the `/maint/arp` command. On the L2/3 GbESM, they are configured using the `/cfg/l3/arp/static` command.



Introduction to server load balancing

This chapter provides a drill down on Layer 4 and Layer 7 functions with the goal being to describe how the underlying mechanisms lead to the benefits discussed elsewhere, what the deployment options are, and guidelines for choosing what features to use when. This section is *not* a guide on how to specifically configure the BladeCenter, GbESM, or other devices to achieve the features discussed. Several sample configurations are included in later chapters of this Redpaper. Additional information is available in the *GbESM Application Guide* and *Command Reference Guide*.

Other terms often used for L4-7 Switching are Server Load Balancing (SLB), Content Switching, or Internet Traffic Management (ITM).

5.1 L4-7 implementation requirements

When deploying Server Load Balancing (SLB), there are a few key aspects to consider:

- ▶ Identical applications and data must be available to each server in the same pool.
 - For applications:
 - Identical application images can be configured on each server in the pool.
 - Remote Deployment Manager can be used to provision an application image on a server before bringing it in to join a pool.
 - For data, each real server in the pool has access to the same data through:
 - Download of identical file systems, content or databases to each local drive
 - Use of a common shared (networked) file system or back-end database server
- ▶ One of the following must be true:
 - User interaction with the service is completed over a single TCP connection.
 - The user interaction with the service is stateless from TCP connection to TCP connection
 - For services that create unique user session-specific states on the application server that persist from TCP connection to TCP connection, either:
 - The state of the user interaction with the service must be trackable from TCP connection to TCP connection, either by the L4-7 switch, or on the user side with such information included in each subsequent request from the user.
 - The load-balancing algorithm used must lead to direct the same users to the same real servers over all reasonable periods where unique user-related state may have been created (for example, these services must use the minmisses or hash metrics). This generally results in less even load balancing, but all of the reliability value propositions of L4-7 switching are retained.
 - Although clients and servers can be connected through the same switch port, in the GbESM case, servers will tend to be connected to internal ports and clients will tend to be connected to external ports. This is the default configuration but each port in use on the switch can be configured to process client requests, server traffic, or both.

These might sound like a lot of restrictions but, in general, L4-7 switching can be configured to support just about any network-based application where the majority of interactions are between users and servers over a network. This is in contrast to high performance computing clusters, for example, where the majority of traffic is between CPUs within the cluster. Different techniques are needed in that environment. For network-oriented application delivery, it is usually not a question of *if* but *how*, as far as the value-added use of L4-7 switching is concerned.

Note: Switch ports configured for Layer 4 client/server processing can simultaneously provide Layer 2 switching and IP routing functions.

5.2 Layer 4 switching: How it works

Layer 4 switching operates on units of TCP connections. You cannot spray packets which are part of the same TCP connection across multiple servers. That means that a Layer 4 switch needs to be TCP-aware in the sense that it needs to be able to:

- ▶ Identify the beginning of a new TCP connection.
- ▶ Assign that connection to a real server (Figure 5-1).
- ▶ Make sure that all ensuing packets related to that TCP connection continue to be sent to the same real server.

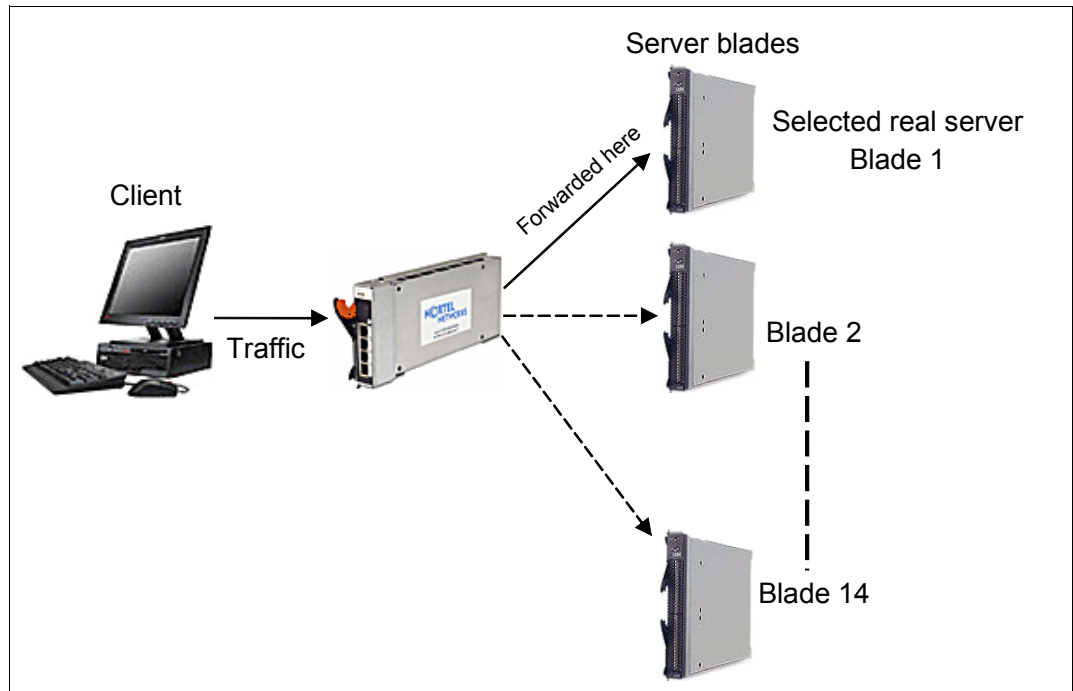


Figure 5-1 Layer 4 switching

As you can see in Figure 5-2 on page 50, TCP connections begin when a client sends a special TCP packet called a *SYN* packet (identified by the setting a bit in the TCP header) to the desired destination to initiate the creation of a connection. If the server is ready, willing, and able to respond, it will reply with a *SYN-ACK* packet that contains some parameters that need to be agreed to and used consistently by both sides of the connection.

The client then sends a *SYN-ACK-ACK* to the server, often with some actual data or a content request included (or immediately sends a request packet after sending the *SYN-ACK-ACK*), to kick off the real conversation part of the connection. When the data transfer related to the TCP connection is complete from the client's point of view, it will send a *FIN* packet to the server and start a final process of handshaking to tear down the connection. Alternatively, if the server has not heard from the client in a while, it will time out the connections.

With L4 switching, TCP connection dynamics change a little bit. L4 switching implements the concept of Virtual Services that are indexed by Virtual IP Addresses (VIPs). Here we can use a specific example to explain how Virtual Services and VIPs work. Consider a service with a host name, in a DNS sense, of *A.com*. Normally, there would be a server set up somewhere with that host name and IP address 100.2.2.2 (for example). With L4 switching, a GbESM takes ownership of the address 100.2.2.2 as a VIP. The GbESM has multiple (in this example, two) real server blades behind it capable of delivering the service *A.com*, with addresses that can be anything. They are only of local significance. In this case, we call them 10.1 and 10.2.

Immediate Binding

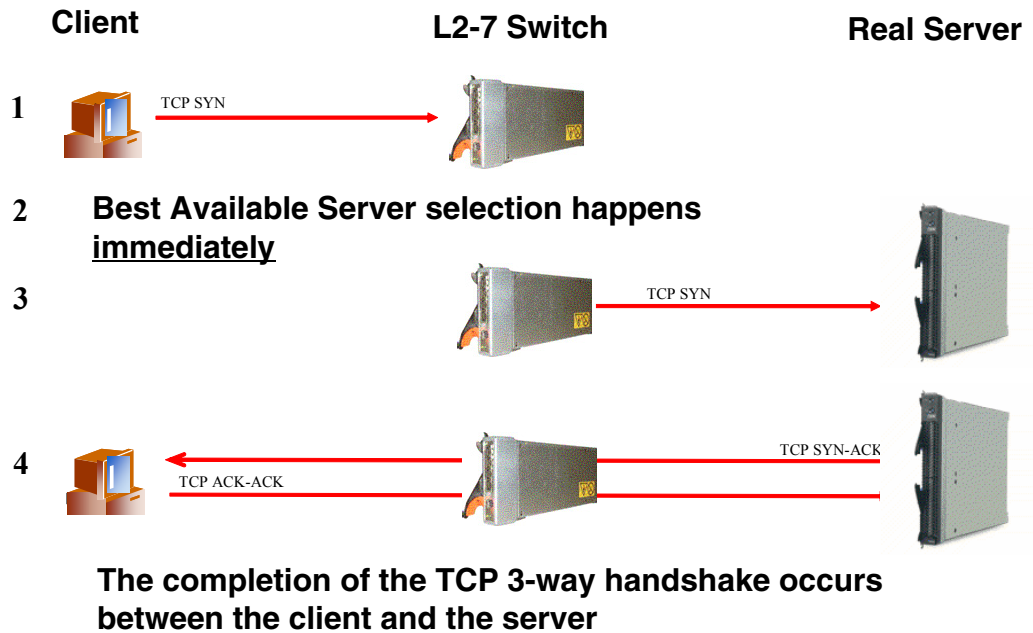


Figure 5-2 Layer 4 load balancing - Immediate binding

1. When the GbESM receives a packet from the user, it inspects the packet to determine what physical server to which to send it. If the packet belongs to an existing connection or user, the packet is sent to the server already assigned to that connection.
2. If the packet does not belong to an existing connection but instead is a SYN packet from a user desiring to establish a new connection, the GbESM determines which physical server to send the request to, based on some knowledge of the health and load of the candidate server blades, and some policy for making such a decision.
3. When one of the servers is chosen, the scheduler uses a redirection mechanism to dispatch the request to the chosen server blade. The request must arrive to that server blade addressed to its IP address and MAC address.
4. At this time, the GbESM creates some state information for this connection. The GbESM maintains a binding table, which reflects the present association of users (or connections, or sessions) to server blades.
5. The chosen server blade receives the packet, which has its own IP address in the destination field; thus to it this looks like a perfectly normal TCP connection request.
6. If the chosen application server is ready, willing and able, it answers the user with a TCP SYN-ACK. When received by the user, the TCP SYN-ACK must have the VIP address in the Source IP Address field. This is achieved by having the GbESM intercept the returned packet to translate the source IP address of the packet back into the Virtual IP address.
7. What would happen if the user's computer received a packet that had the chosen server blade's IP address in the source field instead of the virtual server's address? The user's computer would reject this frame with a TCP RESET. It would notice the imbalance between the destination IP address that it sent the TCP SYN request to, and the source IP address of the responding server for the TCP SYN-ACK.

8. The user's computer sends an ACK for the TCP SYN-ACK, and completes the three-way handshake. The GbESM, again, is the recipient of this frame because of the destination IP address. The GbESM inspects the packet, determines it is associated with an existing session (based on the user's IP address at a minimum), and sends the packet to the same chosen server blade. What would happen if the GbESM did not maintain session state, and it sent the ACK frame to a different server blade? The new server blade would receive a SYN-ACK-ACK frame associated with a session that was never started with a SYN, as far as it is concerned. It would reply back with a TCP RESET.
9. Every packet coming from the user to the load balanced set of server blades must pass through the GbESM so that it can inspect the frames and perform the appropriate load-balancing mechanism.

The ability to interdict the TCP handshaking between the client and server is what puts the Layer 4 into L2-7 GbESM.

The operation where the target server blade is chosen upon the arrival of a TCP SYN packet is called *Immediate Binding* and is what differentiates Layer 4 switching from Layer 7 switching.

In executing this process, the GbESM must execute a couple of critical functions:

1. Deciding the *best available* server.
2. Modifying packets to and from the target server to ensure the proper connectivity throughout the life of the TCP connection.

5.3 Layer 7 Switching: How it works

As we discuss in 5.2, "Layer 4 switching: How it works" on page 48, Immediate Binding is the process by which a given server blade is selected for a TCP connection session at the time a TCP SYN packet arrives. The use of Immediate Binding is, by definition, equivalent to Layer 4 switching.

Layer 4 switching is a very powerful technique. However, it does suffer from one shortcoming: no information about the nature of a request and little-to-no information about the generator of the request is available at the time that the decision is made as to which server blade receives the request. Often that does not matter. Often, all server blades can service any kind of request. Each request to an application adds about the same amount of load as any other. All users are considered of equal importance. Lastly, the nature of the application either requires no persistence of connectivity to one server blade across multiple sequential connections or the Layer 4 persistence features provide whatever is required. When these conditions are all met, Layer 4 switching suffices.

Sometimes, it is useful or necessary, to know something about the nature of the arriving application request or the user making the request. The problem is that the user does not divulge such information until after a TCP connection has been established. In light of that, another optional technique has been developed where the decision as to where to send an arriving request is not made until after a TCP connection has been made and a higher layer application request (such as in the case of HTTP requests) has been made and examined. This technique is called *Delayed Binding* and is, by definition, equivalent to Layer 7 switching. Layer 7 techniques are not limited to HTTP-based applications, but much of the development of Layer 7 techniques and production use of such techniques is centered around HTTP environments.

As HTTP rapidly becomes more prevalent as an underlying protocol for application delivery, the limitation associated with saying applications and architecture are *HTTP-oriented* is

rapidly going away, just as *IP-oriented* went away in the late 1980s and early 1990s. Look at the growth of Web sites, Web mail, Web-based applications based on platforms such as IBM WebSphere, Web versions of packaged enterprise applications such as SAP and Peoplesoft, and Siebel, Web services, XML, and so on, all of which use HTTP as an underlying protocol.

This section discusses the Layer 7 capabilities of the GbESM, with most of the examples based on HTTP environments.

Some example benefits that can be achieved by deploying Layer 7 switching include:

- ▶ Web sites can have so much content associated with their domain name that the content needs to be split up across multiple file systems. You could allow each Web server access to each file system by cross-mounting all the file systems, but this gets unwieldy as the number of file systems gets larger and if it routinely changes. Another approach is to assign access to portions of the directory space to certain Web server clusters, but still advertise the site under one domain name, such as `www.ibm.com`. The GbESM, as a front-end to this site, must be able to inspect the URL request (including filename and pathname) and send the requests for `www.ibm.com/marketing/` to one server, `/research/` to another server, `/admin/` to another server, and so forth.
- ▶ There can be a marketing advantage to advertising a single domain name for a particular service, but having support for discrete communities behind that single domain name. In that way, the number and type of servers supporting each community can be tuned to their needs. If different levels of service are warranted between the communities, traffic can be managed to achieve that.
- ▶ It can be useful to have servers that generate dynamic content to be optimized for the computation/processing function and servers that provide relatively static data such as images, HTML text, and so forth, optimized for fast retrieval from disk storage. URL awareness allows the GbESM to look for requests with dynamic server page calls or CGI script executions, versus static Web page element requests. Dynamic requests would be sent to the application-optimized servers and the static requests would be sent to the storage-optimized servers.

TCP sessions using Delayed Binding (Layer 7 switching) go through the steps in Figure 5-3 on page 53:

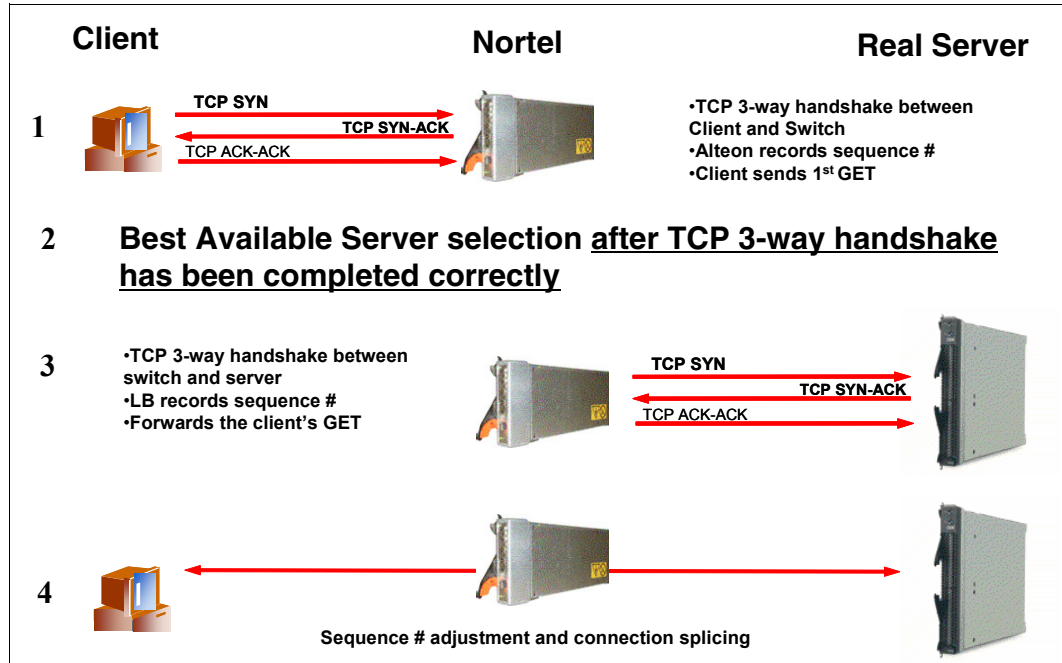


Figure 5-3 Layer 7 load balancing - Delayed Binding

1. Assume for this example that the application being accessed by the user is browser-based and uses HTTP for its communication protocol.
2. The user loads the browser on her computer workstation, then selects a Web site and URL, such as `www.ibm.com/main.html`.
3. The browser must communicate with the server that represents `www.ibm.com` using an IP address, so it must resolve the hostname portion of the URL (`www.ibm.com`) to an IP address. The browser (really the DNS resolver part of the browser) sends a DNS query to its configured local DNS server. This process is usually referred to as *DNS Resolution*.
4. The local DNS performs the necessary functions to return the IP address representing `www.ibm.com` back to the user's workstation. It might have the address already in cache, or it might have to query one or more known DNS servers to obtain the address.
5. When the browser has learned the IP address for the server, the browser begins the process of establishing a TCP connection to the server to retrieve the file (`main.html`) specified in the URL. This process is typically called the *TCP 3-way handshake*.
6. The user's computer forms a special frame, called a TCP SYN or synchronization frame, with its own IP address as the source IP address and the server's IP address as the destination.
7. Part of the frame defines what application the browser is requesting data from. In the case of TCP applications, the TCP Destination Port defines what application is being accessed. In this example, the well-known TCP port for HTTP (port 80) is placed in the destination TCP port field.
8. The server receives the frame, then passes it through several tests to see what to do:
 - Is the destination IP address my own IP address? (Yes)
 - Is the destination TCP port one to which I have an application listening? (Yes, HTTP)
 - Do I have resources to handle the request? (Yes)
9. After passing the frame through these tests, the server now answers the user's browser with a TCP SYN-ACK. The server informs the browser that it received the request and is ready to complete the TCP connection.

10. The user's computer receives the TCP SYN-ACK. After examining it, the computer sends an ACKnowledgement frame of its own back to the server, informing the server that the TCP connection is complete. The browser is ready to request data from the server.
11. The browser now sends its first data request to the server: the HTTP GET request. The GET request looks very simple. In addition to providing information that is required by the HTTP protocol in the request, the GET request is a clear-text message that says *GET /main.html*.
12. The server receives this GET request, examines the path/filename specified in the request, retrieves the file from disk (or other storage location such as an NFS file server) and formulates a proper HTTP response frame with the contents of the main.html file in it.
13. This main.html file might look similar to Example 5-1:

Example 5-1 main.html file

```
<CENTER><IMG SRC="welcome.jpg" </CENTER></P>
<P> At IBM, we strive to lead in the invention, development and
manufacture of the industry's most advanced information
technologies, including computer systems, software, storage
systems and microelectronics.
```

```
We translate these advanced technologies into value for our
customers through our professional solutions, services and
consulting businesses worldwide. </P>
```

```
<P><CENTER><IMG SRC="divbw.jpg" NATURALSIZEFLAG="0">
```

14. This file is mostly text, with some markup tags for HTML formatting. Within this example file, notice two "IMG SRC=..." tags. These tags inform the browser to retrieve the files specified, two JPEG image files here. The browser then retrieves these additional files using HTTP, potentially using the same TCP connection or opening additional TCP connections.
15. Displaying a Web page on a browser from a Web site involves retrieving the first specified file, that then informs the browser of all the subordinate files that are to be retrieved. These files can be images, additional HTML text, sounds, java applets, any file types the browser and server are capable of handling.
16. The location of referenced files might even be different. The HTML page can have a file tag in it that instructs the browser to go to another server to retrieve the images, such as images.ibm.com/welcome.jpg. This would require opening a separate TCP connection to this new server to retrieve the image files.
17. After the last element on the Web page has been retrieved, all TCP connections are closed with another special TCP frame called a TCP FIN.

5.4 Server health checking

Many times, an application process can fail without the knowledge of the TCP or IP processes. In that case, a ping test or even a TCP connection test can succeed even though the server is not able to service application requests. To detect such failures, a higher-level test exercising the application logic is required.

Over the years, the software that the GbESM is based on has been used in a lot of different environments for a lot of different purposes. The GbESM uses a suite of application-level

tests, some predefined and others with user and field definable parameters that can be used to exercise application logic. The list of presently available predefined tests is:

HTTP health checks

HTTP-based health checks can include the hostname for HOST: headers. The HOST: header and health check URL are constructed from the following components: If the HOST: header is required, an HTTP/1.1 GET will occur. Otherwise, an HTTP/1.0 GET will occur. HTTP health check is successful if you get a return code of 200.

UDP-based DNS health checks

This health check is performed by sending a UDP or TCP-based query (for example, for www.nortelnetworks.com), and watching for the server's reply. The domain name to be queried can be modified by specifying the content command if you need to change the domain name.

HTTPS/SSL server health checks

The sslh health check option allows the switch to query the health of the SSL servers by sending an SSL client Hello packet, and then verifying the contents of the server's Hello response.

WAP gateway health checks

There are two types of WAP gateway health checks:

- ▶ Wireless Session Protocol (WSP) content health checks, the *unencrypted* mode of sending WML traffic (similar to HTTPS)
- ▶ Wireless Transport Layer Security (WTLS) health checks, an *encrypted* mode of sending WML traffic (similar to HTTP)

GbESM Switch Module OS provides a content-based health check mechanism where customized WSP packets can be sent to the gateways, and the switch can verify the expected response in a manner similar to scriptable health checks. Wireless Session Protocol content health checks can be configured in two modes: connectionless and connection-oriented.

LDAP health checks

The LDAP health check process consists of three LDAP messages over one TCP connection:

- ▶ Bind request
The switch first creates a TCP connection to the LDAP server on port 339, the default port. After the connection is established, the switch initiates an LDAP protocol session by sending an anonymous bind request to the server.
- ▶ Bind response
On receiving the bind request, the server sends a bind response to the switch. If the result code indicates that the server is alive, the switch marks the server as up. Otherwise, the switch marks the server as down because the server did not respond within the timeout window.
- ▶ Unbind request
If the server is active, the switch sends a request to unbind the server. This request does not require a response. It is necessary to send an unbind request because the LDAP server might crash if too many protocol sessions are active.

Other application-specific health checks

GbESM exercises each listed application here to the extent necessary to verify that the appropriate process is running on the server:

- ▶ FTP server health checks
- ▶ POP3 server health checks
- ▶ SMTP server health checks
- ▶ IMAP server health checks
- ▶ NNTP server health checks
- ▶ RADIUS server health checks

Script-based health checks

Although the GbESM has a lot of *canned*, stock, health checks for testing a lot of environments, the number of applications and environments the GbESM can be used in is far greater than the full extent of these tests. Therefore, a mechanism has been created to allow user and field creation of tests related to specific applications. These health checks you can create are called *send/expect*. These structured health checks dynamically verify application and content availability using scripts. These scripts execute a sequence of tests all of which are oriented towards sending a particular request and checking for delivery of the expected response.

Link health checks

Link health checks are performed at the Layer 1 (physical) level, and should be used only on servers that do not respond to any other type of health check. Intrusion Detection Servers (IDSs) and Firewalls can fall into this category. The server is considered to be *up* when the link (connection) is present, and *down* when the link is absent. This type of health check should be used with caution on blade servers because of the unusual way that the chassis manages the link state of the internal ports. These ports are actually data pathways across the chassis' midplane.

5.5 Advanced server load-balancing functions

This section provides some detail function on several of the advanced functions provided by the Layer 2-7 GbESM.

5.5.1 Persistence

Layer 7 switching is often needed to deal with issues of application state and user persistence in terms of sticky connectivity to the same server across a series of sequential TCP connections.

SSL session tracking

SSL sessions typically span many TCP sessions. The GbESM must ensure that the client remains connected to the originally selected (and authenticated) server for the duration of the SSL session. Source IP Address Binding can meet this goal, but at the sacrifice of some load balancing evenness. If a proxy server is involved, there might be further issues. When a client and specific server handshake to begin an encrypted SSL session, a unique SSL session ID is assigned.

One way to ensure SSL persistency is for the GbESM to monitor the creation of SSL sessions and the assignment of SSL Session IDs. When subsequent packets arrive with the same SSL Session ID, they can be redirected to the same server that was involved in the original SSL

session creation. Persistence based on SSL Session ID ensures completion of complex transactions even in proxy server environments.

This approach requires delayed binding, enough memory to store the SSL session state information, and enough GbESM power to handle all the extra filtering and packet processing required to track the SSL handshake protocol and match Session IDs. The implementation also requires an aging algorithm, to age out SSL Session IDs.

Source IP Address Binding can provide the same persistence, with the trade-off being less load-balancing evenness. Load-balancing evenness is lost because the activity timers must be set to fairly long values, on the order of magnitude of the time you allow Session IDs to remain valid. This time can be an hour, six hours, possibly even 24 hours. For that period of time, all traffic from the Source IP range will reset the timer, so you could end up having those users bound to the same server forever. If you only apply the policy to SSL traffic, or if you only allow short SSL sessions (which we generally believe is a good idea), then you do not need to turn on the SSL Session ID Monitoring feature. However, it is still a good option to have in your back pocket.

Shopping Cart persistence and cookie-based persistence

At some Web sites, a user browses though the site and indicates items he or she might buy by placing them into a *shopping cart*. In some application deployments, the knowledge of what items are held in a particular shopping cart is stored on only one server until the user indicates the desire to buy the items.

The user must remain connected to that server for the duration of the shopping session, which can span many, many TCP sessions. Then, if the purchase requires an SSL session (for communicating credit card information, usually), the user must continue to remain connected to that server. Sometimes, when the user comes out of the SSL session and continues with HTTP, the persistence must be maintained.

This is a tricky set of requirements. Providing appropriate persistence for this situation requires tracking of sequences of TCP sessions from HTTP to SSL and back. Layer 4 Source IP Address Binding over address ranges can provide the persistence required, but at the cost of significantly reduced load balancing evenness if the activity timers need to be set to a long value. How long the activity timers must be is largely related to how long you want the customer to have an active Shopping Cart before they must make a purchasing decision. If that is hours or days, the activity timers would have to be too long for Source IP Address Binding to work effectively.

There is another choice, which is cookie-based persistence. Cookies are a mechanism for maintaining state between clients and servers. When the server receives a client request, the server issues a *cookie*, or token, to the client, which the client then sends to the server on all subsequent requests. Using cookies, the server does not require authentication, the client IP address, or any other time-consuming mechanism to determine that the user is the same user that sent the original request. In the simplest case, the cookie can be just a customer ID assigned to the user. It can be a token of trust, allowing the user to skip authentication while his or her cookie is valid. It can also be a key that associates the user with additional state data that is kept on the server, such as a shopping cart and its contents. In a more complex application, the cookie can be encoded so that it actually contains more data than just a single key or an identification number. The cookie in the complex environment can contain the user's preferences for a site that allows their pages to be customized.

Permanent and temporary cookies

Cookies can either be permanent or temporary. A *permanent* cookie is stored on the client's browser, as part of the response from a Web site's server. It will be sent by the browser when

the client makes subsequent requests to the same site, even after the browser has been shut down. A *temporary* cookie is only valid for the current browser session. Similar to a SSL Session based ID, the temporary cookie expires when you shut down the browser.

Cookie modes of operation

GbESM Switch Module OS supports the following modes of operation for cookie-based session persistence: insert, passive, and rewrite mode.

Insert cookie mode

In the *insert* cookie mode, the switch generates the cookie value on behalf of the server. Because no cookies are configured at the server, the need to install cookie server software on each real server is eliminated. In this mode, the client sends a request to visit the Web site. The switch performs load balancing and selects a real server. The real server responds without a cookie. The switch inserts a cookie and forwards the new request with the cookie to the client.

Passive cookie mode

In *passive* cookie mode, when the client first makes a request, the switch selects the server based on the load-balancing metric. The real server embeds a cookie in its response to the client. The switch records the cookie value and matches it in subsequent requests from the same client.

Note: Passive cookie mode is recommended for temporary cookies. However, you can use this mode for permanent cookies if the server is embedding an IP address.

Rewrite cookie mode

In *rewrite* cookie mode, the switch generates the cookie value on behalf of the server, eliminating the need for the server to generate cookies for each client. Instead, the server is configured to return a special persistence cookie which the switch is configured to recognize. The switch then intercepts this persistence cookie and rewrites the value to include server-specific information before sending it on to the client. Subsequent requests from the same client with the same cookie value are sent to the same real server.

5.5.2 Load balancing metrics

Metrics or algorithms for load balancing are used for selecting which server blade in a group will receive the next client connection. The options for metrics are described in this section. When considering what metric or combinations of metrics to use, you should remember that perfectly equal load balancing between server blades in a Virtual Service Pool is generally difficult to achieve. It is also generally irrelevant. Usually, server blades deliver application services very well when lightly loaded and as the load increases, until a saturation point is reached, representing a *knee in the load versus latency* curve. The main purpose of load balancing is to maintain application availability at whatever level of performance the healthy resources allow. What this generally means in practice is to keep all servers working at operating points below the knee of the curve.

Minimum misses

For L4 switching, the client source IP address and real server blade IP address are used. All requests from a specific client are sent to the same server. This metric is useful for applications where client information must be retained on the server between sessions. With this metric, server load becomes most evenly balanced as the number of active clients with different source or destination addresses increases. When selecting a server, the switch calculates a value for each available real server based on the relevant IP address

information. The server with highest value is assigned the connection. The *minmisses* metric attempts to minimize the disruption of persistency when servers are removed from service. This metric should be used only when persistence is a must.

Hash

The *hash* metric uses IP address information in the client request to select a server. For SLB, the client source IP address is used. All requests from a specific client will be sent to the same server. This option is useful for applications where client information must be retained between sessions.

Note: The hash metric provides more distributed load balancing than *minmisses* at any given instant. It should be used if the statistical load balancing achieved using *minmisses* is not as optimal as desired. If the load balancing statistics with *minmisses* indicate that one server is processing significantly more requests over time than other servers, consider using the hash metric.

Least connections

With the *leastconns* metric, the number of connections currently open on each real server, is measured in real time. The server with the fewest current connections is considered to be the best choice for the next client connection request. This option is the most self-regulating, with the fastest servers typically getting the most connections over time.

Round robin

With the *roundrobin* metric, new connections are issued to each server in turn. That is, the first real server in the group gets the first connection, the second real server gets the next connection, followed by the third real server, and so on. When all the real servers in this group have received at least one connection, the issuing process starts over with the first real server.

Response time

The *response* metric uses real server response time to assign sessions to servers. The response time between the servers and the switch is used as the weighting factor. The switch monitors and records the amount of time it takes for each real server to reply to a health check to adjust the real server weights. The weights are adjusted so they are inversely proportional to a moving average of response time. In such a scenario, a server with half the response time as another server will receive a weight twice as large.

Bandwidth

The *bandwidth* metric uses real server octet counts to assign sessions to a server. The switch monitors the number of octets sent between the server and the switch. Then, the real server weights are adjusted so they are inversely proportional to the number of octets that the real server processes during the last interval. Servers that process more octets are considered to have less available bandwidth than servers that have processed fewer octets. For example, the server that processes half the amount of octets over the last interval receives twice the weight of the other servers. The higher the bandwidth used, the smaller the weight assigned to the server. Based on this weighting, the subsequent requests go to the server with the highest amount of free bandwidth. These weights are automatically assigned.

Weights for real servers

Weights can be assigned to each real server. These weights can bias load balancing to give the fastest real servers a larger share of connections. Weight is specified as a number from 1 to 48. Each increment increases the number of connections the real server gets. By default,

each real server is given a weight setting of 1. A setting of 10 would assign the server roughly 10 times the number of connections as a server with a weight of 1.

Note: Weights are not applied when using the hash or minmisses metrics.

Connection time-outs for real servers

In some cases, open TCP/IP sessions might not be closed properly (for example, the switch receives the SYN for the session, but no FIN is sent). If a session is inactive for 10 minutes (the default), it is removed from the session table in the switch. Examples of the use of this function are shown in Chapter 6, "Load balancing with WebSphere Portal" on page 75 and in Chapter 7, "Load balancing with Citrix MetaFrame and Microsoft Terminal Services" on page 101.

Maximum connections for real servers

You can set the number of open connections each real server is allowed to handle for SLB. Values average from approximately 500 HTTP connections for slower servers to 1500 for quicker, multiprocessor servers. The appropriate value also depends on the duration of each session and how much CPU capacity is occupied by processing each session. Connections that use a lot of Java or CGI scripts for forms or searches require more server resources and thus a lower *maxcon* limit. You might want to use a performance benchmark tool to determine how many connections your real servers can handle. When a server reaches its *maxcon* limit, the switch no longer sends new connections to the server. When the server drops back below the *maxcon* limit, new sessions are again allowed.

5.5.3 Key configuration parameters

This section documents other parameters which can be helpful in specific circumstances.

Real Server configuration parameters

inter	sets the interval between health checks. The range is from 2-60 seconds with a default of 2. There is a tradeoff between the speed at which the switch can detect a failed server and the load generated by the health checks.
addport/remport	allows a server to support multiple ports for the same service with a round-robin balance between the ports. This would enable, for example, multiple instances of Apache to be deployed on the same physical server using different TCP port numbers.
submac	causes the switch to substitute its MAC address as the origin on inbound processing of an SLB request so that the reply is sent to its MAC rather than that of the original requestor. This is not required but is intended for use when the requestor, the real server address, and the virtual server address are all part of the same subnet. (for example, no Layer 3 routing is taking place).
proxy	enables the use of a proxy IP address when a server initiates a session. This allows a server to act as a client and initiate a request that will be load balanced without the possibility that the request will be balanced back to the originating server. It also allows servers configured with non-routable (10.x.x.x) real addresses to initiate requests to servers on the public Internet or corporate intranets without exposing their real addresses.

Virtual Service configuration parameters

rport	specifies the TCP port on the real servers which will actually receive incoming requests. For example, service 80 can be configured with rport 8080. Incoming http requests will then transparently be forwarded to port 8080. Examples of this use can be seen in Example 6-4 on page 90 and Example 6-6 on page 97.
pbind	enables one of the types of persistent binding that the switch supports. All of these function to ensure that a client will continue to be serviced by the same server to which it was originally assigned. The available modes are: clientip - client's IP address; cookie - use a cookie to ensure repeat connections go to the same server; sslid - use SSL session ID to ensure repeat connections go to the same server. This command is also used in Example 6-4 on page 90.
dbind	enables delayed binding, which is required for all Layer 7 functions including the use of cookies and SSL session IDs for persistence as mentioned previously.

5.6 Design examples and best practices

This section discusses recommended designs for use with the Layer 2-7 GbESM.

5.6.1 Definitions

These are terms we use when discussing design.

Client processing	a configuration option which identifies ports on the GbESM through which clients can be reached
Real server	a server which can be identified by its IP address. Note that a VMware virtual server is a real server from the switch's perspective.
Real server group	identified a collection of real servers associated with one or more services
Real IP service	the IP address of a real server, which is configured on the switch a particular service offered by one or more real servers, identified by its TCP or UDP port number (for example, 80 for Web services)
Server processing	a configuration option that identified ports on the GbESM through which servers can be reached
Virtual IP	the IP address of a virtual server, or in other words the address of a load balancing pool or servers.
Virtual server	a pool of servers which offer the same service and which has an IP address configured.

5.6.2 Minimum configuration required for SLB

This section presents a minimal configuration for a single load-balanced service. This is intended as a building block for more complex configurations; several complex examples are included in later chapters.

The minimum requirements to make load balancing function are:

- Configure one or more real servers where the service will run, identifying them by their IP addresses.

- ▶ Define a group with members that are the real configured servers.
- ▶ Configure a virtual server which will represent the pool of real servers, with an IP address different from that of any of the real servers.
- ▶ Configure a service, which identifies the TCP/UDP port which clients will use to access the service.
- ▶ Configure client and server processing on appropriate ports of the GbESM to identify where clients and servers can be reached.

Example 5-2 is a sample configuration fragment. This configuration uses servers in blade slots 1 and 2. It uses the default values for load-balancing metric, and for health-checking of the real servers.

Example 5-2 Minimal configuration for Server Load Balancing

```

/* define two real servers at addresses 9.1.1.10 and .11
/* these are the addresses which should be configured to the OS on the two servers
/cfg/slb/real 1
    enable
    rip 9.1.1.10
/cfg/slb/real 2
    enable
    rip 9.1.1.11
/*
/* create a group with the two real servers as members
/cfg/slb/group 1
    add 1
    add 2
/*
/* create a virtual server at address 9.1.1.100
/* this address must be different from all the real server addresses and can
/* be on a different subnet if desired
/cfg/slb/virt 1
    vip 9.1.1.100
    ena
/* define http service on the virtual server using the real servers in group 1
/cfg/slb/virt 1/service http
    group 1
/*
/* identify internal ports 1 and 2 as server ports. Typically all of the internal
/* ports will be server ports.
/cfg/slb/port int1
    server ena
/cfg/slb/port int2
    server ena
/* identify external port 1 as a client port. Typically all of the external ports will be
/* client ports; they can also, simultaneously, be server ports in some cases.
/cfg/slb/port ext1
    client ena

```

5.6.3 Multiple services examples and considerations

Most customers will have multiple services, more than one Virtual Server, and more than two real servers. This section will present the rules and limits on what can be configured.

Architectural limits

The GbESM can handle up to 64 real servers, up to 64 virtual servers, and up to 256 services. The real servers can be in the same chassis where the GbESM is located, in another chassis in the same facility, or can be rack-mounted or tower servers outside of a chassis.

General constraints

There is substantial flexibility in the ways in which real servers, virtual servers, and groups can be configured. The rules are:

- ▶ A real server can be a member of more than one group. Groups can have overlapping membership with other groups and can even have identical membership with other groups.
- ▶ A virtual server can have the same or different groups for different services, but only one group for any specific service.
- ▶ If a virtual server uses the same group for more than one service, then those services are bound together. This has the following consequences:
 - If one of the bound services is down (fails its health check) on a real server, then all the other services on that real server are disabled from the switch perspective. No requests for any service associated with the group will be sent to that server by the switch.
 - If you want to not have all the services on a real server disabled when one service is unavailable, then the services must be associated with different groups. This is true even if it results in two or more groups having the same membership.

An example of this is shown in Example 5-3:

- ▶ HTTP and HTTPS, running on real servers 1 and 2, fail together on virtual server 1 (9.1.1.100)
- ▶ HTTP and HTTPS running on real servers 1 and 2 fail independently on virtual server 2 (9.1.1.105)
- ▶ FTP is reachable on virtual server 2 (9.1.1.105) and runs on real servers 2 and 3. It has to binding to the HTTP or HTTPS services.

Example 5-3 Multiple Services Load Balancing

```
/* define real servers at addresses 9.1.1.10 .11 and .12
/* these are the addresses which should be configured to the OS on the two servers
/cfg/slb/real 1
  enable
  rip 9.1.1.10
/cfg/slb/real 2
  enable
  rip 9.1.1.11
/cfg/slb/real 3
  enable
  rip 9.1.1.12
/*
/* create groups
/cfg/slb/group 1
  add 1
  add 2
/cfg/slb/group 2
  add 1
  add 2
/cfg/slb/group 3
```

```

        add 2
        add 3
/*
/* create a virtual server at address 9.1.1.100
/* this address must be different from all the real server addresses and can
/* be on a different subnet if desired
/cfg/slb/virt 1
    vip 9.1.1.100
    ena
/* define http and https services on the virtual server using the real servers in group 1
/* if either service fails its health check on one of the servers the other service will be
/* marked "blocked" by the switch and will not receive any requests
/cfg/slb/virt 1/service http
    group 1
/cfg/slb/virt 1/service https
    group 1
/* define another virtual and configure it so that http and https are independent
/* despite using the same real servers
/cfg/slb/virt 2
    ena
    vip 9.1.1.105
/*
/cfg/slb/virt 2/service http
    group 1
/cfg/slb/virt 2/service https
    group 2
/* define ftp service using virtual address 2 - independent of http and https and running
/* on a different set of servers (group 3 - real servers 2 & 3)
/cfg/slb/virt 2/service ftp
    group 3
/*
/* identify internal ports 1-3 as server ports. Typically all of the internal
/* ports will be server ports.
/cfg/slb/port int1
    server ena
/cfg/slb/port int2
    server ena
/cfg/slb/port int3
    server ena
/* identify external port 1 as a client port. Typically all of the external ports will be
/* client ports; they can also, simultaneously, be server ports in some cases.
/cfg/slb/port ext1
    client ena

```

5.6.4 SLB across multiple BladeCenter chassis

There are at least two reasons for using SLB in such a way that it spans multiple Blade Center chassis.

- ▶ First, the application may simply need more than 14 servers.
- ▶ Second, High Availability can be enhanced to enable an application to survive the failure of an entire chassis.

In either case, the GbESM can be configured to include servers other than those in the same chassis in a load balance group. The servers can be in a different chassis or they can be stand alone servers. The requirements for doing this are:

- ▶ The EXTERNAL port(s) through which the servers are reached must be configured with server processing enabled. (`/cfg/slb/port EXT<x>/server ena`)

- ▶ Any necessary IP routes to reach the servers must be configured. This is actually no different than the case where all the servers are in the same chassis but it might be more likely that the external servers are on a different subnet or VLAN.
- ▶ If all of the servers supporting an application must have access to the same data store then this must be provided such as by using a SAN or NAS server.

5.7 High availability design considerations

This section provides an explanation of the Trunk Failover and Hot Standby features, the Broadcom Advanced Services Protocol driver, and VRRP and of how they work together in a load-balancing configuration to provide a High Availability IBM BladeCenter environment.

When the High Availability features are used with load balancing, it is possible to protect against any of the following failures, and keep an application available for use:

- ▶ Server failure

As long as one of the members of a load balancing pool supporting a particular application is up, the application can be used.

- ▶ Switch failure

By configuring VRRP, and the Broadcom BASP driver, the environment will recover from a switch failure or removal and continue operating on all available servers. There are extensions to the VRRP standard implemented on the L2-7 GbESM to allow VRRP to better support server load balancing.

- ▶ Switch uplink failure or upstream device failure

By configuring VRRP, BASP, and hot standby, the GbESM can detect when it cannot successfully forward traffic to the appropriate upstream device outside of the Blade Center and redirect traffic through the other GbESM.

Note that multiple simultaneous failures easily can cause an application to be unreachable. For example, if all of the servers supporting a given application fail, then the application will not be available. Similarly, if a GbESM module fails and the core switch which is upstream from the second GbESM fails at the same time, the application will be unreachable.

5.7.1 Introduction to Hot Standby and Trunk Failover

Hot Standby works by shutting down designated internal ports (connected to the blade servers) when the upstream links go down, isolating the switch from the remainder of the network. The selected internal ports are put into disabled state, and the servers react as though the cable to the network card on a free-standing server had been unplugged. When the external links recover, the internal ports are reenabled.

Trunk Failover performs a very similar function. Either Hot Standby or Trunk Failover (but not both simultaneously) can be used with VRRP to support a high availability design. The differences between Hot Standby and Trunk Failover are:

- ▶ Hot Standby requires a connection between the two switches (either directly wired or with a common upstream switch) on a VLAN which is used for no other purpose. (VRRP requires a connection as well but does not require a dedicated VLAN).
- ▶ Hot Standby is only triggered when all of the external ports are down. This can be problematic if only some of the external ports connect to the upstream switch and others connect to external servers, intrusion detection appliances, or network analyzers.

- ▶ Hot Standby requires the use of VRRP “group” mode, where all of the VRRP resources will cut over from one switch to another together.
- ▶ Hot Standby allows each internal port to be configured to be disabled or not when the uplink ports fail. If some servers communicate exclusively with other servers in the chassis, their ports may not need to be disabled.
- ▶ Trunk Failover allows specific trunks, containing external ports, to be identified as critical; other external ports states are ignored for purposes of Trunk Failover.
- ▶ Trunk Failover as implemented on the current release (20.2.2.x) of software on the L2-7 switch will always disable all of the internal ports when the critical uplink trunks are down. Additional granularity will be introduced to Trunk Failover in the next software release.

Hot Standby and Trunk Failover are both intended to prevent the following failure modes, when used as part of a High Availability design:

- ▶ The connections between a Nortel GbESM and upstream switches fail, due to a cable problem, the failure of the upstream switches, or any other cause.
- ▶ The Nortel GbESM continues to function, and the server blades continue to send traffic to it.
- ▶ The Nortel GbESM, having lost its upstream connections, has no place to forward the server blades’ traffic and therefore discards it.

Note: If the Nortel GbESM itself fails, High Availability can be provided through the use of other features such as NIC teaming and VRRP. The Hot Standby feature is not necessary to protect against failures of that nature.

5.7.2 Introduction to NIC Teaming

NIC Teaming is a function that is provided by Broadcom and Intel, the manufacturers of the NIC chips used on the Blade Servers, in software. Broadcom provides the Broadcom Advanced Services Protocol (BASP) which includes NIC Teaming, as well as the Broadcom Advanced Control Suite (BACS) which is a Windows application which helps configure NIC Teaming.

There are similar tools available from Intel which provide essentially identical function but use different graphical tools for configuration. VMware provides a similar capability in their ESX product; details and examples of its configuration are shown in Chapter 8, “Load balancing with VMware” on page 115.

NIC Teaming allows two or more physical NICs to be treated as a single logical network object in Windows or a single /dev/eth file in Linux®. The single object or file can then be assigned network properties such as an IP address in the same way as any other NIC.

The BACS application allows several types of teams to be created. For HA designs, the Smart Load Balancing or Smart Load Balancing (no failback) type is used. Layer 2 designs can have both of the adapters (on an HS20 blade) as active members of the team; for Layer 3 designs, an active or standby team is used with one adapter as an active member of the team and the second adapter as a standby member of the team.

Note: *Smart Load Balancing* is a term used by Broadcom to refer to a feature that they provide which provides load balancing of network connections. It is NOT the same as Server Load Balancing, which balances the load on applications. Since both of these features are abbreviated SLB, be sure to be aware of which feature is being referred to based on the context.

This is the only section of this Redpaper where the Broadcom SLB is discussed. In the remainder of this Redpaper, SLB always means Server Load Balancing.

NIC Teaming is intended to provide both additional capacity (bandwidth) as well as High Availability. The team will detect loss of signal on any of its member NICs and continue to send traffic through other active members, or activate standby members if necessary. In the IBM BladeCenter, NIC Teaming will detect the failure of a NIC chip on the server blade, the loss of connection to a switch module via the midplane, and the failure of a switch module (including intentional removal or power-off). Of these, intentional removal or power-off of a switch module is by far the most common.

The BASP drivers also provide support for 802.1q tagging of the server NIC. This allows support for multiple VLANs on a single physical NIC or on a group of teamed NICs. When this capability is used, each VLAN has its own network object (windows) or /dev/eth file (Linux). Thus, each VLAN can be assigned its own IP address. This can be useful to isolate different categories of traffic from each other or to provide different Quality of Service (QoS) configurations for different types of traffic whose target is the same server.

Notes: Be aware of the following conditions.

- ▶ The BASP driver can be configured to use standards-based Port Aggregation (802.3-ad) teaming. This is useful on HS40 blades or HS20 blades with the SCSI sidecar, both of which have two ports connecting them to each switch module. Only ports connected to the same switch should be teamed in this way.
- ▶ The current production version of the GbESM software(20.2.2.x) does not support trunking on internal ports. The next software release will add this function.
- ▶ Some of the descriptions contain as is information based on a test in our specific environment with BASP 7.12.01, the latest version as we write this Redpaper. Your experiences and environments might differ, as might future software releases.

For more information about BASP NIC teaming, refer to the BACS online help and *BCM570X Broadcom NetXtreme Gigabit Ethernet Teaming* white paper, which is available at:

<http://www.broadcom.com/collateral/wp/570X-WP100-R.pdf>

5.7.3 Configuration of Trunk Failover

Trunk Failover is configured on the Nortel GbESM with the **failover enaldis** command, as follows:

```
/cfg/12/trunk <trunk instance number>  
failover ena
```

If Trunk Failover is used in concert with VRRP, then the following must also be part of the configuration:

```
/cfg/13/vrrp/trnkfo enabled
```

If there are multiple trunk groups which are critical upstream connections, such as to multiple upstream switches, then they should all have the failover feature enabled. Failover will not occur until all of them fail at the same time.

In most cases, you should configure Trunk Failover on all Nortel Networks L2/7 GbESM in the IBM BladeCenter if the server blades are running NIC Teaming. These two features work together to provide a High Availability design.

Restriction: The currently available release (20.2.2.7) of software for the Nortel Networks L2/7 GbESM does not support Trunk Failover for trunks configured with LACP. This feature is to be added in the next software release. This results in a slight change in the command syntax required. We were able to validate this briefly with an early test version of the next release of software. Additional granularity options for Trunk Failover will also be introduced in the forthcoming release.

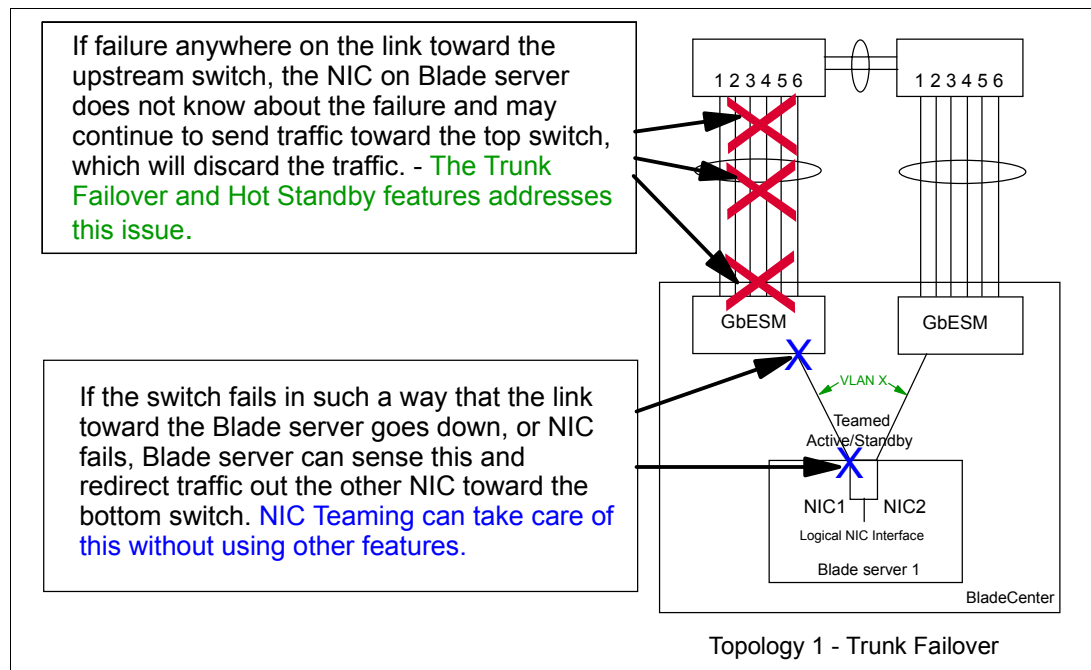


Figure 5-4 What Trunk Failover and Hot Standby can protect against

5.7.4 Configuration of Hot Standby

Configuration of Hot Standby is more complex than Trunk Failover. A complete example is shown in Example 8-3 on page 121. The configuration requires the following steps, which should be performed on both switches:

1. Globally enable Hot Standby, which is always used with VRRP:


```
/cfg/13/vrrp/hotstan enabled
```
2. Configure each internal port that you want to have disabled in the event of a failure. (By default, an internal port will not be disabled):


```
/cfg/slb/port INTx/hotstan ena
```
3. Configure an interswitch port on each switch which will be connected to the partner switch. This port must be in a VLAN used to carry no other traffic; VLAN 999 is used in this example:


```
/cfg/s1b/port EXT4/intersw ena
/cfg/12/vlan 999/ena
/cfg/12/vlan 999/def EXT4
```

4. Create an IP interface on the VLAN used in step 3.

```
/cfg/13/ip/if <instance>
  addr xxx.xxx.xxx.xxx
  mask xxx.xxx.xxx.xxx
  vlan 999
  ena
```

5. Create a VRRP instance for the Interface created in step 4.

```
/cfg/13/vrrp/vr <instance number>
  vrid <unique number for the link>
  prio 101 (or higher - on the primary switch only)
  if <instance>
  ena
```

6. Create a VRRP group instance.

```
/cfg/13/vrrp/group
  vrid <unique number>
  prio 101 (or higher - on primary switch only)
  if <instance>
  track/ports e
  ena
```

5.7.5 Introduction to VRRP

Virtual Router Redundancy Protocol (VRRP) is a Layer 3 protocol used to enable switches to back each other up in a way which is transparent to client and server computers. VRRP works by defining an address which is shared between the switches. One switch which is the **Master** is the only one which will answer to the shared address. One or more other switches in **Backup** state are configured to take over from the master in the event of a failure. An instance of VRRP is configured for each VLAN where a shared address is to be used. This implies that if there is one VLAN for the internal ports and an additional VLAN for the external ports, then there can be two instances of VRRP, providing a shared address on the internal VLAN and a different shared address on the external VLAN.

VRRP priority

Each switch in a group running VRRP has a configured priority. When VRRP first becomes active, the switch with the highest priority becomes the Master switch. The master switch sends out periodic hello packets announcing that it is still operational. The backup switch with the highest configured priority takes over when the hello packets are no longer received.

There are configuration options, called *tracking* options, which adjust the priority of a switch dynamically based on the number of certain categories of resources (such as ports) which are available. Use of these options can allow a backup switch to take over even if the current master is still running but has lost some of the tracked resources.

VRRP configuration

A full description of the configuration of VRRP is found in the *Application Guide* and the *Command Reference* for the L2/7 GbESM. A brief discussion of the key parameters follows.

Concepts

VRRP configuration encompasses the following terms:

- ▶ VRRP address

VRRP works by creating an address which is shared by the participating switches. (There are typically two switches, one *master* and one *standby*, but the standard allows as many as 64 with one master and the remainder standby.) This address can be the same as the address of the master switch on the VLAN, but it is recommended that this not be done. At any moment, only the switch that is master will answer to the VRRP address.

► VSR address

If the selected VRRP address is the same as the Virtual Address (VIP) of a Load Balance group, then it is considered a *Virtual Server Router* (VSR). In this case, the Load Balance group will be accessed only through the current master switch.

► Elections and priority

When a VRRP instance is initialized, the participating switches hold an election to determine which switch will be the master. The switch configured with the highest priority wins; if there are two switches of equal priority then the one with the higher MAC address will win.

► Instances

A set of switches can have multiple VRRP instances, one or more on each of one or more VLANs that they share. Each instance must have its own VRRP address, and its own identifying number which is unique within a VLAN. This number is set on the VRID parameter in the GbESM configuration. This is not the same as the VR instance number which is unique within the particular GbESM switch. Instances are configured to be associated with IP interfaces rather than directly with VLANs.

► Tracking

A VRRP instance is configured with a base priority but can have its priority adjusted upwards based on the status of other resources within the switch. The priority can thus change during normal operation. By default, a switch gains higher priority than the current master switch will preempt that switch and become the master. Trackable resources on the GbESM include ports, real servers, and others.

Minimal configuration example

Example 5-4 shows the configuration of a single VRRP instance. It includes the specification of a nondefault base priority and enabling port tracking to adjust that priority dynamically.

Example 5-4 VRRP instance configuration

```
/* globally enable VRRP */
/cfg/13/vrrp/on
/* create one instance */
/cfg/13/vrrp/vr 1 /* 1 is the arbitrary number of the instance on this switch */
  vrid 12 /* this number must be unique on the VLAN associated with the if below */
  if 12 /* refers to an IP instance; multiple VRID's can be on the same if */
  prio 101 /* default is 100 */
  ena /* enable this VRRP instance */
  track
    ports e /* enable port tracking */
```

5.7.6 Some important rules for ensuring High Availability

For High Availability (HA) to be truly effective, it needs to be well thought out. A complete High Availability design should encompass servers, storage, and more of the network than just the portions connected to the BladeCenter chassis. The object is to ensure that there is no single point of failure which can cause the application(s) to become unavailable or unreachable.

The following are some important design considerations to try to ensure connectivity is maintained under various failure scenarios:

- ▶ For NIC teaming to work properly with Trunk Failover, you must have external Layer 2 connectivity between the GbESMs. This can be done by cabling the GbESM modules directly to each other or by connecting them both to the same collection of upstream switches.
- ▶ VRRP also requires a Layer 2 connection between switches. This connection must carry all the VLANs which have a VRRP instance configured.
- ▶ To provide robust HA in a Layer 3 design:
 - The two Nortel GbESMs should be configured with VRRP.
 - The blade servers need to be using the VRRP address(es) for the VLANs where they are configured as their default gateway.
 - It is possible to use VRRP (or equivalent) on the upstream switches as well to provide an even more robust HA design.

Note that the failure of a NIC within the blade server, the failure of a link between the GbESM and the blade server, and the hard failure of a GbESM would all result in a link down condition and would be successfully detected by NIC Teaming without the use of Trunk Failover.

5.8 Additional functions

The functions described in this section are not required to implement SLB, but are higher level (4-7) functions of the L2/7 GbESM. Complete details of their configuration can be found in the GbESM *Application Guide* and *Command Reference*.

5.8.1 Filters

Filters are used to allow and deny traffic selectively. Multiple filters can be applied to particular ports. They are processed in numerical order (see below). When a particular filter is matched, the configured action is taken and higher numbered filters are not processed. Filters can be created to match source and destination IP addresses, source and destination MAC addresses, and additional IP header fields. Layer 7 filters can be created to match data in the payload, rather than in any of the TCP or IP headers.

Minimal configuration example

Example 5-5 on page 71 allows Telnet (TCP port 23) traffic from to a particular destination address (10.10.0.1) to originate only from two subnets (10.1.0.0 and 10.2.0.0); any other Telnet traffic to the destination will be dropped. Filters 10, 20, and 30 will execute in order: 10 and 20 each allow one subnet, and 30 disallows all other telnet traffic to the configured destination. All other traffic is allowed by default.

The filter will be applied to all EXternal ports on the switch.

Example 5-5 Telnet filter for EXternal switch ports

```
/c/slb/filt 10
  ena
  action allow
  sip 10.1.0.0
  smask 255.255.255.0
  dip 10.10.0.1
  dmask 255.255.255.255
```

```

        proto tcp
        dport telnet
        vlan any
/c/slb/filt 20
    ena
    action allow
    sip 10.2.0.0
    smask 255.255.255.0
    dip 10.10.0.1
    dmask 255.255.255.255
    proto tcp
    dport telnet
    vlan any
/c/slb/filt 30
    ena
    action deny
    dip 10.10.0.1
    dmask 255.255.255.255
    proto tcp
    dport telnet
    vlan any

```

To apply these filters to the external ports, you must configure each of these for each port:

```

/c/slb/port EXT<x>
    filt ena
    add 10
    add 20
    add 30

```

5.8.2 Network Address Translation

Network Address Translation (NAT) is a process performed on packets to allow the use of an alias in place of the actual address of the source or destination of the packet. NAT is configured through the use of filters with an action of NAT. The primary function of NAT is to allow servers' true addresses to be hidden from the public Internet (or from a corporate network) to provide additional security.

NAT can be implemented on a one-to-one basis, where there are equal numbers of public and private (or inside and outside) addresses in use. It can also be implemented on a many-to-one basis, where many private or inside addresses share the use of one public or outside address. Both of these modes are available on the GbESM.

Further discussion and examples of NAT can be found in the *Application Guide* and *Command Reference*.

5.8.3 SYN attack mitigation

A *SYN attack* is a form of Denial of Service (DoS) attack where an attacker will start to open large numbers of sessions, hundreds or thousands per second, with their target and leave these sessions in a partially open state. All the open sessions typically overflow the session table on the target server and render it incapable of opening any useful sessions.

The GbESM can protect servers from attacks of this nature in the following ways:

- ▶ By interposing itself between the attacker and the target server, newly created sessions can be on the GbESM rather than the servers. When a session is fully opened, the

GbESM opens a session to the target server and splices the server session and the client session together. Thus, the half-open sessions never touch a server.

- ▶ The GbESM has a larger session table than the typical server and its firmware includes processes to clean half-open sessions out of the table.
- ▶ The GbESM can be configured to detect SYN attacks based on the number of half-open sessions from any one source in a specified amount of time (a small number of seconds).

SYN attack mitigation is always provided whenever delayed binding is in use with load balancing. It can also be configured exclusive of load balancing when needed. The parameters to adjust the time intervals discussed above can be found in the **/cfg/s1b/adv/syntak** submenu. The defaults are:

- ▶ Check for an attack every 2 seconds.
- ▶ An *attack* is considered to exist when more than 10,000 half-open sessions are created per second.



Load balancing with WebSphere Portal

This chapter discusses WebSphere Portal load-balancing capabilities and gives examples for HTTP and HTTPS.

6.1 Introduction to WebSphere Portal

A *portal* offers a single point of personalized, unified access to applications, content, processes and people. A portal has the following advantages:

- ▶ Delivers integrated content and applications, plus offers a unified, collaborative workplace.
- ▶ Provides other valuable functions such as security, search, and workflow.
- ▶ Is an open, standards-based framework supporting a wide array of options across databases, directories, platforms and security.

Portals are the next-generation desktop, delivering e-business applications over the Web to many different client devices. Portals are designed to meet the needs of all enterprises from small and medium businesses to the largest enterprises that demand the most scalable, secure, and robust infrastructure.

“A complete portal solution should provide users with convenient access to users with everything they need to get their tasks done anytime, anywhere, in a secured manner.”
Stefen Liesche, IBM Portal Architect

A consistent, integrated user experience is achieved by portals that do not only aggregate components into a single view, but, in addition, allow integration of these components within context. This is often called *integration on the glass*, as all the applications are integrated in context by the portal into one single screen that displays on the monitor of the portal end user. This is a very powerful concept that, in today's world of widely fractured IT infrastructures, allows the delivery of consistent and integrated views on multiple IT services. Integration on the glass improves the user experience and productivity of the IT user. Instead of dealing with different IT systems with potential different user interfaces, integration on the glass provides a single, consistent view.

In addition to contextual integration capabilities, portals can provide rich programming frameworks for building user interfaces for component-oriented applications in service oriented architectures. Service Oriented Architecture (SOA) is an approach for building distributed systems that deliver application functionality as services to either end-user applications or other services. SOA provides means to integrate and manage these different services, refer to this link for more:

<http://www.ibm.com/SOA>

Portals provide first-class UI support in Service-Oriented Architectures. *Portlets*, their basic building block, let developers focus on unique aspects of their application, while the middleware handles common functions for life cycle, per-user customization, aggregation, and integration with other components. In addition portals might provide valuable service functions such as security, search, collaboration, and workflow. Portals provide the ability to aggregate and integrate the UI in a similar way SOA runtimes can combine and integrate services. Component UIs are aggregated into larger, higher value UIs, giving users a single view of IT services with a single UI to master. Applications originally designed separately can be integrated (aggregation and context) together to enable new function. The portal model allows for improved agility for on-demand businesses. Portal administrators become application integrators who create new applications for their users without programming: by defining new pages, adding portlets to them, connecting the portlets together in context, and setting entitlements. With portal technologies end users can become their own application assemblers by customizing their portal based workspaces.

There are many reasons why a portal would benefit your organization. For example:

- ▶ Control information glut
- ▶ Improve cycle times
- ▶ Empowering knowledge workers
- ▶ Reduce IT complexity
- ▶ Enhance partner and supplier communication
- ▶ Streamline™ processes

WebSphere Portal supports multiple industry portals and various communities within a company. Portal consists of four basic services: Framework, Integration, Content, and Collaboration.

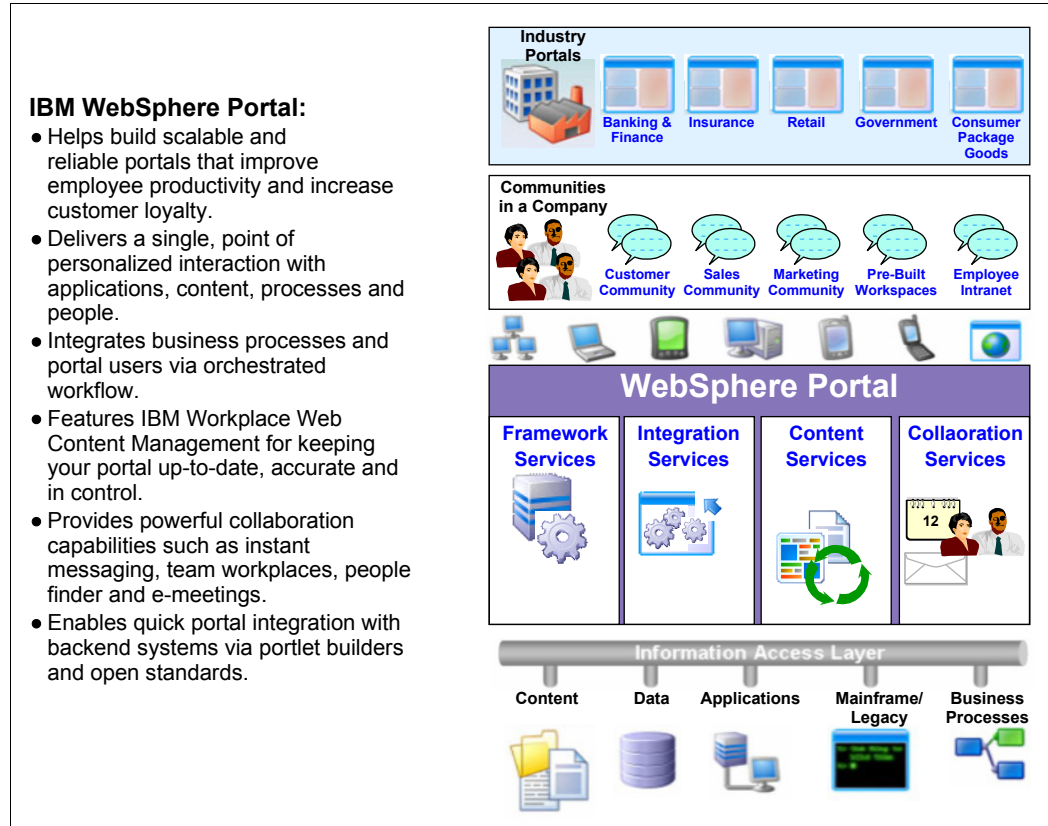


Figure 6-1 IBM WebSphere Portal Framework

WebSphere and the WebSphere Portal constitute a three-tier architecture (Figure 6-1) designed to support very scalable applications accessed over the Internet and with a Web browser. The tiers are a Web Server tier, which manages connections to client systems; an Application Server tier, where most processing takes place; and a Database Server tier, where data is kept and managed. Both the Web and Application Server tiers can grow by scaling out, adding additional servers rather than increasing the power of existing servers.

The remainder of this discussion focuses on the Web and Application Server tiers.

6.2 Value of load balancing with WebSphere Portal

Because the Web and Application tiers of a WebSphere configuration scale out, they are both candidates for load balancing. WebSphere documentation calls for an IP sprayer to be deployed in front of the Web Servers to allocate incoming client request to the various servers. An “IP Sprayer” is, of course, a load balancing device, and the L2-7 GbESM qualifies

for that role. The configurations shown in 6.3.1, “Configuration examples” on page 79 show how to use the GbESM to provide load balancing of requests to multiple Web Server instances.

One of the functions of the Web Servers, with a plug-in component, is to allocate requests to the Application Servers. Each Application Server is registered in the plug-in’s configuration file. Typically, requests are allocated to the Application Servers using simple round-robin load balancing. We attempted to implement load balancing of the requests forwarded by the Web Servers to the Application Servers using the GbESM. We were unable to make this configuration stable due to the structure of the cookies used to ensure persistence - to keep a client attached to the same Application Server for the duration of their interaction.

We were able to implement a configuration which used the GbESM to take on many of the functions of the Web Server tier, and directly forward client requests (from Web browsers) to the Application Servers. This configuration is shown in 6.3.1, “Configuration examples” on page 79. The decision to implement this configuration has both benefits and detriments, both of which are discussed in 6.3, “Implementing load balancing with WebSphere Portal” on page 78.

6.3 Implementing load balancing with WebSphere Portal

This section illustrates two different topologies through which WebSphere Portal can be implemented and provided with needed load balancing services. The first topology is the orthodox WebSphere three-tier design with Web, Application, and Database servers; the GbESM provides load balancing for the Web Servers. The second topology uses the capabilities of the GbESM to take on some of the functions of the Web Servers and load balances client requests directly to the Application server tier.

Decision criteria: choosing between two topologies

The benefits of using the topology with both Web and Application servers include:

- ▶ Static content can be provided with the Web Servers without taking up resources (memory, CPU, disk) on the Application Servers, which can be dedicated to processing the appropriate application logic to provide dynamic content.
- ▶ The Web Servers can be positioned in a different security zone than the Application Servers, typically by placing a firewall between the two tiers. This can provide additional protection for the Application Servers.
- ▶ Multiple clusters of Web Servers can be provisioned in different geographic locations and still feed transactions to the same Application Servers. This will potentially allow a fast perceived response to the end user regardless of geography and conserve bandwidth where it is most beneficial to do so (e.g. across oceans). Note that the next software release of the GbESM will support *Global Load Balancing*, a capability which will enable end users in all geographies to use the same URL and, yet, be directed to the closest cluster of Web Servers.

The benefits of using the topology where the GbESM balances client requests directly to the Application Servers include:

- ▶ The GbESM supports more flexible and versatile load balancing algorithms than the WebSphere plug-in. These include sending the next request to the server with the smallest number of existing connections or to the server which has the fastest response time to a health check. The various load balancing capabilities are discussed in Chapter 5, “Introduction to server load balancing” on page 47.

- ▶ Health checks of all servers are constantly performed by the GbESM. Servers that fail the health check are automatically removed from the load balancing pool. This is designed to prevent client requests from being sent to a malfunctioning server.
- ▶ The GbESM provides security functions including the ability for clients to reach a server through a virtual address, shielding knowledge of the real address of the server from the clients. The GbESM can also function as a packet filter, blocking inappropriate or undesired access to the servers based on destination port, client address, and other criteria. The GbESM can also be provisioned in a different security zone than the servers themselves and firewalls can be provisioned on server blades within the same Blade Center chassis as the Application Servers if desired.
- ▶ Using this approach allows all the available servers to potentially be deployed as Application Servers, thus further spreading the load.

6.3.1 Configuration examples

This section show three configuration examples. GbESM configuration files and WebSphere configuration information is included. The examples are:

- ▶ Using the GbESM to load balance three Web Servers using http. A similar configuration could be developed using https and Web Servers configured as needed.
- ▶ Using the GbESM to load balance directly to three Application servers using http. Both port 80 and an alias port configured into WebSphere are supported (9081).
- ▶ Using the GbESM to load balance directly to three Application servers using https. Both port 443 and an alias port configured into WebSphere are supported (9444).

The following IP addresses applies to all three examples. All DNS names are qualified under the subdomain of *.ITSO.RAL.IBM.COM.

- ▶ Web Servers are:
 - BC2SRV1 - 9.42.171.85
 - BC2SRV2 - 9.42.171.86
 - BC2SRV7 - 9.42.171.63
- ▶ Application Servers are:
 - BC2SRV5 - 9.42.171.45
 - BC2SRV6 - 9.42.171.57
 - BC2SRV8 - 9.42.171.243
- ▶ Virtual addresses are:
 - RUFUS3 - 9.42.171.252
 - RUFUS2 - 9.42.171.248

The configuration is shown in Figure 6-2 and Figure 6-3 on page 80.

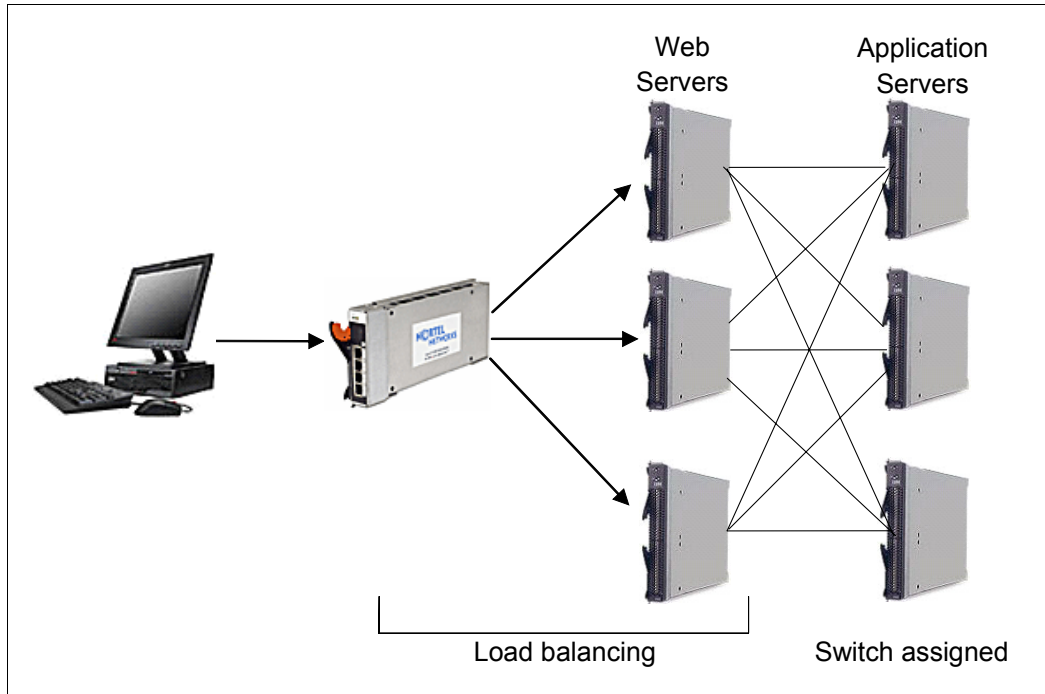


Figure 6-2 3-tier WebSphere environment

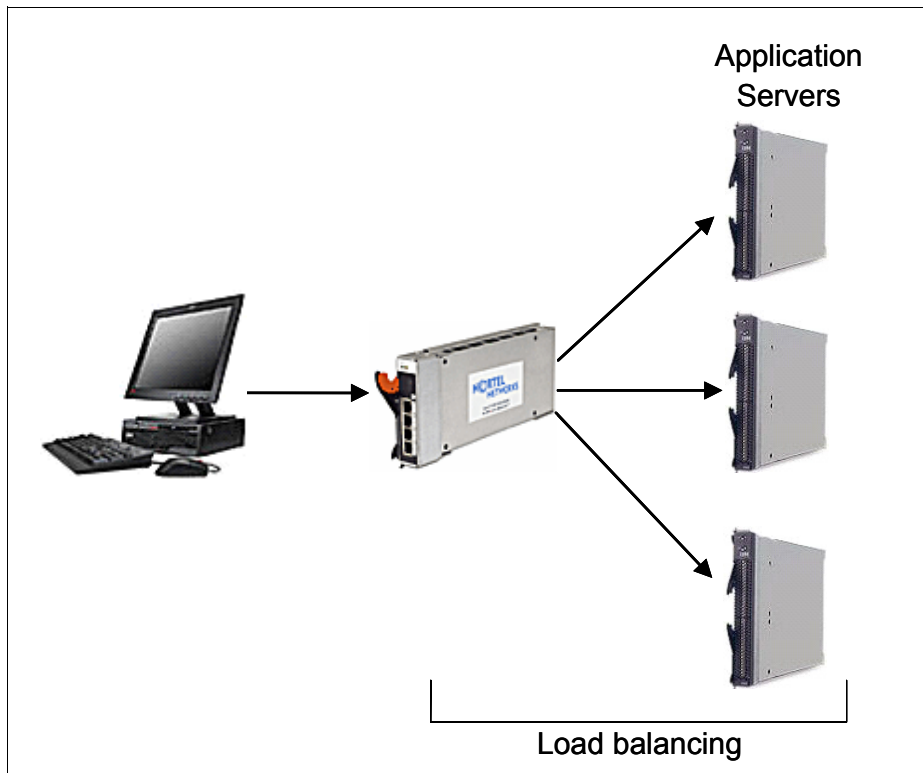


Figure 6-3 2-tier WebSphere environment

Load balancing WebSphere Web Servers (http)

The following example shows standard load balancing on a group of three WebSphere Web Servers.

In Example 6-1, it is not important that the client be repeatedly directed to the same Web Server. The Web Server plug-in will forward the request to the appropriate Application Server and ensure persistence on the Application Server. The Web Servers use cookies to select the appropriate Application Server by using the clone id to identify the appropriate servers. These cookies are generated by the Application Server, not by the Web Server.

As a result of this design, none of the persistence options of the GbESM switch are used in this example. We attempted to use the GbESM to provide load balancing of requests between the Web and Application servers but were unable to make such a configuration stable. The example does include configuration for the use of VRRP and Trunk Failover to achieve high availability.

Note: Ports 1, 2, and 7, which hold the Web Servers, are configured with both client and server processing. This is from tests of load balancing between the Web and Application servers using the GbESM, and is not required for this configuration. Only server processing is needed.

Example 6-1 Load balancing WebSphere Web Servers (http)

```
script start "Nortel Networks Layer2-7 GbE Switch Module" 4 /**** DO NOT EDIT THIS LINE!
/* Configuration dump taken 0:34:54 Fri Jan 2, 2070
/* Version 20.2.2.6, Base MAC address 00:0f:06:eb:58:00
/c/12/vlan 1
    def INT1 INT2 INT3 INT4 INT5 INT6 INT7 INT8 INT9 INT10 INT11 INT12 INT13 INT14 EXT1 EXT2
EXT3 EXT4
/c/12/stg 1/off
/c/12/stg 1/clear
/c/12/stg 1/add 1
/c/12/trunk 1
    ena
    failovr ena
    add EXT2
/c/13/if 1
    ena
    addr 9.42.171.247
    mask 255.255.255.0
    broad 9.42.171.255
/c/13/gw 1
    ena
    addr 9.42.171.3
/c/13/vrrp/on
/c/13/vrrp/trnkfo enabled
/c/13/vrrp/vr 1
    ena
    vrid 10
    if 1
    addr 9.42.171.252
    track
        ports e
/c/slb/adv
    direct ena
    grace ena
/c/slb/sync
    prios d
    reals e
    state e
/c/slb/sync/peer 1
    ena
    addr 9.42.171.21
```

```

/c/slb/real 1
  ena
  rip 9.42.171.63
/c/slb/real 2
  ena
  rip 9.42.171.85
/c/slb/real 3
  ena
  rip 9.42.171.86
/c/slb/real 5
  ena
  rip 9.42.171.45
/c/slb/real 6
  ena
  rip 9.42.171.57
/c/slb/real 8
  ena
  rip 9.42.171.243
/c/slb/group 1
/* WebSphere Web Servers
  metric minmisses
  add 1
  add 2
  add 3
/c/slb/group 2
/* Web Sphere App Servers
  health http
  add 5
  add 6
  add 8
/c/slb/port INT1
  client ena
  server ena
/c/slb/port INT2
  client ena
  server ena
/c/slb/port INT5
  server ena
/c/slb/port INT6
  server ena
/c/slb/port INT7
  client ena
  server ena
/c/slb/port INT8
  server ena
/c/slb/port EXT2
  client ena
/c/slb/virt 1
  ena
  vip 9.42.171.252
/c/slb/virt 1/service http
  group 1
/
script end /**** DO NOT EDIT THIS LINE!

```

Load balancing WebSphere Web Servers: https

This example shows a configuration for use where the connection between the client browser and the Web Servers is intended to use https. In addition to the GbESM configuration, the

IBM HTTP Server, which is the core of the WebSphere Web Server, must be configured to accept HTTPS. This requires the following:

- ▶ Configuring the server to listen on port 443 and expect encrypted traffic.
- ▶ Creating appropriate SSL keys, which is done using the **ikkeyman** utility.

These steps are summarized here.

Example 6-2 is a portion of the contents of the httpd.conf file used by the IBM HTTP Server portion of the WebSphere Web server. The IHS server is based on the Apache Web server. The portions of the file changed to allow it to accept encrypted https traffic are shown in bold; much of the remainder of the file has been removed for brevity's sake.

Example 6-2 Sample httpd.conf file for https support

```
ServerName bc2srv7
# This is the main server configuration file. See URL http://www.apache.org/
# for instructions.

# Do NOT simply read the instructions in here without understanding
# what they do, if you are unsure consult the online docs. You have been
# warned.

# Originally by Rob McCool

# Note: Where filenames are specified, you must use forward slashes
# instead of backslashes. e.g. "c:/apache" instead of "c:\apache". If
# the drive letter is omitted, the drive where Apache.exe is located
# will be assumed

# this is a True Config File
# see http://www.apache.org/info/three-config-files.html

# ResourceConfig /dev/null
# AccessConfig /dev/null

# ServerType must be standalone.

ServerType standalone

# ServerRoot: The directory the server's config, error, and log files
# are kept in

ServerRoot "C:\IBM HTTP Server"

#
# The following lists extra modules that can be uncommented to be loaded
# to enable extra functionality. See the manual
# (http://www.apache.org/docs/mod/) for details on the functionality
# of each module.
#
# The stanza below loads the IBM support for 128 bit SSL
LoadModule ibm_ssl_module modules/IBMModuleSSL128.d11
LoadModule ibm_app_server_http_module
"C:\WebSphere\AppServer/bin/mod_ibm_app_server_http.d11"

# Port: The port the standalone listens to.
Port 80
```

```

# Timeout: The number of seconds before receives and sends time out

Timeout 300

# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.

KeepAlive On

# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
# We reccomend you leave this number high, for maximum performance.

MaxKeepAliveRequests 100

# KeepAliveTimeout: Number of seconds to wait for the next request

KeepAliveTimeout 15

# -----
# This section defines server settings which affect which types of services
# are allowed, and in what circumstances.

# Each directory to which Apache has access, can be configured with respect
# to which services and features are allowed and/or disabled in that
# directory (and its subdirectories).

# Note: Where filenames are specified, you must use forward slashes
# instead of backslashes. e.g. "c:/apache" instead of "c:\apache". If
# the drive letter is omitted, the drive where Apache.exe is located
# will be assumed

# First, we configure the "default" to be a very restrictive set of
# permissions.

# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.

# This should be changed to whatever you set DocumentRoot to.

<Directory "C:\IBM HTTP Server\htdocs/en_US">

# This may also be "None", "All", or any combination of "Indexes",
# "Includes", "ExecCGI", or "MultiViews".

# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.

Options Indexes

# This controls which options the .htaccess files in directories can
# override. Can also be "All", or any combination of "Options", "FileInfo",
# "AuthConfig", and "Limit"

AllowOverride None

# Controls who can get stuff from this server.

```



```

order allow,deny
allow from all

</Directory>

# C:\IBM HTTP Server/cgi-bin should be changed to whatever your ScriptAliased
# CGI directory exists, if you have that configured.

<Directory "C:\IBM HTTP Server/cgi-bin">
AllowOverride None
Options None
</Directory>
#
# The below enables traffic on port 443. Note that the expected URL includes the FQDN
# of the virtual server address (rufus3). The VirtualHost section below uses the REAL
# IP address of THIS web server and would be different on the other web servers in the
# load balanced group. The directives after the </VirtualHost> disable SSL for all ports
# other than 443.
#
Listen 443
<VirtualHost 9.42.171.63:443>
ServerName rufus3.itso.ral.ibm.com
SSLEnable
SSLClientAuth none
</VirtualHost>
SSLDisable
SSLV2Timeout 100
SSLV3Timeout 1000
KeyFile "C:/IBM HTTP Server/ssl/ikeyman/en_US/key.kdb"
#
#The below identifies the configuration file for the WebSphere plugin, which sends
# requests to Application Servers where appropriate (i.e. not static content).
WebSpherePluginConfig "C:\WebSphere\AppServer/config/cells/plugin-cfg.xml"

```

In Figure 6-4, Figure 6-5 on page 86, and Figure 6-6 on page 87, the `ikeyman` utility is shown. Using this program to generate the key file referred to in the example is required for SSL to function properly.

The key file to be created is of type CMS key database file as shown in Example 6-2 on page 83. The file name must match the name shown in the `httpd.conf` file as shown in Example 6-2. The certificate created should match the fully qualified domain name (FQDN) which resolved to the virtual address of the load balancing pool. This means that there is only one certificate required for each pool, instead of one for each real server, which can result in cost savings when a public Certificate Authority is used to provide the certificates.

Note that the example shown (is of a self-signed certificate, which is sufficient for corporate intranet use if the company's security policies allow this. In the case of a certificate obtained from a public (or corporate) CA, it would be imported, not created, by the `ikeyman` utility.

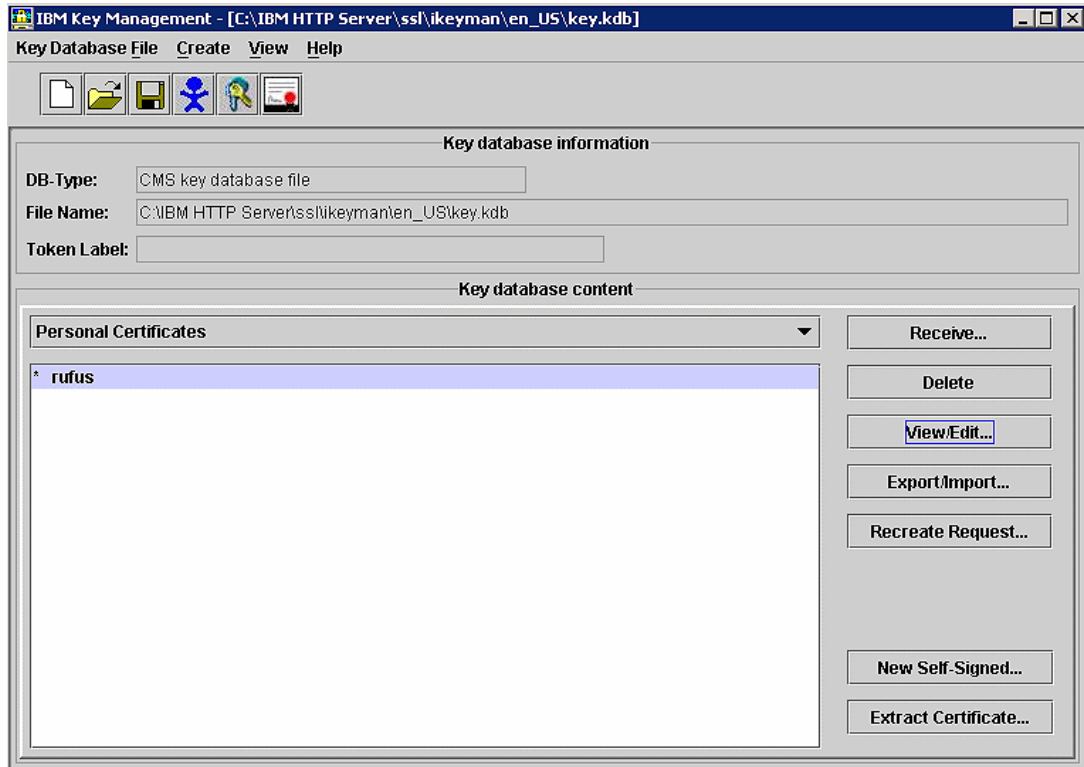


Figure 6-4 IBM Key Management utility

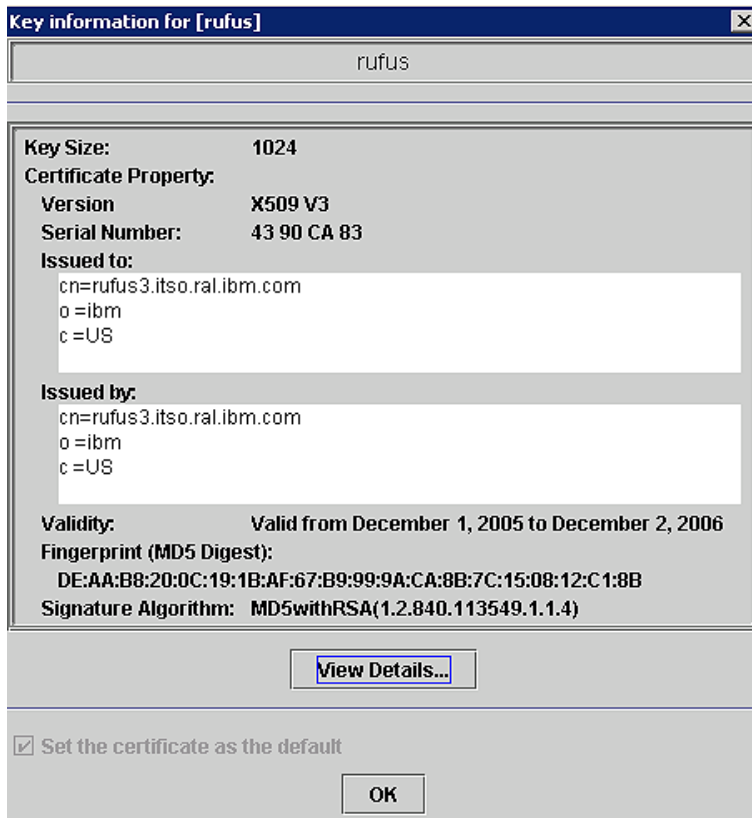


Figure 6-5 Key information window

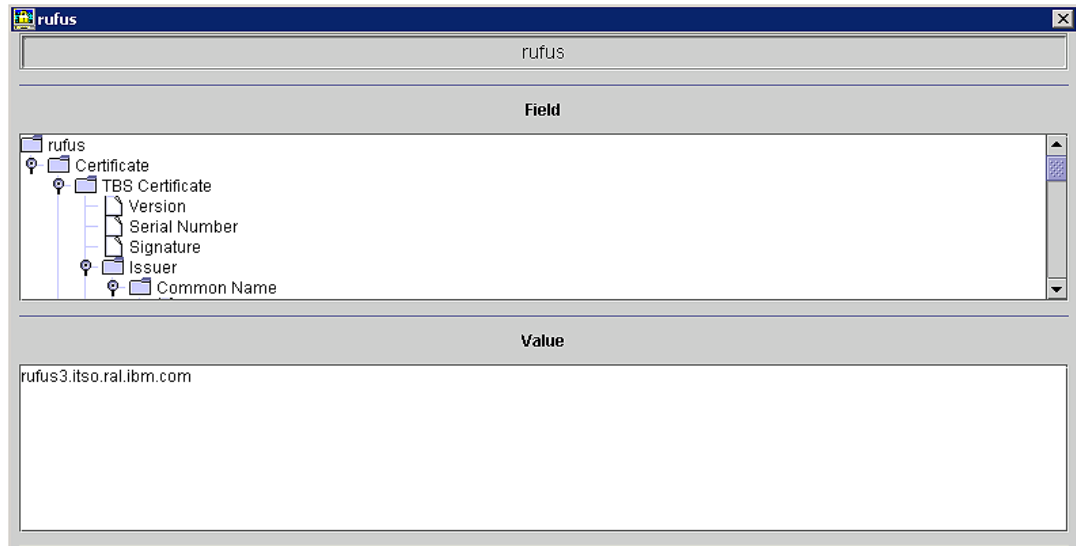


Figure 6-6 Subject key information

The GbESM configuration required to support https connectivity to the Web Servers is shown in Example 6-3. In this configuration, sslid persistence is used to ensure that repeat connections which are part of the same conversation are directed to the same Web Server. However, it is *not* required that this be done, because the encrypted cookies will be used to ensure that the proper Application Server is used. Rather, this avoids the workload required to generate a new SSL session ID each time the enter key is pressed.

Timers (tmout and slowage) are also set in this configuration to ensure that the sslid persistence will survive relatively long idle times between connections. This also is not required but is included along with the use of sslid for illustrative purposes. A discussion of the use of these parameters is included in the section on Application Servers in Example 6-3.

Example 6-3 Load balancing WebSphere Web Servers (https)

```
script start "Nortel Networks Layer2-7 GbE Switch Module" 4 /**** DO NOT EDIT THIS LINE!
/* Configuration dump taken 0:34:54 Fri Jan 2, 2070
/* Version 20.2.2.6, Base MAC address 00:0f:06:eb:58:00
/c/12/vlan 1
    def INT1 INT2 INT3 INT4 INT5 INT6 INT7 INT8 INT9 INT10 INT11 INT12 INT13 INT14 EXT1 EXT2
EXT3 EXT4
/c/12/stg 1/off
/c/12/stg 1/clear
/c/12/stg 1/add 1
/c/12/trunk 1
    ena
    failovr ena
    add EXT2
/c/13/if 1
    ena
    addr 9.42.171.247
    mask 255.255.255.0
    broad 9.42.171.255
/c/13/gw 1
    ena
    addr 9.42.171.3
/c/13/vrrp/on
/c/13/vrrp/trnkfo enabled
/c/13/vrrp/vr 1
```

```

ena
vrid 10
if 1
addr 9.42.171.252
track
    ports e
/c/slb/adv
    direct ena
    grace ena
/c/slb/sync
    prios d
    reals e
    state e
/c/slb/sync/peer 1
    ena
    addr 9.42.171.21
/c/slb/real 1
    ena
    rip 9.42.171.63
/c/slb/real 2
    dis
    rip 9.42.171.85
/c/slb/real 3
    ena
    rip 9.42.171.86
/c/slb/real 5
    ena
    rip 9.42.171.45
/c/slb/real 6
    ena
    rip 9.42.171.57
/c/slb/real 8
    ena
    rip 9.42.171.243
/c/slb/group 1
    metric minmisses
    add 1
    add 2
    add 3
/c/slb/port INT1
    client ena
    server ena
/c/slb/port INT2
    client ena
    server ena
/c/slb/port INT5
    server ena
/c/slb/port INT6
    server ena
/c/slb/port INT7
    client ena
    server ena
/c/slb/port INT8
    server ena
/c/slb/port INT9
    server ena
/c/slb/port INT10
    server ena
c/slb/port EXT1
    client ena

```

```
server ena
/c/slb/port EXT2
client ena
/c/slb/virt 1
ena
vip 9.42.171.252
/c/slb/virt 1/service http
group 1
/c/slb/virt 1/service https
group 1
dbind ena
/c/slb/virt 1/service 443/pbind sslid
/
script end /**** DO NOT EDIT THIS LINE!
```

Load balancing WebSphere Application Servers: http

This configuration uses the GbESM to provide load balancing directly to the Application Servers. It uses the same cookie, the JSESSIONID cookie, that the Web Server plug-in does to ensure persistence to the initially assigned Application Server. This cookie contains two fields:

- ▶ A *clone id*, which is unique to each application server
- ▶ A *session id*, which is unique to each client session

The clone id is used to ensure persistence. It is also possible, and we did successfully test this in our residency, using the session id. However, there are far more unique session IDs than clone IDs, and this would take up more memory on the switch. Using the clone ID is sufficient to ensure persistence and will scale better than using the session ID.

Because the servers are generating appropriate unique cookies, the switch is configured in passive cookie mode. Other modes are available for applications where the server does not generate cookies (cookie insert mode) or where the cookies are not unique (cookie rewrite mode).

In order to use cookie persistence, this example uses delayed binding. Cookie persistence is a Layer 7 capability, in that it looks at the cookie, which is part of the network payload and of the Application layer to make switching decisions. It is part of the http standard and not of any lower-layer standard.

The format of the WebSphere JSESSIONID cookie is as follows (Figure 6-7 on page 90):

- ▶ Session ID - 27 bytes long
- ▶ Colon (:) - delimiter
- ▶ Clone ID - 9 bytes long.

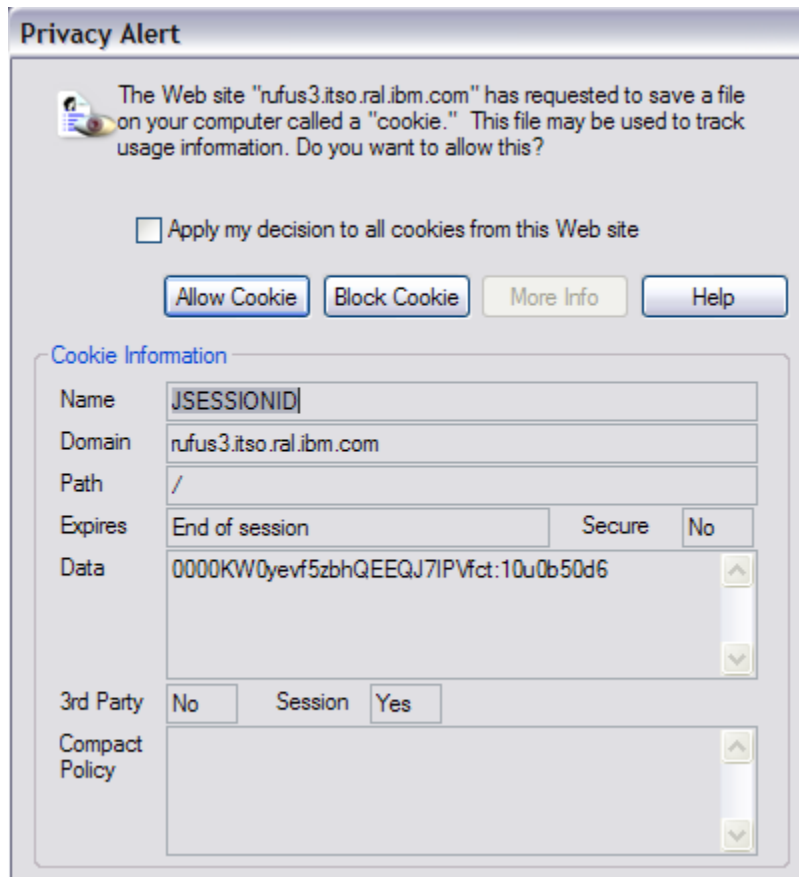


Figure 6-7 WebSphere JSESSIONID cookie

The configuration in Example 6-4 uses the Clone ID, which is also shown in the plugin configuration file in Example 6-6 on page 97 to ensure persistence.

Note: The service is defined to accept http on port 9081 as well as port 80. This is a result of the default configuration of the Application Servers and the Web Server plugin to communicate using http on port 9081. The configuration file for the plugin is shown in Example 6-6. It may be possible to customize the application servers to expect traffic on port 80 but our approach was to use port mapping on the GbESM to accomplish the same thing.

Example 6-4 Load balancing WebSphere Application Servers (http)

```
script start "Nortel Networks Layer2-7 GbE Switch Module" 4 /**** DO NOT EDIT THIS LINE!
/* Configuration dump taken 0:34:54 Fri Jan 2, 2070
/* Version 20.2.2.6, Base MAC address 00:0f:06:eb:58:00
/c/12/vlan 1
    def INT1 INT2 INT3 INT4 INT5 INT6 INT7 INT8 INT9 INT10 INT11 INT12 INT13 INT14 EXT1 EXT2
EXT3 EXT4
/c/12/stg 1/off
/c/12/stg 1/clear
/c/12/stg 1/add 1
/c/12/trunk 1
    ena
    failovr ena
    add EXT2
```

```

/c/13/if 1
  ena
  addr 9.42.171.247
  mask 255.255.255.0
  broad 9.42.171.255
/c/13/gw 1
  ena
  addr 9.42.171.3
/c/13/vrrp/on
/c/13/vrrp/trnkfo enabled
/c/13/vrrp/vr 1
  ena
  vrid 10
  if 1
  addr 9.42.171.252
  track
    ports e
/c/13/vrrp/vr 2
  ena
  vrid 2
  if 1
  addr 9.42.171.248
  track
    ports e
/c/slb/adv
  direct ena
  grace ena
/c/slb/sync
  prios d
  reals e
  state e
/c/slb/sync/peer 1
  ena
  addr 9.42.171.21
/c/slb/real 1
  ena
  rip 9.42.171.63
/c/slb/real 2
  ena
  rip 9.42.171.85
/c/slb/real 3
  ena
  rip 9.42.171.86
/c/slb/real 5
  ena
  rip 9.42.171.45
/c/slb/real 6
  ena
  rip 9.42.171.57
/c/slb/real 8
  ena
  rip 9.42.171.243
/c/slb/group 1
/* Web Servers
  add 1
  add 2
  add 3
/c/slb/group 2
/* App Servers
  health http

```

```

    add 5
    add 6
    add 8
/c/slb/port INT1
    client ena
    server ena
/c/slb/port INT2
    client ena
    server ena
/c/slb/port INT5
    server ena
/c/slb/port INT6
    server ena
/c/slb/port INT7
    client ena
    server ena
/c/slb/port INT8
    server ena
/c/slb/port INT9
    server ena
/c/slb/port INT10
    server ena
/c/slb/port INT12
    server ena
/c/slb/port EXT1
    client ena
    server ena
/c/slb/port EXT2
    client ena
/c/slb/virt 1
    ena
    vip 9.42.171.252
/c/slb/virt 1/service 9081
    group 2
    dbind ena
/c/slb/virt 1/service http
    group 2
    dbind ena
    rport 9081
/c/slb/virt 1/service 9081/pbind cookie passive JSESSIONID 29 9 disable
/c/slb/virt 1/service 1/rcount 1
/c/slb/virt 1/service 80/pbind cookie passive JSESSIONID 29 9 disable
/c/slb/virt 1/service 80/rcount 1
/
script end /**** DO NOT EDIT THIS LINE!

```

The file shown in Example 6-5 on page 93 is the plugin file generated by the installation of WebSphere Portal. Several key lines in the file are shown in **bold** and are discussed below:

- ▶ The **VirtualHost Name** entries identify which incoming connections will be managed by the plugin, by host header and port. In Example 6-5 on page 93, any host header will be accepted on the ports indicated. Some of the ports indicated are for the Web Servers themselves (80, 443). Some are passed through to the Application Servers. Some are for administrative tools rather than for the portal proper.
- ▶ **Server CloneID** defines a member of the portal cluster and specifies its unique clone ID. This ID will be used in cookies to identify the Application Server to be used in a connection. If an Application Server receives a request with a different clone ID than its own, it will attempt to redirect it.

- ▶ The highlighted Transport stanza identifies the fully qualified name of the Application Server and the ports and protocols it expects to receive. The Application Servers in this portal cluster expect http on port 9081 and https on port 9444. Note that changing this file will not change the expected ports on the Application Servers since the file itself is generated as part of the installation process.
- ▶ The other Transport stanzas shown (which use ports 9091, 9044, 9080, 9443) are for various administrative tools which can be reached with a Web browser. We did not load balance these in our testing because we believed that anyone using them would want to access a specific server rather than an arbitrarily chosen member of a load balance group.
- ▶ There are also specifications for the non-portal versions of the WebSphere Application Server; we did not use them in our testing.

Example 6-5 WebSphere Plugin Configuration File

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<Config ASDisableNagle="false" AcceptAllContent="false"
AppServerPortPreference="HostHeader" ChunkedResponse="false" IISDisableNagle="false"
IISPluginPriority="High" IgnoreDNSFailures="false" RefreshInterval="60"
ResponseChunkSize="64" VHostMatchingCompat="false">
  <Log LogLevel="Error" Name="C:\WebSphere\AppServer\logs\http_plugin.log"/>
  <Property Name="ESIEnable" Value="true"/>
  <Property Name="ESIMaxCacheSize" Value="1024"/>
  <Property Name="ESIInvalidationMonitor" Value="false"/>
  <VirtualHostGroup Name="default_host">
    <VirtualHost Name="*:9080"/>
    <VirtualHost Name="*:80"/>
    <VirtualHost Name="*:9443"/>
    <VirtualHost Name="*:9081"/>
    <VirtualHost Name="*:9444"/>
    <VirtualHost Name="*:443"/>
  </VirtualHostGroup>
  <ServerCluster CloneSeparatorChange="false" LoadBalance="Round Robin"
Name="NortelCluster" PostSizeLimit="-1" RemoveSpecialHeaders="true" RetryInterval="60">
    <Server CloneID="10u0b4rhd" ConnectTimeout="0" ExtendedHandshake="false"
LoadBalanceWeight="2" MaxConnections="-1" Name="bc2srv5_WebSphere_Portal"
WaitForContinue="false">
      <Transport Hostname="bc2srv5.itso.ral.ibm.com" Port="9081" Protocol="http"/>
      <Transport Hostname="bc2srv5.itso.ral.ibm.com" Port="9444" Protocol="https">
        <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
        <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
      </Transport>
      <Transport Hostname="bc2srv5.itso.ral.ibm.com" Port="9091" Protocol="http"/>
      <Transport Hostname="bc2srv5.itso.ral.ibm.com" Port="9044" Protocol="https">
        <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
        <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
      </Transport>
    </Server>
    <Server CloneID="10u0b50d6" ConnectTimeout="0" ExtendedHandshake="false"
LoadBalanceWeight="2" MaxConnections="-1" Name="bc2srv6_WebSphere_Portal_2"
WaitForContinue="false">
      <Transport Hostname="bc2srv6.itso.ral.ibm.com" Port="9081" Protocol="http"/>
      <Transport Hostname="bc2srv6.itso.ral.ibm.com" Port="9444" Protocol="https">
        <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
        <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
      </Transport>
      <Transport Hostname="bc2srv6.itso.ral.ibm.com" Port="9091" Protocol="http"/>
      <Transport Hostname="bc2srv6.itso.ral.ibm.com" Port="9044" Protocol="https">
        <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
        <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
      </Transport>
    </Server>
  </Config>
```

```

        </Transport>
    </Server>
    <Server CloneID="10u8mkhb6" ConnectTimeout="0" ExtendedHandshake="false"
LoadBalanceWeight="2" MaxConnections="-1" Name="bc2srv8_WebSphere_Portal_3"
WaitForContinue="false">
        <Transport Hostname="bc2srv8.itso.ral.ibm.com" Port="9081" Protocol="http"/>
        <Transport Hostname="bc2srv8.itso.ral.ibm.com" Port="9444" Protocol="https">
            <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
            <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
        </Transport>
        <Transport Hostname="bc2srv8.itso.ral.ibm.com" Port="9091" Protocol="http"/>
        <Transport Hostname="bc2srv8.itso.ral.ibm.com" Port="9044" Protocol="https">
            <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
            <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
        </Transport>
    </Server>
</PrimaryServers>
<ServerCluster>
    <Server Name="bc2srv5_WebSphere_Portal"/>
    <Server Name="bc2srv6_WebSphere_Portal_2"/>
    <Server Name="bc2srv8_WebSphere_Portal_3"/>
</PrimaryServers>
</ServerCluster>
<ServerCluster CloneSeparatorChange="false" LoadBalance="Round Robin"
Name="server1_bc2srv5_Cluster" PostSizeLimit="-1" RemoveSpecialHeaders="true"
RetryInterval="60">
    <Server ConnectTimeout="0" ExtendedHandshake="false" MaxConnections="-1"
Name="bc2srv5_server1" WaitForContinue="false">
        <Transport Hostname="bc2srv5.itso.ral.ibm.com" Port="9080" Protocol="http"/>
        <Transport Hostname="bc2srv5.itso.ral.ibm.com" Port="9443" Protocol="https">
            <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
            <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
        </Transport>
    </Server>
</PrimaryServers>
</ServerCluster>
<ServerCluster CloneSeparatorChange="false" LoadBalance="Round Robin"
Name="server1_bc2srv6_Cluster" PostSizeLimit="-1" RemoveSpecialHeaders="true"
RetryInterval="60">
    <Server ConnectTimeout="0" ExtendedHandshake="false" MaxConnections="-1"
Name="bc2srv6_server1" WaitForContinue="false">
        <Transport Hostname="bc2srv6.itso.ral.ibm.com" Port="9080" Protocol="http"/>
        <Transport Hostname="bc2srv6.itso.ral.ibm.com" Port="9443" Protocol="https">
            <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
            <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
        </Transport>
    </Server>
</PrimaryServers>
</ServerCluster>
<ServerCluster CloneSeparatorChange="false" LoadBalance="Round Robin"
Name="WebSphere_Portal_bc2srv6_Cluster" PostSizeLimit="-1" RemoveSpecialHeaders="true"
RetryInterval="60">
    <Server ConnectTimeout="0" ExtendedHandshake="false" MaxConnections="-1"
Name="bc2srv6_WebSphere_Portal" WaitForContinue="false">
        <Transport Hostname="bc2srv6.itso.ral.ibm.com" Port="9081" Protocol="http"/>
        <Transport Hostname="bc2srv6.itso.ral.ibm.com" Port="9444" Protocol="https">
            <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>

```

```

        <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
    </Transport>
    <Transport Hostname="bc2srv6.itso.ral.ibm.com" Port="9091" Protocol="http"/>
    <Transport Hostname="bc2srv6.itso.ral.ibm.com" Port="9044" Protocol="https">
        <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
        <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
    </Transport>
</Server>
</PrimaryServers>
<Server Name="bc2srv6_WebSphere_Portal"/>
</PrimaryServers>
</ServerCluster>
<ServerCluster CloneSeparatorChange="false" LoadBalance="Round Robin"
Name="dmgr_bc2srv7Manager_Cluster" PostSizeLimit="-1" RemoveSpecialHeaders="true"
RetryInterval="60">
    <Server ConnectTimeout="0" ExtendedHandshake="false" MaxConnections="-1"
Name="bc2srv7Manager_dmgr" WaitForContinue="false"/>
    <PrimaryServers>
        <Server Name="bc2srv7Manager_dmgr"/>
    </PrimaryServers>
</ServerCluster>
<ServerCluster CloneSeparatorChange="false" LoadBalance="Round Robin"
Name="server1_bc2srv8_Cluster" PostSizeLimit="-1" RemoveSpecialHeaders="true"
RetryInterval="60">
    <Server ConnectTimeout="0" ExtendedHandshake="false" MaxConnections="-1"
Name="bc2srv8_server1" WaitForContinue="false">
        <Transport Hostname="bc2srv8.itso.ral.ibm.com" Port="9080" Protocol="http"/>
        <Transport Hostname="bc2srv8.itso.ral.ibm.com" Port="9443" Protocol="https">
            <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
            <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
        </Transport>
    </Server>
    <PrimaryServers>
        <Server Name="bc2srv8_server1"/>
    </PrimaryServers>
</ServerCluster>
<ServerCluster CloneSeparatorChange="false" LoadBalance="Round Robin"
Name="WebSphere_Portal_bc2srv8_Cluster" PostSizeLimit="-1" RemoveSpecialHeaders="true"
RetryInterval="60">
    <Server ConnectTimeout="0" ExtendedHandshake="false" MaxConnections="-1"
Name="bc2srv8_WebSphere_Portal" WaitForContinue="false">
        <Transport Hostname="bc2srv8.itso.ral.ibm.com" Port="9081" Protocol="http"/>
        <Transport Hostname="bc2srv8.itso.ral.ibm.com" Port="9444" Protocol="https">
            <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
            <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
        </Transport>
        <Transport Hostname="bc2srv8.itso.ral.ibm.com" Port="9091" Protocol="http"/>
        <Transport Hostname="bc2srv8.itso.ral.ibm.com" Port="9044" Protocol="https">
            <Property Name="keyring" Value="C:\WebSphere\AppServer\etc\plugin-key.kdb"/>
            <Property Name="stashfile" Value="C:\WebSphere\AppServer\etc\plugin-key.sth"/>
        </Transport>
    </Server>
    <PrimaryServers>
        <Server Name="bc2srv8_WebSphere_Portal"/>
    </PrimaryServers>
</ServerCluster>
<UriGroup Name="default_host_Norte1Cluster_URIs">
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/federateTest/*"/>

```

```

        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/federateTestServlet/*"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/wps/richText/*"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/wps/richTextServlet/*"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/wps/spreadSheet/*"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
... several additional stanzas like the above have been removed for brevity ...
.. each matches a particular function within the portal and routes it appropriately ..
Name="/wps/PA_1_0_CHServlet/*"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/wps/*"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/wsrp/*"/>
    </UriGroup>
    <Route ServerCluster="NortelCluster" UriGroup="default_host_NortelCluster_URIs"
VirtualHostGroup="default_host"/>
    <UriGroup Name="default_host_server1_bc2srv5_Cluster_URIs">
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/snoop/*"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/hello"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/hitcount"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsp"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsw"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsw"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/j_security_check"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/ibm_security_logout"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid"
Name="/servlet/*"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/ivt/*"/>
        <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/bpe/*"/>
    </UriGroup>
    <Route ServerCluster="server1_bc2srv5_Cluster"
UriGroup="default_host_server1_bc2srv5_Cluster_URIs" VirtualHostGroup="default_host"/>
    <RequestMetrics armEnabled="false" loggingEnabled="true" rmEnabled="false"
traceLevel="HOPS">
        <filters enable="false" type="URI">
            <filterValues enable="false" value="/servlet/snoop"/>
            <filterValues enable="false" value="/webapp/examples/HitCount"/>
        </filters>
        <filters enable="false" type="SOURCE_IP">
            <filterValues enable="false" value="255.255.255.255"/>
            <filterValues enable="false" value="254.254.254.254"/>
        </filters>
    </RequestMetrics>
</Config>

```

Load balancing WebSphere Application Servers: HTTPS

The Example 6-6 on page 97 configuration uses the GbESM to provide load balancing directly to the Application Servers. In this case, we use https to connect from the client browser to the Application servers.

As a result, even though the same cookies are still used, they are transmitted in encrypted form and the GbESM cannot see them. Instead, the GbESM is configured to use the SSL session ID to ensure persistence. This session ID is created when the client and the server

complete the initial https handshake and is reused through multiple TCP connections between the client and server.

In order to use SSL-ID persistence, this example uses delayed binding just as the preceding example does, and is also an example of Layer 7 switching. Similar to the previous example, SSL connections are supported and load balanced on port 9444 as well as on port 443, to match the default configuration of the Application Server. Port 443 is redirected to 9444 similar to the treatment of ports 80 and 9081 above.

Note that the configuration from the HTTP example is included within this example. This is to allow the use of a redirect. This option can be configured so that an end user can initially specify HTTP and have the server send his request to the appropriate https URL without additional user effort or the need to remember to type **https://** ... at the beginning of the URL.

Example 6-6 Load balancing WebSphere Application Servers (https)

```
script start "Nortel Networks Layer2-7 GbE Switch Module" 4 /**** DO NOT EDIT THIS LINE!
/* Configuration dump taken 0:34:54 Fri Jan 2, 2070
/* Version 20.2.2.6, Base MAC address 00:0f:06:eb:58:00
/c/12/vlan 1
    def INT1 INT2 INT3 INT4 INT5 INT6 INT7 INT8 INT9 INT10 INT11 INT12 INT13 INT14 EXT1 EXT2
EXT3 EXT4
/c/12/stg 1/off
/c/12/stg 1/clear
/c/12/stg 1/add 1
/c/12/trunk 1
    ena
    failovr ena
    add EXT2
/c/13/if 1
    ena
    addr 9.42.171.247
    mask 255.255.255.0
    broad 9.42.171.255
/c/13/gw 1
    ena
    addr 9.42.171.3
/c/13/vrrp/on
/c/13/vrrp/trnkfo enabled
/c/13/vrrp/vr 1
    ena
    vrid 10
    if 1
    addr 9.42.171.252
    track
        ports e
/c/slb/adv
    direct ena
    grace ena
/c/slb/sync
    prios d
    reals e
    state e
/c/slb/sync/peer 1
    ena
    addr 9.42.171.21
/c/slb/real 1
    ena
    rip 9.42.171.63
```

```
/c/slb/real 2
  ena
  rip 9.42.171.85
/c/slb/real 3
  ena
  rip 9.42.171.86
/c/slb/real 5
  ena
  rip 9.42.171.45
/c/slb/real 6
  ena
  rip 9.42.171.57
/c/slb/real 8
  ena
  rip 9.42.171.243
/c/slb/group 1
  metric minmisses
  add 1
  add 2
  add 3
/c/slb/group 2
  health http
  add 5
  add 6
  add 8
/c/slb/port INT1
  client ena
  server ena
/c/slb/port INT2
  client ena
  server ena
/c/slb/port INT5
  server ena
/c/slb/port INT6
  server ena
/c/slb/port INT7
  client ena
  server ena
/c/slb/port INT8
  server ena
/c/slb/port INT9
  server ena
/c/slb/port INT10
  server ena
/c/slb/port INT12
  server ena
/c/slb/port EXT1
  client ena
  server ena
/c/slb/port EXT2
  client ena
/c/slb/virt 1
  ena
  vip 9.42.171.252
/c/slb/virt 1/service 9081
  group 2
  dbind ena
/c/slb/virt 1/service http
  group 2
  rport 9081
```

```

    dbind ena
/c/slb/virt 1/service 9444
  group 2
  dbind ena
/c/slb/virt 1/service https
  group 2
  rport 9444
  dbind ena
/c/slb/virt 1/service 9081/pbind cookie passive JSESSIONID 29 9 disable
/c/slb/virt 1/service 9081/rcount 1
/c/slb/virt 1/service 80/pbind cookie passive JSESSIONID 29 9 disable
/c/slb/virt 1/service 80/rcount 1
/c/slb/virt 1/service 9444/pbind sslid
/c/slb/virt 1/service 443/pbind sslid
/
script end /**** DO NOT EDIT THIS LINE!

```

6.3.2 Time-out configuration issues

All load balanced connections managed by the switch create session table entries. Sessions where persistence is in use have additional entries to hold the data required for persistence to work, such as cookie data, SSL session ID data, and so on. All of these entries need to be removed from the session table when they are no longer valid or no longer useful. There are three timing parameters which can be configured to control the removal of stale entries from the session table. They are described here.

- ▶ *fastage* (`cfg/slb/adv/fastage`) is a global parameter which controls the removal of sessions after they have been normally closed - by a standard TCP FIN exchange between the session partners. The default interval before entries are removed is 2 seconds. This parameter is rarely changed. Note that the default value of the parameter is 0, and adding 1 to it doubles the time interval, so a value of 2 would correspond to a time interval of 8 seconds ($2 * 2^2$).
- ▶ *tmout* (`/cfg/slb/real <x>/tmout`) controls the removal of sessions which have been idle for an extended period of time on the configured real server. It is intended to remove sessions where one of the session partners abnormally terminated the software associated with the session (e.g. client or server process, or their entire OS). However, long idle periods with no traffic are normal for some software packages. This parameter can be used to prevent the switch from unexpectedly terminating sessions which have been idle when long idle periods are part of the normal operation of the software. The *tmout* parameter also controls the length of time a persistence entry will be kept in the session table when it does not have any active sessions associated with it. The default value of this parameter is 10 minutes.
- ▶ *slowage* (`cfg/slb/adv/slowage`) is a global parameter which controls the frequency of the slowage sweep, the process which removes session table entries. It also has a multiplier effect on the actual time required before entries are removed as a result of the *tmout* parameter. Examples are shown immediately following. The default interval for slowage is 2 minutes; adding 1 to the value doubles the interval. At the maximum setting of 15, table entries can survive for over a month.

Note: Use care when setting the `slowage` parameter to larger values. The session table can grow to the point that it consumes the maximum amount of memory available for it. There is a tradeoff between the maximum age of idle session entries and the space available to hold them.


Some examples are:

- Use of tmout and slowage together.
 - Set an idle timeout of 30 minutes.
 - If the slowage parameter to is set to 4, the slowage interval is $2*2^4 = 32$ minutes. The effective tmout value is also multiplied by 2^4 (16), so it becomes 8 hours.
- If the objective is for idle sessions to last throughout a work day (8 hours) and persistence last overnight (24 hours), then:
 - Set tmout to 480 minutes (8 hours). When applied with the slowage parameter the result will be $480*2*2$ or 32 hours.
 - Set slowage to 2. The session table will then be cleaned out every $2*2*2$ or 8 minutes.

Settings such as these are useful for environments such as Citrix and WebSphere to ensure that disconnected or idle users retain their association with the selected member of a load balance pool.

Note: For additional material discussing the configuration options regarding WebSphere Application Server using SSL, see Deploying a Secure Portal Solution on Linux Using WebSphere Portal V5.0.2 and Tivoli Access Manager V5.1:

<http://www.redbooks.ibm.com/abstracts/redp9121.html?open>



Load balancing with Citrix MetaFrame and Microsoft Terminal Services

Citrix MetaFrame provides a super-set of the functionality of Microsoft's Terminal Services. Terminal Services is now a standard part of Microsoft Windows 2000 Server and Windows 2003 Server. MetaFrame allows a richer remote desktop environment than Terminal Services through the use of a proprietary client known as the *ICA Client*. Other client implementations such as Java are also available. The key use of Citrix and of Terminal Services in an organization is to move the processing of desktop applications from client desktop and mobile computers which can number in the thousands to a comparatively small number of servers. The clients, potentially including so-called thin clients, can then access the applications over a network.

Citrix provides additional value-add components such as its Web Access Portal where load balancing can also be useful. See 7.3.4, "Load balancing additional Citrix services" on page 110.

7.1 Value of load balancing with Microsoft Terminal Services

Using load balancing with MTS provides the scalability and availability that comes with load balancing in general. It enables end users to transparently access an available server from a pool of Terminal Servers using the Remote Desktop client. Outages on servers within the pool will be largely invisible to end users.

7.2 Value of load balancing with Citrix MetaFrame

During the writing of this Redpaper, Citrix Access Suite 4.0 was released. The key business and technical benefits of Citrix Access Suite 4.0 can be viewed at the following Web site:

<http://www.citrix.com/English/ps2/products/product.asp?contentID=12752>

However, the software used during the development of this Redbook was Citrix MetaFrame XPe. Of the Citrix MetaFrame product suite, there are three levels of licensing for the MetaFrame product: XPs (standard), XPa (advanced), and XPe (enterprise). The two higher levels of the product include software-based load balancing across multiple MetaFrame servers which are members of the same server farm.

Note: Based on the latest release of Citrix Access Suite 4.0, you should review its software load-balancing features to determine if these features address your business requirements.

The purpose of this chapter is to discuss the implementation of the hardware-based load balancing using the Nortel GbESM with Citrix MetaFrame. Also, the rationale for using hardware-based load balancing instead of software will be covered.

Key aspects of the justification for hardware-based load balancing are integrated high-availability functions and high performance of load balancing. Several sophisticated algorithms are available to provide an even balance *as possible* across the pool servers.

In addition, ancillary components of Citrix such as the Web Interface and Secure Portal can be load balanced. There is value in doing so since Citrix itself does not provide the ability to load balance these services. A sample configuration for this purpose is shown in 7.3.4, “Load balancing additional Citrix services” on page 110.

7.3 Implementing load balancing with Citrix MetaFrame

The steps to implement load balancing with Citrix are:

1. Create the Citrix servers and configure the server farm. In order to use the GbESM to provide load balancing, it is best if one or more groups of servers which provide the identical applications are configured, as opposed to one server for Word and a different server for Excel®, and so forth.
2. Configure sign-on security for the Citrix servers to use a shared mechanism, such as a Domain Controller. Users should be able to use the same credentials to get the same level of access to any server.
3. Test the configuration at this point. Users should be able to sign on to servers and to applications.

4. Define the configuration for the GbESM switches. As always, two switches are recommended for redundancy. A sample configuration is shown in Example 7-1, “GbESM base configuration for Citrix and Terminal Server” on page 108 below.
5. Review persistency requirements. A discussion of the configuration parameters effected by this is in 7.3.3, “Persistence and timing considerations” on page 109. The relevant questions are:
 - How long should a disconnected user maintain a logical association with the last Citrix server he used? For example, should the association survive overnight, over a weekend, over a two week vacation?
 - How long should an idle user (generating no traffic) maintain their connection to a Citrix server before being disconnected?
6. Install and test the configuration by creating a custom connection for a desktop associated with the virtual server address from the GbESM configuration. See 7.3.1, “Functions that can be load balanced” on page 103.
7. Install and test published applications associated with the virtual server.

7.3.1 Functions that can be load balanced

Citrix provides a variety of ways of accessing a MetaFrame server. They are listed here:

Desktop access

Citrix offers the ability to use a remote desktop on a MetaFrame server which is similar to that provided by Microsoft Terminal Services. Both provide configurable access to disks and other resources on the client machine and both can be displayed within a window or take up the entire screen.

The GbESM can load balance access to desktops. This is done by creating a load balance group for the appropriate ports (1494 and 2589), and creating a custom ICA connection. The ICA connection would point to the configured virtual address (VIP) for the Citrix servers.

The GbESM configuration to do this is shown in 7.3.2, “GbESM Configuration for Citrix and Terminal Server” on page 108. Desktops are typically accessed through the Citrix ICA client, which is installed on client systems. The custom connections page of the ICA client and a sample of the details of a custom connection are shown in Figure 7-1 on page 104 through Figure 7-3 on page 106. The items identified with the IP address 9.42.171.248 are pointing to the load balancing pool.

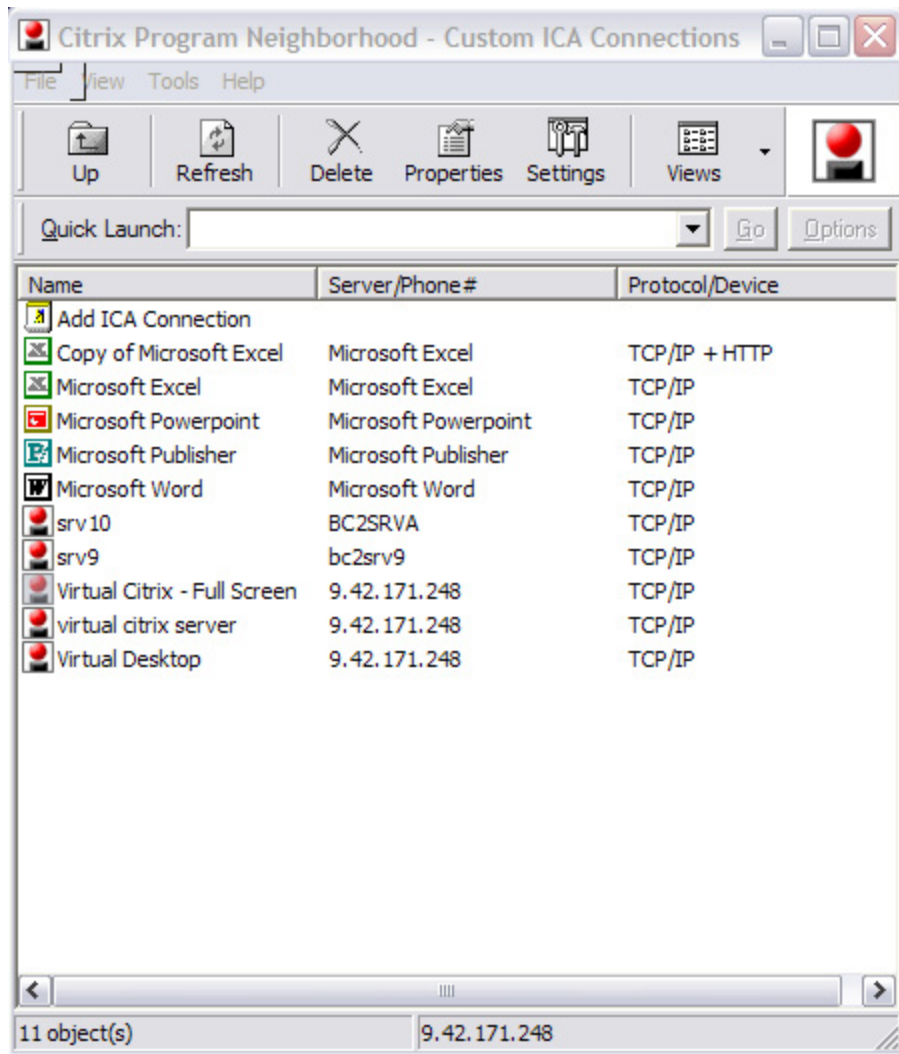


Figure 7-1 Custom ICA connections for Citrix

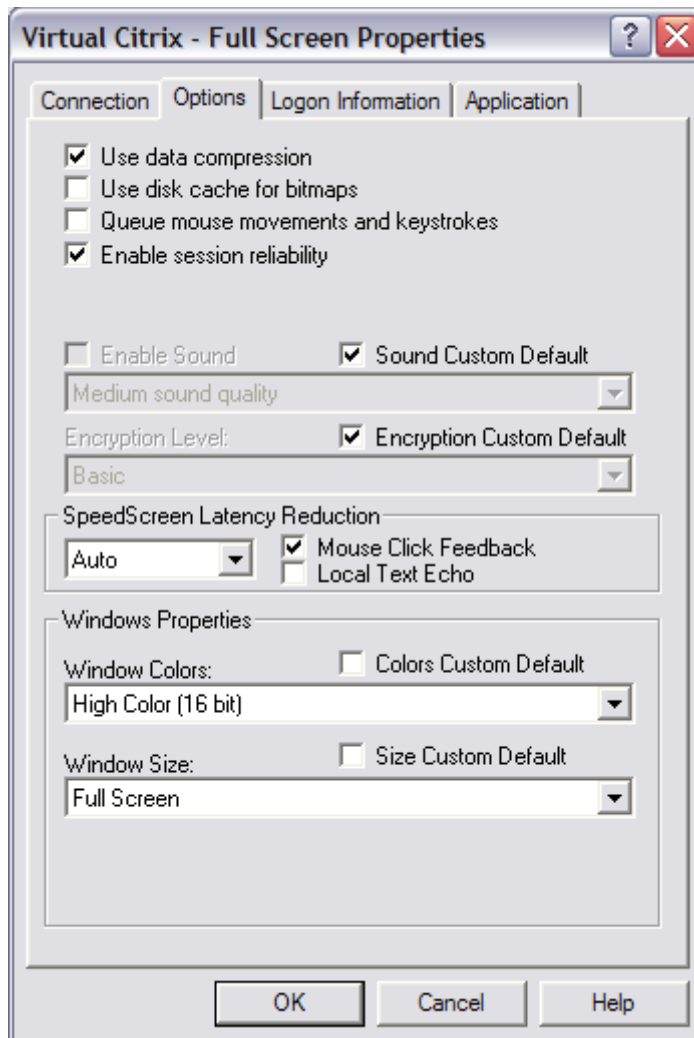


Figure 7-2 Full Screen Desktop - Options

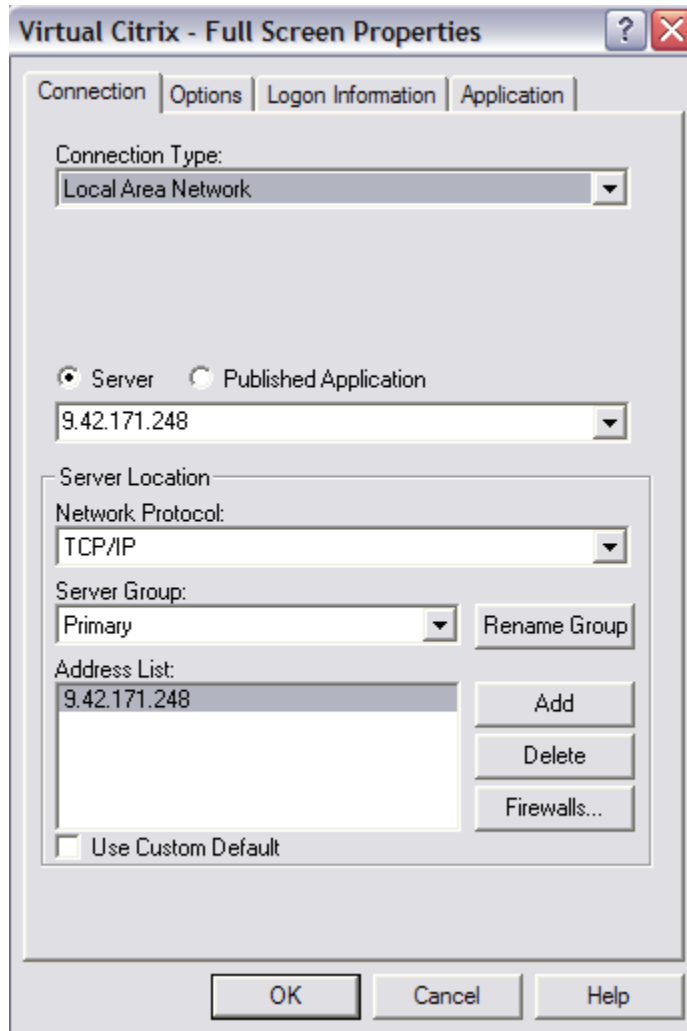


Figure 7-3 Full Screen Desktop - Connection Properties

Note that in Figure 7-2 on page 105, the **Enable Session Reliability** box is checked. This option will cause connections to use port 2598 rather than 1494. Also, the window is set to take the full screen by the settings shown in the lower half of the dialog box.

Published Application Access

Citrix provides another form of access, where desktops and specific applications can be published and thus made accessible to end users at client machines. For applications, a window appears on the client desktop which is hard to distinguish from the window which would appear if the application were being run locally on the client machine. It is possible for a client to have multiple remote application windows open with each application running on a different Citrix server.

When this technique is used, the Citrix client performs a browsing function to determine which servers can provide the selected application, and chooses one of the available servers. The browsing function itself can be load balanced. It works through an XML responder on port 80, but there is not much added value obtained by doing that.

A sample custom ICA connection to load balance the browsing function is shown in Figure 7-4 on page 107. Note that the server address shown, in both places, is that of the VIP

configured on the GbESM, and that this is a custom connection item, which is different from and can be created from the item in the published application set.

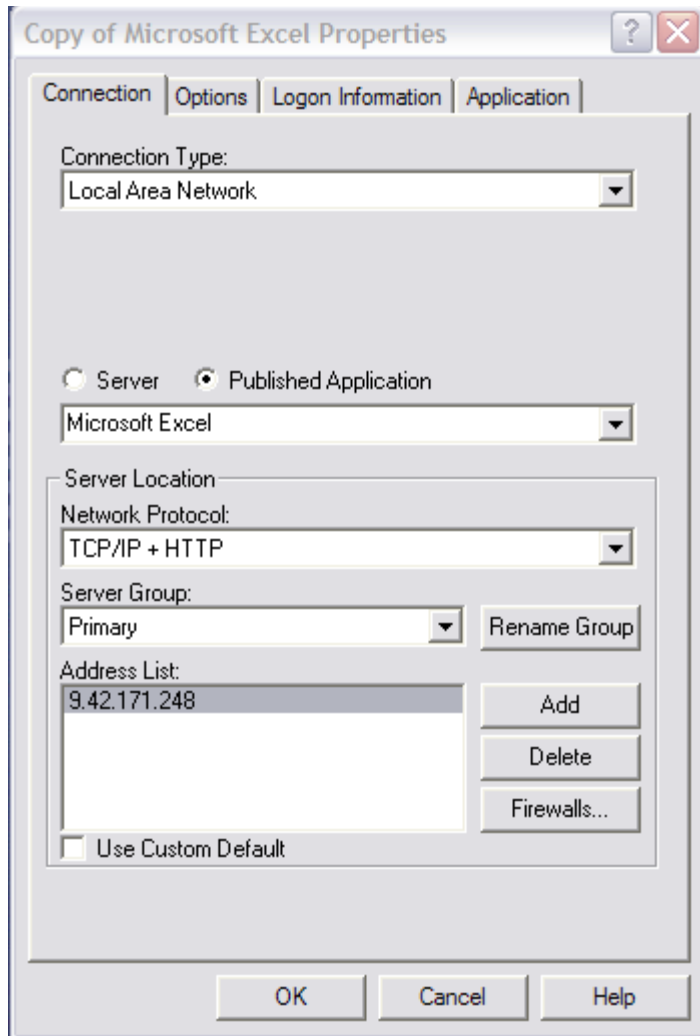


Figure 7-4 Custom ICA Connection for a Published Application

Web Interface Server

Citrix users can access published applications through a Web Interface Server. This server presents a view of the available applications in a client browser rather than in the ICA client. The Web Interface can work with the ICA client, but if there is no client software installed on a client computer, then there are other options as follows:

- ▶ Java client - for any browser which can support a JVM™
- ▶ Active X client for MS Internet Explorer
- ▶ Netscape plug-in

Published applications can be accessed using the Web Interface Server, whichever underlying Citrix client it is using. However, there does not appear to be the ability to create custom connections using the Web Interface.

The GbESM can load balance access to the Web Interface Server itself. An example of how this is done is included in 7.3.4, “Load balancing additional Citrix services” on page 110.

7.3.2 GbESM Configuration for Citrix and Terminal Server

A sample switch configuration for Citrix and Terminal Server is shown in Example 7-1. The primary function of this configuration is to balance requests to the Citrix servers on port 2598. Legacy Citrix clients can optionally be load balanced on port 1494 and the application browser service and the Web Interface server on port 80. Also, clients using the Microsoft Remote Desktop are balanced on port 3389.

Note: An enhancement to improve load balancing of Terminal Services will be introduced in the next software release for the L2-7 GbESM. It is likely to introduce additional configuration commands.

The configuration shown in Example 7-1 uses Trunk Failover with VRRP to provide a High Availability environment. Hot Standby could be used here instead, if you want. A sample configuration with Hot Standby is included in 6.3.1, "Configuration examples" on page 79.

Example 7-1 GbESM base configuration for Citrix and Terminal Server

```
script start "Nortel Networks Layer2-7 GbE Switch Module" 4 /**** DO NOT EDIT THIS LINE!
/* Configuration dump taken 7:06:03 Mon Jan 5, 2070
/* Version 20.2.2.6, Base MAC address 00:0e:62:38:19:00
/c/12/vlan 1
    def INT1 INT2 INT3 INT4 INT5 INT6 INT7 INT8 INT9 INT10 INT11 INT12 INT14 EXT1 EXT2
EXT3 EXT4

/c/12/stg 1/off
/c/12/stg 1/clear
/c/12/stg 1/add 1
/c/12/trunk 1
    ena
    failovr ena
    add EXT2
/c/13/if 1
    ena
    addr 9.42.171.21
    mask 255.255.255.0
/c/13/gw 1
    ena
    addr 9.42.171.3
/c/13/vrrp/on
/c/13/vrrp/trnkfo enabled
/c/13/vrrp/vr 2
    ena
    vrid 2
    if 1
    prio 101
    addr 9.42.171.248
    track
        ports e
/c/slb/adv
    direct ena
    grace ena
/c/slb/sync
    prios d
    reals e
    state e
/c/slb/sync/peer 1
    ena
    addr 9.42.171.247
```



```

/c/slb/real 9
    ena
    rip 9.42.171.215
    tmtout 1000
/c/slb/real 10
    ena
    rip 9.42.171.156
/c/slb/group 3
    metric minmisses
    add 9
    add 10
/c/slb/port INT9
    server ena
/c/slb/port INT10
    server ena
/c/slb/port EXT1
    client ena
    server ena
/c/slb/port EXT2
    client ena
/c/slb/virt 2
    ena
    vip 9.42.171.248
/c/slb/virt 2/service 2598
/* ICA sessions with session reliability enabled */
    group 3
    pbind clientip
/c/slb/virt 2/service 1494
/* sessions without session reliability and from older ICA clients */
    group 3
    pbind clientip
/c/slb/virt 2/service http
/* application browser and web interface server */
    group 3
    pbind clientip
/c/slb/virt 2/service 3389
/* MS Remote Desktop Client access */
    group 3
    pbind clientip
script end /**** DO NOT EDIT THIS LINE!

```

7.3.3 Persistence and timing considerations

All load-balanced connections managed by the switch create session table entries. Sessions where persistence is in use have additional entries to hold the data required for persistence to work, such as cookie data, SSL session ID data, and so on. All of these entries need to be removed from the session table when they are no longer valid or no longer useful. There are three timing parameters which can be configured to control the removal of stale entries from the session table. The details of these parameters can be found in 6.3.2, “Time-out configuration issues” on page 99.

The key issue for Terminal Server and for Citrix is how long to keep an association between a disconnected client and a particular MetaFrame server. With appropriate configuration the maximum time can be several weeks, if you want. As always, there is a tradeoff between the length of time entries can be kept in the session table and the maximum size of the table.

Session table entries in the configuration above will persist after the sessions are closed (normally or through being disconnected) because of the use of the `pbind clientip` command.

7.3.4 Load balancing additional Citrix services

There are other components of Citrix which can usefully be load balanced. The components are:

- ▶ Citrix Secure Gateway (SG)
- ▶ Citrix Web Interface (WI)
- ▶ Citrix Secure Ticket Authority (STA)
- ▶ Citrix MSAM Portal (MSAM)

These items are not load-balanced by the MetaFrame servers, so the GbESM adds value by providing load balancing for them. The use of the GbESM adds High Availability to the design as well. For example, an environment could survive the failure of all but one WI/SG server, and all but one STA server, and all but one MSAM server, even if all of these failures happened simultaneously. If two switches were deployed, as is typical, failure of a switch or of the connection between a switch and the remainder of the network would not disrupt availability of the Citrix environment. An architecture such as this makes it possible to use Citrix for mission critical applications.

Because a load balanced pool uses one URL for the entire pool as opposed to one URL for each server, the number of digital certificates required for the secure components is reduced. It is quite possible to have an entire BC chassis filled with 14 WI servers using one URL and one certificate instead of 14 certificates. This represents a potential dollar savings of roughly \$7 000. Larger designs with more than one BC full of WI servers are also possible, and similar server farms can be deployed for the MSAM service.

A sample configuration with commentary is shown Example 7-2 on page 111. This configuration was developed from work done in Dallas in 2004 and was not replicated during the production of this Redpaper.

Citrix additional items example summary and configuration

Six servers were used. The details of the servers are as follows:

- ▶ Addresses 192.168.3.10 and 11 - Web Interface and Secure Gateway components are on these servers, names WISG01 and WISG02. The virtual address that points to this load balancing pool is "WISGCOMMON", 192.168.3.101. Server group 1 is associated with port 80 and group 11 is associated with group 443.
- ▶ Addresses 192.168.3.14 and 15 - Secure Ticket Authority. The servers are named "STAMF" (with metaframe) and "STA02" (without). Virtual address "STACOMMON" (192.168.3.103) points to this pool. Server group 3 is associated with port 80 and group 33 is associated with port 443 for the STA servers.
- ▶ Addresses 192.168.3.17 and 18 - MSAM portal servers. Pointed to by virtual address "MSAMCOMMON", 192.168.3.102. Server group 2 is associated with port 80 and group 22 is associated with port 443.
- ▶ All of the internal ports which were used are configured with both client and server processing enabled. This allows the various components to exploit load balancing on those other components with which they communicate; in particular it allows the SG to use a virtual address to access the STA.
- ▶ VRRP is included in the sample configuration shown. A similar configuration with differing VRRP priorities would be used on another switch to effectively exploit VRRP.

Example 7-2 Load Balancing for additional Citrix Services

```
script start "Nortel Networks Layer2-7 GbE Switch Module" 4 /**** DO NOT EDIT THIS LINE!
/* Configuration dump taken 3:54:10 Thu Jan 1, 2070
/* Version 20.1.1, Base MAC address 00:0e:62:38:7d:00
/c/port INT1
    pvid 103
/c/port INT2
    pvid 103
/c/port INT3
    pvid 103
/c/port INT4
    pvid 103
/c/port INT10
    pvid 103
/c/port INT12
    pvid 103
/c/port EXT1
    pvid 103
    tag ena
/c/port EXT2
    tag ena
    pvid 103
/c/port EXT3
    tag ena
    pvid 103
/c/port EXT4
    tag ena
    pvid 103
/c/12/vlan 103
    ena
    name "VLAN 103"
    def INT1 INT2 INT3 INT4 INT10 INT12 EXT1 EXT2 EXT3 EXT4
/c/12/vlan 1
    ena
    def INT5 INT6 INT7 INT8 INT9 INT11 INT13 INT14
/* Vlan Layout: */
/* VLAN 1 (default vlan) - not used for production traffic */
/* VLAN1 includes internal ports for missing or unused blades */
/* Vlan 103 - main VLAN for blades being used and external ports */
/c/12/stg 1/off
/c/12/stg 1/clear
/c/12/stg 1/add 1 103
/c/12/trunk 1
    ena
    failovr dis
    add EXT1
    add EXT2
    add EXT3
    add EXT4
/c/13/if 1
    ena
    addr 192.168.3.200
    vlan 103
/c/13/gw 1
    ena
    addr 192.168.3.254
    arp enabled
/c/13/dns
    prima 192.168.3.254
/c/13/vrrp/on
```

```

/* virtual router address for inbound static routes */
/c/13/vrrp/vr 1
  ena
  vrid 1
  if 1
  addr 192.168.3.1
/* Web Interface & Secure Gateway VSR */
/c/13/vrrp/vr 101
  ena
  vrid 101
  if 1
  addr 192.168.3.101
/* Secure Ticket Authority VSR*/
/c/13/vrrp/vr 102
  ena
  vrid 102
  if 1
  addr 192.168.3.102
/* MSAM Portal - VSR */
/c/13/vrrp/vr 103
  ena
  vrid 103
  if 1
  addr 192.168.3.103
/c/slb/adv
  direct ena
/c/slb/sync
  prios d
/c/slb/sync/peer 1
  ena
  addr 192.168.3.201
/c/slb/real 10
  ena
  rip 192.168.3.10
  name "wisg1"
/c/slb/real 11
  ena
  rip 192.168.3.11
  name "wisg2"
/c/slb/real 14
  ena
  rip 192.168.3.14
  name "stamf"
/c/slb/real 15
  ena
  rip 192.168.3.15
  name "sta02"
/c/slb/real 17
  ena
  rip 192.168.3.17
  name "msam02"
/c/slb/real 18
  ena
  rip 192.168.3.18
  name "msam01"
/c/slb/group 1
  metric hash
  add 10
  add 11
  name "wisg"

```

```
/c/slb/group 2
  metric hash
  add 17
  add 18
  name "msam"
/c/slb/group 3
  metric hash
  add 14
  add 15
  name "sta"
/c/slb/group 11
  metric hash
  add 10
  add 11
  name "WISG443"
/c/slb/group 22
  metric hash
  add 17
  add 18
  name "MSAM443"
/c/slb/group 33
  metric hash
  add 14
  add 15
  name "STA443"
/c/slb/port INT1
  client ena
  server ena
/c/slb/port INT2
  client ena
  server ena
/c/slb/port INT3
  client ena
  server ena
/c/slb/port INT4
  client ena
  server ena
/c/slb/port INT10
  client ena
  server ena
/c/slb/port INT12
  client ena
  server ena
/c/slb/port EXT1
  client ena
/c/slb/port EXT2
  client ena
/c/slb/port EXT3
  client ena
/c/slb/port EXT4
  client ena
/c/slb/virt 1
  ena
  vip 192.168.3.101
  dname "wisgcommon"
/c/slb/virt 1/service https
  group 11
/c/slb/virt 1/service http
  group 1
/c/slb/virt 2
```

```
ena
vip 192.168.3.102
/c/slb/virt 2/service http
group 2
/c/slb/virt 2/service https
group 22
/c/slb/virt 3
ena
vip 192.168.3.103
/c/slb/virt 3/service https
group 33
/c/slb/virt 3/service http
group 3
/
script end /**** DO NOT EDIT THIS LINE!
```



Load balancing with VMware

VMware, specifically the VMware ESX product, is an operating system which allows the provisioning of multiple operating systems (Windows, Linux) running in *Virtual Machines* (VMs) which share the hardware of one or more physical servers. The key benefit of this is to use more fully the capacity of the available servers, and potentially reduce their number. An added benefit of VMware is that the virtual machines use a standard and consistent set of drivers, allowing them to be easily migrated from one server to another even if the servers in question are not the same model, do not have the same CPU, or the same number of CPUs, and so forth.

8.1 Value of load balancing with VMware

In general, load balancing with VMware's virtual machines would be implemented for the same reasons that it would be implemented across a collection of physical servers. That is, load balancing allows an application to grow by scaling out (more servers). Applications that scale out are typically limited by I/O and network requirements rather than by CPU or memory.

Load balancing with VMware allows the implementation of High Availability designs which can survive outages on one or more servers. Virtual machines can also be used as backup or overflow servers for an application (even one running on dedicated, real servers) while other virtual machines take up the bulk of the processing power of the physical server blade and support an entirely different application.

8.2 Implementing load balancing with VMware

The standard requirements for load balancing are unchanged when implemented under VMware. What is different is how VMware supports NIC teaming, a capability which allows a server and its applications to survive a variety of network outages. In Windows and Linux implementations on the currently available server blades, NIC teaming is done by configuring drivers from Broadcom or Intel (in the case of the HS40). When Windows or Linux is implemented under VMware ESX, these client operating systems do not have access to the physical NICs but rather use a generic virtual NIC to access the network. Therefore, NIC teaming is implemented by appropriate configuration of ESX, which does have access and visibility to the physical NICs.

The details in this chapter illustrate the necessary configuration to implement NIC teaming under VMware ESX, both for use by the client virtual machines as well as the ESX console.

8.2.1 Preparation

The steps in this section are useful, if not necessary, for any implementation of VMware in the BladeCenter. Most of this information can be found in (and has been adapted from) the most current version of the VMware installation and administration guides.

Note: All of the steps below to implement High Availability, NIC teaming, and VLANs are done only to VMware and not to any of the guest operating systems. Elsewhere, in this document and in the L2/3 Redbook, there are detailed configuration for the functions mentioned but they should never be used for a VMware guest machine. Get the L2/3 Redbook here:

<http://www.redbooks.ibm.com/abstracts/redp3586.html?Open>

This is a potential benefit because when you have configured VMware, all of the guest machines can take advantage of the HA functions without needing to explicitly configure them.

NIC teaming with VMware

Unlike the other operating systems which can run on the server blades (different versions of Windows, Linux, and so on), VMware does not use drivers from Broadcom or Intel (in the case of the HS40) to implement NIC teaming. Instead, various VMware configuration files are edited to create what are referred to as *bonds* from multiple physical NICs, and to designate the *home* NIC within a bond.

VMware will by default use the first NIC for the *virtual console*, leaving the remaining NICs for use by the virtual machines. This creates a problem for HS20 and other one-slot blade servers, which will typically be configured with only two NICs.

In summary, the recommended configuration for a one-slot server blade involves the following:

1. Share the first NIC between the virtual console and the guest virtual machines.
2. Create a bond using the two available physical NICs.
3. Designate one NIC, typically the one connected to I/O bay 1, as the home NIC for the bond.
4. Configure the virtual console to use the bond for network access. The virtual console will be reached by the IP address configured for VMware itself while each virtual machine will have its own IP address(es).
5. Enable bond failover so that VMware will switch from the home NIC to a backup when needed.
6. Optionally, the virtual console and any virtual machines can be assigned to different VLANs.

The steps above are outline in more detail in these next sections and sample configuration files are provided. This is an overview and is not intended to be a substitute for the relevant VMware documentation which provides far more information.

Share service console NIC with virtual machines

This is accomplished by the use of the `vmkpcidivv` utility, use of which requires root access. (It may also be possible to use the Web version at <http://<service console address>/pcidivv>.) Using the text version, accept all the defaults or existing configured values until you reach the NIC which was initially allocated to the service console (marked with an 'c'). Change its configuration to a 's' for shared. The remaining NICs will be marked 'v' which indicates that they are dedicated to use by the guest virtual machines.

Create the bond (NIC team) and identify the home NIC

This step is done by editing the `/etc/vmware/hwconfig` file. For a two-NIC server, the following lines are added at the end of the file:

```
nicteam.vmnic0.team = "bond0"  
nicteam.vmnic1.team = "bond0"  
nicteam.bond0.home_link = "vmnic1"
```

This text, like all of the VMware configuration that follows, is case-sensitive. The spaces and quotation marks are required. Bonds can be numbered from 0 to 9 if there is a need to create more than one.

Note that the first NIC, which was initially dedicated to the console and which is connected to the switch in bay 1 will commonly be identified as `vmnic1` on a 2-NIC server, and with the highest number on larger servers such as HS40s.

Configure the service console to use the bond

The file, `/etc/rc.local`, which is executed at VMware boot time, is edited. These steps should be performed from the VMware physical console or the Management Module remote access page.

The following commands are added at the end of the file:

```
/etc/rc.d/init.d/network stop  
rmmod vmxnet_console
```

```
insmod vmxnet_console devName=bond0
/etc/init.d/network start
mount -a
```

Note that the 'N' in devName must be capitalized.

If the use of multiple VLANs is planned, then the console can be assigned to a specific VLAN by changing the insmod line to read:

```
insmod vmxnet_console devName=bond0.<Vlan#>
```

Virtual machines can be configured to use VLANs through the creation of *virtual switches and associated port groups* (VLANs) using Web access to the VMware Management Interface. The internal ports on the L2-7 GbESM must be configured to support all of the same VLANs as are assigned to either the service console or any of the virtual machines.

Note: VMware does not support the use of an untagged VLAN when tagged VLANs are in use on a NIC or bond. The ability to configure the default (PVID) VLAN to be tagged will be added in the next software release for the GbESM (/cfg/port INT<x>/tag-pvid). In the interim the best approach is to create an unused VLAN, and assign it as the PVID for all tagged ports connecting to VMware machines. See the example in Appendix B, "Workaround for VMware use of VLAN tags" on page 133.

Enable Bond Failover

This parameter can be changed through Web access to the Management Interface and root access is required.

- ▶ Click the **options** tab and go to **advanced settings**.
- ▶ Set Net.ZeroSpeedlinkdown to **1**.

Note: Unlike the Broadcom BASP driver, there is no option in VMware to enable failover with no fallback. When the home link again becomes active, it will preempt whichever backup link was in service. This means that preempt must be enabled on the GbESM, which it is by default. It also means that the no-fallback option is not available to help avoid the disruption caused when the primary returns to service. We recommend returning the primary link to service during a scheduled maintenance window.

8.2.2 Sample configuration files

The samples in this section come from a test environment which includes the following:

- ▶ HS40 server with a SCSI sidecar
- ▶ VMware ESX 2.5.1
- ▶ Two virtual machines running Windows 2003 server
- ▶ Two L2-7 GbESM modules running firmware version 20.2.2.6.

The configurations are annotated to show how they would be different when an HS20 or similar blade is used.

The GbESM configurations and screen shots show High Availability functions; these configurations will work equally well with stand-alone server blades if they have the necessary drivers (Broadcom or other).

Example 8-1 on page 119 was taken from our test HS40 and thus has four NICs. In an HS20, only nic0 and nic1 would appear. The lines which would not appear in an HS20 or similar blade are marked. Also, the lines which were manually added as part of our tests are marked;

the remainder of the file is generated automatically. The manually added entries do the following:

- ▶ Define two NIC teams, one for the NICs in the left hand (lower numbered) side of the HS40 and one for the right hand (higher numbered slot) side. The higher numbered side would not exist in a single-wide server blade.
- ▶ Define the home link for each NIC team.

Example 8-1 /etc/vmware/hwconfig file

```
cat /etc/vmware/hwconfig
device.0.0.0.class = "060000"
device.0.0.0.devID = "0014"
device.0.0.0.name = "ServerWorks: Unknown device 0014 (rev 33)"
... configuration for additional hardware devices skipped ...
device.1.1.0.name = "Intel Corporation 8254NXX Gigabit Ethernet Controller (rev 04)"
device.1.1.0.subsys_devID = "34b1"
device.1.1.0.subsys_vendor = "8086"
device.1.1.0.vendor = "8086"
device.1.2.0.class = "020000"
device.1.2.0.devID = "1028"
device.1.2.0.name = "Intel Corporation 8254NXX Gigabit Ethernet Controller (rev 04)"
device.1.2.0.subsys_devID = "34b1"
device.1.2.0.subsys_vendor = "8086"
device.1.2.0.vendor = "8086"
device.2.1.0.class = "020000"
device.2.1.0.devID = "107b"
device.2.1.0.name = "Intel Corporation 8254NXX Gigabit Ethernet Controller (rev 03)"
device.2.1.0.subsys_devID = "34b1"
device.2.1.0.subsys_vendor = "8086"
device.2.1.0.vendor = "8086"
device.2.1.1.class = "020000"
device.2.1.1.devID = "107b"
device.2.1.1.name = "Intel Corporation 8254NXX Gigabit Ethernet Controller (rev 03)"
device.2.1.1.subsys_devID = "34b1"
device.2.1.1.subsys_vendor = "8086"
device.2.1.1.vendor = "8086"
device.3.1.0.class = "010000"
device.3.1.0.devID = "0030"
device.3.1.0.name = "Symbios Logic Inc. (formerly NCR) LSI Logic Fusion MPT 53C1030 (rev
07)"
device.3.1.0.subsys_devID = "026d"
device.3.1.0.subsys_vendor = "1014"
device.3.1.0.vendor = "1000"
.. the sections below (next 8 lines in boldface) was added by the changes made wih the
.. vmkpcidivy utlity; it causes the service console to share the NICs allocated to the
.. virtual machines
device.esx.1.1.0.owner = "shared"
device.esx.1.2.0.owner = "VM"
device.esx.2.1.0.owner = "VM" <-- not in HS20
device.esx.2.1.1.owner = "VM" <-- not in HS20
device.esx.3.1.0.owner = "shared"
devicenames.001:01.0.nic = "vmnic3" < -- not in HS20; vmnic1 would be the shared NIC
devicenames.001:02.0.nic = "vmnic0"
devicenames.002:01.0.nic = "vmnic1"
devicenames.002:01.1.nic = "vmnic2" < -- not in HS20
devicenames.003:01.0.scsi = "vmhba0"
hypertexting = "true"
.. the six lines below were added manually ...
nicteam.bond0.home_link = "vmnic3" < -- would be vmnic1 on HS20
```

```

nicateam.bond1.home_link = "vmnic1" < -- would be vmnic0 on HS20
nicateam.vmnic0.team = "bond0"
nicateam.vmnic1.team = "bond1" < -- would be bond0 on HS20
nicateam.vmnic2.team = "bond1" < -- not on HS20
nicateam.vmnic3.team = "bond0" < -- not on HS20
swapfile.enable = "yes"
swapfile.filename = "SwapFile.vswp"
swapfile.sizeMB = "5119"
swapfile.volume = "vmhba0:1:0:1"

```

In Example 8-2, the manually added lines are in **boldface**. These entries ensure that the service console shares the NIC team defined above and thus can exploit the high availability capabilities of the GbESM.

Example 8-2 /etc.rc.local file

```

cat /etc/rc.local
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

# BEGINNING_OF_VMWARE_RC_DOT_LOCAL
if ( uname -a | grep -q vmnix ); then
  R="VMware ESX Server 2.5.2"
else
  R="Linux"
fi
arch=$(uname -m)
a="a"
case "$arch" in
  _a*) a="an";;
  _i*) a="an";;
esac

NUMPROC=`egrep -c "^cpu[0-9]+" /proc/stat`
if [ "$NUMPROC" -gt "1" ]; then
  SMP="$NUMPROC-processor "
  if [ "$NUMPROC" = "8" -o "$NUMPROC" = "11" ]; then
    a="an"
  else
    a="a"
  fi
fi

# This will overwrite /etc/issue at every boot. So, make any changes you
# want to make to /etc/issue here or you will lose them when you reboot.
echo "" > /etc/issue
echo "$R" >> /etc/issue
echo "Kernel $(uname -r) on $a $SMP$(uname -m)" >> /etc/issue

cp -f /etc/issue /etc/issue.net
echo >> /etc/issue
# END_OF_VMWARE_RC_DOT_LOCAL
#vmxnet console thru bond0
/etc/rc.d/init.d/network stop
rmmod vmxnet_console
insmod vmxnet_console devName=bond0

```

```
/etc/init.d/network start
mount -a
```

The following examples show the GbESM configuration files to provide High Availability to the virtual machines. There are two examples, one using hot standby and one using trunk failover. Both are annotated to show where the two switches which would be used for High Availability would have different configurations. Also, any of the options shown in other application configurations (for example, Citrix, WebSphere) and others could be added to the examples if needed by the applications running on guest virtual machines.

Example 8-3 illustrates load balancing of two Web servers running in two different virtual machines. Note that some of the internal ports (1,2,3,12) have **hotstan ena** as part of their configuration. These are the only internal ports which would be disabled. The HS40 is configured to use the NICs in slot 12 for our testing; those in slot 13 are also configured but not being used.

Example 8-3 VMware GbESM configuration with Hot Standby

```
script start "Nortel Networks Layer2-7 GbE Switch Module" 4 /**** DO NOT EDIT THIS LINE!
/* Configuration dump taken 23:59:28 Thu Jan 1, 2070
/* Version 20.2.2.6, Base MAC address 00:0f:06:eb:58:00
/c/port EXT4
    pvid 99
/c/12/vlan 1
    def INT1 INT2 INT3 INT4 INT5 INT6 INT7 INT8 INT9 INT10 INT11 INT12 INT13 INT14 EXT1 EXT2
    EXT3
/c/12/vlan 99
    ena
    name "VLAN 99"
    def EXT4
/c/12/stg 1/off
/c/12/stg 1/clear
/c/12/stg 1/add 1 99
/c/12/trunk 1
    ena
    failovr dis
    add EXT2
/c/13/if 1
    ena
    addr 9.42.171.247
    mask 255.255.255.0
    broad 9.42.171.255
/c/13/if 99
    ena
    addr 10.99.0.3
    mask 255.255.255.0
    broad 10.99.0.255
    vlan 99
/c/13/gw 1
    ena
    addr 9.42.171.3
/c/13/vrrp/on
/c/13/vrrp/hotstan enabled
/c/13/vrrp/vr 1
    ena
    prio 101 (default to 100 onsecond switch)
    vrid 10
    if 1
    addr 9.42.171.252
```

```

    track
      ports e
/c/13/vrrp/vr 99
  ena
  prio 101 (default to 100 on second switch)
  vrid 99
  if 99
  addr 10.99.0.1
/c/13/vrrp/group
  ena
  prio 101 (default to 100 on second switch)
  vrid 99
  if 99
  track
    vrs dis
    ifs dis
    ports ena
    l4pts dis
    reals dis
    hsrp dis
    hsrv dis
/c/slb/adv
  direct ena
/c/slb/sync
  prios d
  reals e
  state e
/c/slb/sync/peer 1
  ena
  addr 9.42.171.21 (would have interface address of this switch (.247) on 2nd switch)
/c/slb/real 61
  ena
  rip 9.42.171.169
/c/slb/real 62
  ena
  rip 9.42.171.174
/c/slb/group 1
  metric minmisses
  add 61
  add 62
/c/slb/port INT1
  client ena
  server ena
  hotstan ena
/c/slb/port INT2
  client ena
  server ena
  hotstan ena
/c/slb/port INT3
  hotstan ena
/c/slb/port INT5
  server ena
/c/slb/port INT6
  server ena
/c/slb/port INT7
  client ena
  server ena
/c/slb/port INT8
  server ena
/c/slb/port INT9

```

```

server ena
/c/slb/port INT10
server ena
/c/slb/port INT12
server ena
hotstan ena
/c/slb/port EXT1
client ena
server ena
/c/slb/port EXT2
client ena
/c/slb/port EXT4
intersw ena
/c/slb/virt 1
ena
vip 9.42.171.252
/c/slb/virt 1/service http
group 1
script end /**** DO NOT EDIT THIS LINE!

```

Example 8-4 provides the same kind of failover as Example 8-3 on page 121, but uses Trunk Failover. Note that all internal ports will be disabled when the uplinks fail.

Key configuration differences between the two configurations are marked in **bold**. Note that in the below configuration we also mark those items which would be different in the second switch in the chassis.

Example 8-4 VMware GbESM configuration with Trunk Failover

```

script start "Nortel Networks Layer2-7 GbE Switch Module" 4 /**** DO NOT EDIT THIS LINE!
/* Configuration dump taken 0:38:03 Thu Jan 1, 2070
/* Version 20.2.2.6, Base MAC address 00:0e:62:38:19:00
/c/port EXT4
pvid 99
/c/12/vlan 1
def INT1 INT2 INT3 INT4 INT5 INT6 INT7 INT8 INT9 INT10 INT11 INT12 INT13 INT14 EXT1 EXT2
EXT3
/* note that there is no need for VLAN 99 in this config
/c/12/vlan 99
ena
name "VLAN 99"
def EXT4
/c/12/stg 1/off
/c/12/stg 1/clear
/c/12/stg 1/add 1 99
/c/12/trunk 1
ena
failovr ena
add EXT2
/c/13/if 1
ena
addr 9.42.171.21 (different address in the other switch)
mask 255.255.255.0
broad 9.42.171.255
/* not needed in this config
/c/13/if 99
ena
addr 10.99.0.2 (different address in the other switch)
mask 255.255.255.0
broad 10.99.0.255

```

```

    vlan 99
/c/13/gw 1
    ena
    addr 9.42.171.3
/c/13/vrrp/on
/c/13/vrrp/trnkfo enabled
/c/13/vrrp/vr 1
    ena
    vrid 10
    if 1
    prio 101 (default to 100 in the other switch)
    addr 9.42.171.252
    track
        ports e
/c/slb/adv
    direct ena
/c/slb/sync
    prios d
    reals e
    state e
/c/slb/sync/peer 1
    ena
    addr 9.42.171.247 (would have .21 in the other switch)
/c/slb/real 61
    ena
    rip 9.42.171.169
/c/slb/real 62
    ena
    rip 9.42.171.174
/c/slb/group 1
    add 61
    add 62
/c/slb/port INT1
    client ena
    server ena
/c/slb/port INT2
    client ena
    server ena
/c/slb/port INT5
    server ena
/c/slb/port INT6
    server ena
/c/slb/port INT7
    client ena
    server ena
/c/slb/port INT8
    server ena
/c/slb/port INT9
    server ena
/c/slb/port INT10
    server ena
/c/slb/port INT12
    server ena
/c/slb/port EXT1
    client ena
    server ena
/c/slb/port EXT2
    client ena
/c/slb/virt 1
    ena

```



```
    vip 9.42.171.252
/c/slb/virt 1/service http
    group 1
/
script end /**** DO NOT EDIT THIS LINE!
```

8.2.3 Using multiple VLANs

It is possible to assign different virtual machines to different VLANs if desired. There are several steps involved in this task. Note that assigning the service console to a VLAN is discussed above in 8.2.1, “Preparation” on page 116.

The required steps are:

1. Create the desired VLANs on the GbESM and assign the internal ports where the VMware server is to them. All internal ports have tagging enabled by default and are members of VLAN 1, so if the ports are added to additional VLANs then they will have tags in their frames. In such a case neither virtual machines nor the service console should be configured to use VLAN 1, for example:

```
    /* create vlan 10 and 20
/cfg/12/vlan 10/ena
/cfg/12/vlan 20/ena
/* assign port 12 to vlan 10 and 20
/cfg/12/vlan 10/add INT12
/cfg/12/vlan 20/add INT12
```

2. Create a new Virtual Switch if needed (Figure 8-1 on page 126). Note that the initial virtual switch created by default can be configured to support additional VLANs. Only create a new Virtual Switch if you are going to use physical NICs which are not already in use (such as on an HS40 server or other blade which has more than 2 NICs to each switch module). Note that you must follow steps like those previously shown if you wish to implement failover on these additional NICs.

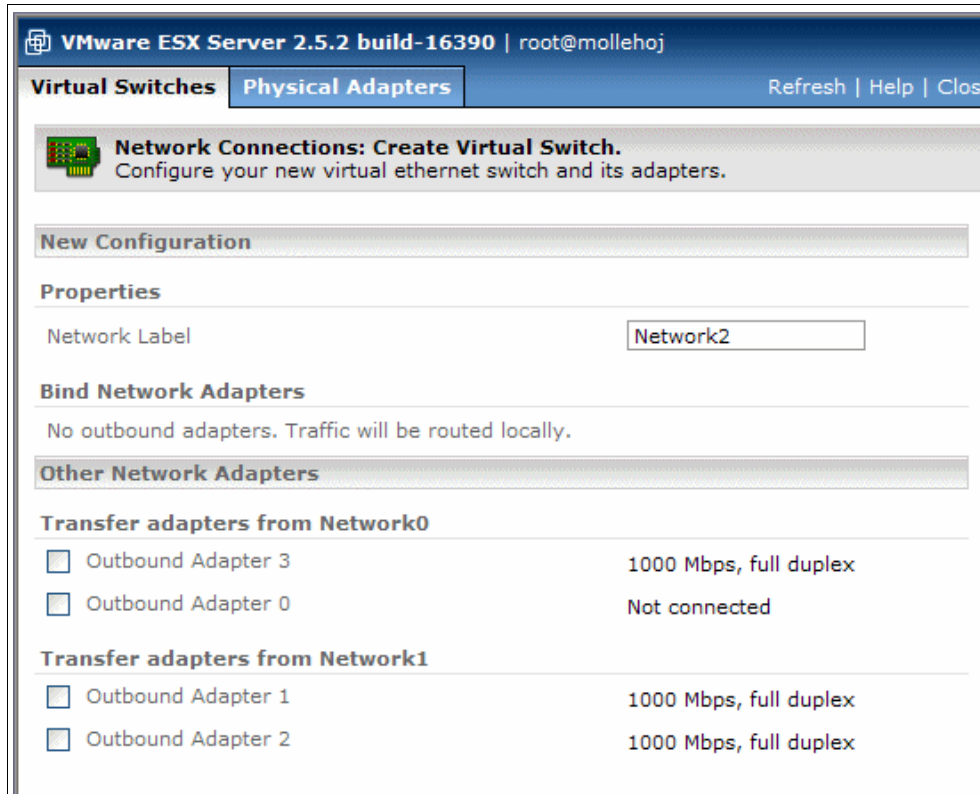


Figure 8-1 Creation of virtual switch

3. Create port groups associated with the virtual switches which use the NIC or bond which you wish to have carry traffic to the virtual machines. In many cases, there will be only one bond available and it should be used rather than any of the NICs to provide High Availability. Each port group has an associated VLAN number which must match the VLAN number configured on the switch (10 or 20 in Figure 8-2). This is done from the Network Connections item under Options on the VMware Management Interface.

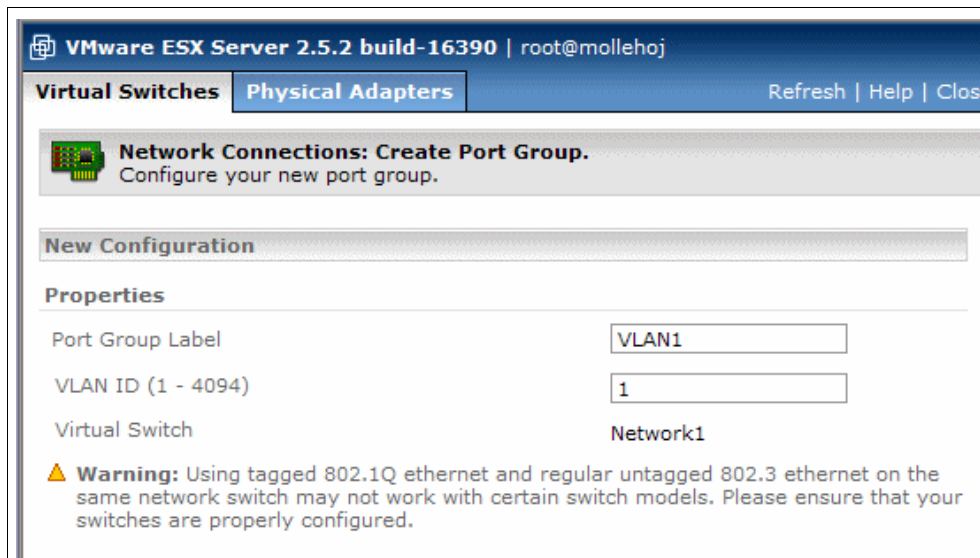


Figure 8-2 Creation of port groups (VLAN)

4. Assign each virtual machine to the appropriate port group (Figure 8-3). Then, configure the network object or /dev/eth<x> on the virtual machines to use an IP address on the subnet associated with the selected VLAN for that virtual machine. This is done by modifying the Network Adapter in the configuration of the individual virtual machine.

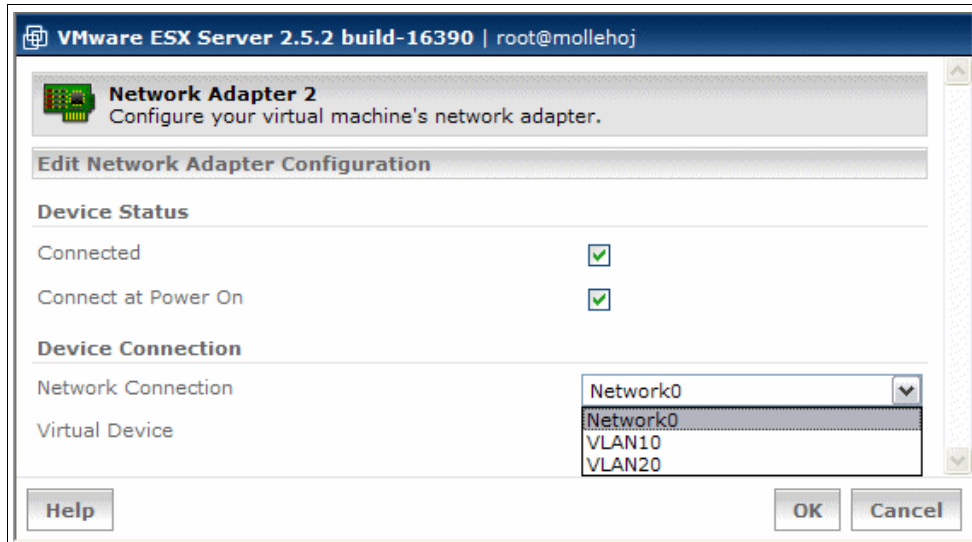


Figure 8-3 Assignment of a guest virtual machine to a port group

5. These steps allow multiple guest Virtual Machines to be assigned to VLANs. The steps shown illustrate assigning each guest VM to one VLAN. Just as with real machines, it is possible to have a VM assigned to multiple VLANs. With real machines, this is done by using a driver which creates additional network objects and assigns them to VLANs on real physical NICs. With guest VMs, the process is different and is managed by VMware: additional virtual NICs are created and added to the guest's configuration and assigned to VLANs using the same graphical tool as is shown in Figure 8-4 on page 128.

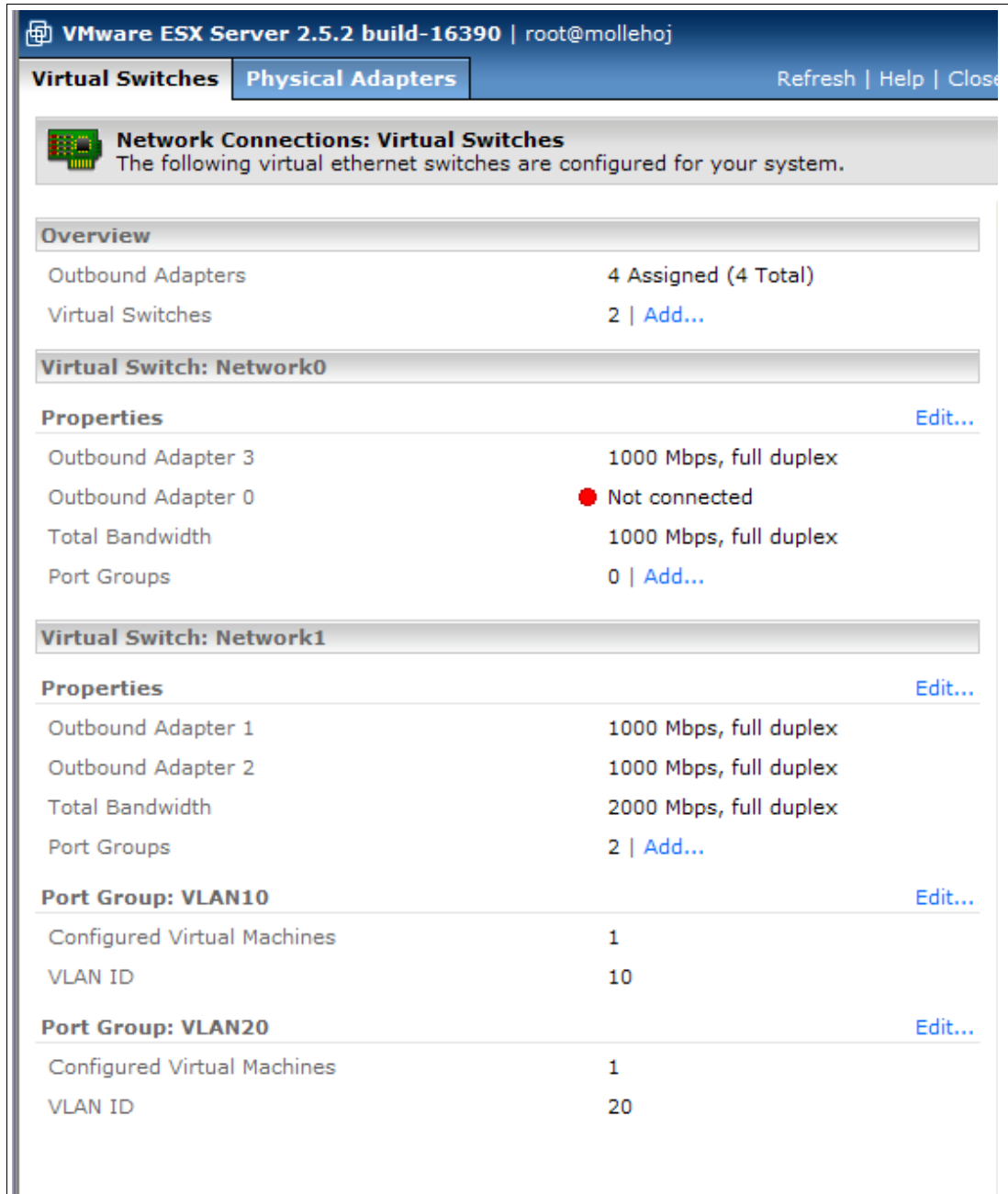


Figure 8-4 Display of Network Connections with two Port Groups

8.2.4 Diagnosis and troubleshooting

This section shows various commands and files which should be inspected as part of a troubleshooting effort.

Where are the virtual servers?

In the examples that follow, the two virtual servers and the service console reside in the same blade and thus all are reachable through port INT12. The virtual server's IP addresses are .169 and .174 and the VMware service console is at .209. Note that the MAC addresses of the guest virtual machines are unique and begin with 00:0c:29, and the service console uses the MAC of the physical hardware NIC. See Example 8-5 on page 129.

Example 8-5 VMware server location information from the GbESM

```
>> Main# /i/s1b/du
Real server state:

61: 9.42.171.169, 00:0c:29:60:ab:81, vlan 1, port INT12, health 3, up
62: 9.42.171.174, 00:0c:29:81:33:c0, vlan 1, port INT12, health 3, up

>> Main# /i/l3/arp/d
IP address  Flags  MAC address  VLAN Port  Referenced SPs
-----
9.42.171.3           00:09:12:48:4d:02  1 EXT2  empty
9.42.171.21         P  00:0e:62:38:19:00  1      1 2
9.42.171.45         00:09:6b:00:18:00  1 INT5  empty
9.42.171.57         00:09:6b:00:b6:d0  1 INT6  empty
9.42.171.63         00:09:6b:00:16:a7  1 INT7  empty
9.42.171.83         00:02:55:4f:10:8a  1 EXT2  empty
9.42.171.85         00:09:6b:00:14:d9  1 INT1  empty
9.42.171.86         00:09:6b:00:13:bb  1 INT2  empty
9.42.171.131        00:11:20:3a:3a:34  1 EXT2  empty
9.42.171.152        00:09:6b:64:a8:33  1 EXT2  empty
9.42.171.156        00:09:6b:00:17:dc  1 INT10 empty
9.42.171.169        00:0c:29:60:ab:81  1 INT12 empty
9.42.171.174        00:0c:29:81:33:c0  1 INT12 empty
9.42.171.204        00:0d:60:30:a3:89  1 EXT2  empty
9.42.171.209        00:50:56:f2:fc:9c  1 INT12 empty
9.42.171.215        00:09:6b:ff:09:23  1 INT9  empty
9.42.171.243        00:09:6b:00:12:61  1 INT8  empty
9.42.171.247        00:0f:06:eb:58:00  1 EXT2  empty
9.42.171.248        P 4 00:00:5e:00:01:02  1      1 2
9.42.171.252        P 4 00:00:5e:00:01:0a  1      1 2
10.42.171.21        P  00:0e:62:38:19:00 4095   1 2
10.99.0.1           P  00:00:5e:00:01:63  99     1 2
10.99.0.2           P  00:0e:62:38:19:00  99     1 2
```

VMware NIC status

VMware provides near real time status of network elements in the `/proc/vmware/net` directory. Subdirectories are created for each bond, and each physical NIC. Note that only `bond0` and `bond1` (Example 8-6) are in use on the test server. The other eight are created by default.

Example 8-6 VMware status files in `/proc/vmware/net`

```
[root@mollehoj net]# pwd
/proc/vmware/net
[root@mollehoj net]# ls
bond0 bond2 bond4 bond6 bond8 stats vmnic1 vmnic3
bond1 bond3 bond5 bond7 bond9 vmnic0 vmnic2
[root@mollehoj net]#
```

The directory for `bond0` contains statistics files for each MAC address which is using the bond. See Example 8-7. Note that the two MAC addresses for the two virtual servers created on the test machine are shown as well as the MAC used by the service console.

Example 8-7 Status files for `bond0`

```
[root@mollehoj bond0]# ls
00:0c:29:60:ab:81 00:0c:29:81:33:c0 00:50:56:f2:fc:9c config stats
```

The configuration of bond0 is shown in Example 8-8. The link status of the various physical NICs in the bond is shown as well as the identification of the configured home link, among other parameters:

Example 8-8 /proc/vmware/net/bond0/config

```
[root@mollehoj bond0]# cat config
VlanHwTxAcce1      Yes
VlanHwRxAcce1      Yes
VlanSwTagging       Yes
PromiscuousAllowed  No
InterruptClustering No
Link state:         Up
Speed:              1000 Mbps, full duplex
Queue:              Running
PCI (bus:slot.func): -1:-1.-1
Minimum Capabilities 0x0
Device Capabilities  0x76b
Maximum Capabilities 0x76b
NICTeamingSlaves:
                   Name      LinkUp  BeaconState
                   vmnic0   No      Off
                   vmnic3   Yes     Off
NICTeamingLoadBalance: Off (HomeLink: vmnic3)
NICTeamingSwitchFailover: Off

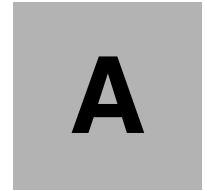
Interrupt vector    0xfffffffffe
DebugSocket         Closed
```

Similar files exist for each physical NIC, such as vmnic3 (which is shared with the service console). Note that the file in Example 8-9 shows that vmnic3 is a member of bond0:

Example 8-9 Status of vmnic3

```
[root@mollehoj vmnic3]# cat config
VlanHwTxAcce1      Yes
VlanHwRxAcce1      Yes
VlanSwTagging       Yes
PromiscuousAllowed  No
InterruptClustering No
Link state:         Up
Speed:              1000 Mbps, full duplex
Queue:              Running
PCI (bus:slot.func): 1:1.0
Minimum Capabilities 0x0
Device Capabilities  0x76b
Maximum Capabilities 0x76b
NICTeamingMaster: bond0
TeamFailoverBeacon: Off

Interrupt vector    0x81
DebugSocket         Closed
```



Filters on L2/3 and L2/7

This appendix presents an example of the command syntax to implement the same filter on both the L2/3 and L2/7 GbESM.

Filter Syntax

The syntax used to specify a filter is different on the two models of GbESM. The syntax for a simple filter - taken from Chapter 6, "Load balancing with WebSphere Portal" on page 75 - is shown below for both platforms. An ultimate, converged syntax for configuring filters has not been defined or scheduled at this point.

L2/7 configuration

This filter allows Telnet access to the destination address 10.10.0.1 only from clients on the subnets 10.1.0.0 and 10.2.0.0. Telnet connections to the destination address from other sources are blocked. All other traffic is allowed.

```
/c/slb/filt 10
  ena
  action allow
  sip 10.1.0.0
  smask 255.255.255.0
  dip 10.10.0.1
  dmask 255.255.255.255
  proto tcp
  dport telnet
  vlan any
/c/slb/filt 20
  ena
  action allow
  sip 10.2.0.0
  smask 255.255.255.0
  dip 10.10.0.1
  dmask 255.255.255.255
  proto tcp
  dport telnet
  vlan any
/c/slb/filt 30
  ena
```

```
action deny
dip 10.10.0.1
dmask 255.255.255.255
proto tcp
dport telnet
vlan any
```

L2/3 Filter Syntax

The same filter would be implemented on the L2/3 switch as follows:

```
/cfg/acl/acl 10/ipv4/sip 10.1.0.0 255.255.255.0
/cfg/acl/acl 10/ipv4/dip 10.10.0.1 255.255.255.255
/cfg/acl/acl 10/tcpudp/dport 23 0xffff
/cfg/acl/acl 10/action permit
/cfg/acl/acl 20/ipv4/sip 10.2.0.0 255.255.255.0
/cfg/acl/acl 20/ipv4/dip 10.10.0.1 255.255.255.255
/cfg/acl/acl 20/tcpudp/dport 23 0xffff
/cfg/acl/acl 20/action permit
/cfg/acl/acl 30/ipv4/dip 10.10.1.1 255.255.255.255
/cfg/acl/acl 30/tcpudp/dport 23 0xffff
/cfg/acl/acl 30/action deny
```




Workaround for VMware use of VLAN tags

This configuration fragment shows the use of VLAN 10 and 20 for the virtual machines. The ports will be configured with PVID (default VLAN) 5 which will not be used by any virtual machines. The physical machine running VMware is installed in slot 12:

```
/cfg/port int12/tag e /* enable tagging */  
  
/cfg/port int12/pvid 5 /* unused, untagged VLAN */  
  
/cfg/l2/vlan 5/add int12  
  
/cfg/l2/vlan 10/add int12  
  
/cfg/l2/vlan 20/add int12
```


Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this Redpaper.

IBM Redbooks

For information about ordering these publications, see “How to get IBM Redbooks” on page 136. Note that some of the documents referenced here may be available in softcopy only.

- ▶ *Nortel Networks L2/3 Ethernet Switch Module for IBM eServer BladeCenter*, REDP-3586-00
- ▶ *IBM eServer BladeCenter Layer 2-7 Network Switching*, REDP-3755-00
- ▶ *IBM eServer BladeCenter Networking Options*, REDP-3660-00

Other publications

These publications are also relevant as further information sources:

- ▶ Nortel Networks Layer 2/7 GbE Switch Module Installation Guide
- ▶ Alteon OS Application Guide for the L2/7 Nortel GbESM
- ▶ Nortel Networks Layer 2/7 GbE Switch Module Application Guide and Command Reference Guide

Online resources

These Web sites and URLs are also relevant as further information sources:

- ▶ Nortel Networks L2/3 Ethernet Switch Module for IBM eServer BladeCenter:
<http://www.redbooks.ibm.com/abstracts/redp3586.html?0open>
- ▶ IBM products
<http://www.ibm.com/products/us/>
- ▶ IBM @server BladeCenter:
<http://www.ibm.com/servers/eserver/bladecenter/index.html>
- ▶ IBM @server BladeCenter storage:
<http://www.pc.ibm.com/us/eserver/xseries/storage.html>
- ▶ BladeCenter advanced server management:
http://www-1.ibm.com/servers/eserver/xseries/systems_management/xseries_sm.html
- ▶ Service Oriented Architecture:
<http://www.ibm.com/SOA>
- ▶ Broadcom NetXtreme Gigabit Ethernet Teaming paper:
<http://www.broadcom.com/collateral/wp/570X-WP100-R.pdf>

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

Numerics

- 10/100/1000 Mbps connections 22
- 1000Base-T 23
- 100BASE-TX 23
- 100-ohm STP 23
- 10BASE-T 23
- 64-bit computing 4
- 802.1D Spanning Tree support 22
- 802.1p Priority Queuing 43
- 802.1P/Q MIB 23
- 802.1Q Tagged 22
- 802.1x Port Authentication 43
- 8677 17

A

- Active X client 107
- advanced server load balancing functions 56
- ANSI interface 12
- ANSI/IEEE 802.3 NWay auto-negotiation 23
- application management 30
- application server 4
- application serving 4
- Application Workload Manager 6
- apply 34
- apply command 34
- architectural limits 23
- ARP cache 33
- authentication method 38
- autosensing 22

B

- backbone 22
- bandwidth 20
- bandwidth metric 59
- BBI 40
- bind request 55
- bind response 55
- blade server 4, 6, 22
- BladeCenter chassis 2
- BladeCenter HS20 8–9
- BOOTP 42
- bootstrap protocol (BOOTP) 23
- Broadcom Advanced Control Suite (BACS) 66
- Broadcom Advanced Services Protocol (BASP) 66
- Broadcom BASP driver 65, 118
- Browser Based Interface 40
- Browser Based Interface (BBI) 32
- browser-smart load balancing 24

C

- cd 34
- chassis 17

- chip cache 16
- Citrix 101, 108
- Citrix MSAM Portal (MSAM) 110
- Citrix Secure Gateway (SG) 110
- Citrix Secure Ticket Authority (STA) 110
- Citrix Web Interface (WI) 110
- CLI 32
 - configuring the switch 32
- Client Processing 61
- CMS 32
- collaboration 3
- command
 - /boot/gtimg 35
 - /boot/reset 36
 - /cfg/ptcfg 36
 - /cfg/sys/access/user 39
 - boot 32, 36
 - cfg 32
 - diff 33
 - help 33
 - maint 33
 - oper 32
 - stats 32
- Command Line Interface (CLI) 30
- comparison to L2/3 switch module 25
- configuration
 - capture 36
- configuration control commands 34
- Configuring Power On Self Test (POST) 32
- connection time-outs 60
- console cable 30
- constraints 63
- content-based load balancing 24
- cookie-based persistence 57
- cookie-based preferential load balancing 24
- copper ports 22
- current configuration
 - capture 36

D

- data traffic 21
- database applications 4
- daughter card 9
- default addresses 19
- default gateway 21
- Default Gateway load balancing 42
- delayed binding 97
- Denial of Service (DoS) 25
- diff 34
- diff flash 34
- Domino 4
- drivers 21
- dynamic host configuration protocol (DHCP) 23

E

- EIA/TIA-568 100-ohm STP 23
- EIA/TIA-568B 100-ohm STP 23
- Electronic Service Agent 6
- enterprise applications 4
- Enterprise Storage Server (ESS) 5
- Equal Cost Multi Path (ECMP) 43
- ERP 4
- EtherLAN interface 8
- Ethernet activity 20
- Ethernet connectivity 17
- Ethernet daughter card 17, 19
- Ethernet interface 8, 29
- Ethernet link 20
- Ethernet management ports 29
- Ethernet module 12
- Ethernet switch error 20
- Ethernet Switch Module 12
- Ethernet switching 17
- exit 34
- Expansion Switch Module 9
- Ext1 17
- external copper ports 22
- external Ethernet interface 20
- external Ethernet ports 30
- external ports 12
 - enable for management 19

F

- fastage 99
- Fibre Channel 5, 9
- Fibre Channel daughter card 9
- File Transfer Protocol (FTP) 35
- file-and-print 3–4
- filter blocks 44
- filter groups 44
- filter-based load balancing 24
- firewalls 56
- firmware 21
 - files 35
 - upgrade 35
- flash memory 22
- forwarding table age time 22
- FTP server health checks 56
- fully qualified domain name (FQDN) 85

G

- general switch information 21
- Gigabit Ethernet path 8
- Gigabit/sec Ethernet (GbE) 2
- Global Load Balancing 78
- Global Server Load Balancing 25
- gting command 32

H

- hash metric 59
- High Availability 64, 70, 116
- High Availability (HA) 25

- history 34
- Hot Standby 65, 68
- HS20 4
- HS20 architecture 10
- HS40 4
- HTTP-based health checks 55
- HTTPS 43
- HTTPS/SSL server health checks 55

I

- I/O Module Tasks 12
- I2C 12, 32
- I2C bus 29
- IBM Director 6, 41
- IBM TotalStorage 5
- ICA Client 101
- ICA connection 103
- identification label 18
- IEEE 802.1Q Tagged VLAN 23
- IEEE 802.3 10BASE-T Ethernet 23
- IEEE 802.3ab 1000BASE-T 23
- IEEE 802.3u 100BASE-TX Fast Ethernet 23
- IEEE 802.3x Full-duplex Flow Control 23
- IEEE 802.3z Gigabit Ethernet 23
- IGMP filtering 43
- IGMP snooping 44
- ikeyman utility 85
- IMAP server health checks 56
- immediate binding 51
- in-band 22
- in-band management 30–31
- insert cookie mode 58
- IntelliStation 6
- Interface 128 18
- interface MIB 23
- internal Ethernet ports 31
- internal full-duplex 10/100 Mbps ports 22
- internal full-duplex gigabit ports 22
- internal network interface 12
- Internet Engineering Task Force (IETF) 2
- Internet Traffic Management (ITM) 47
- Intrusion Detection Servers (IDSs) 56
- IP forwarding per interface 42

J

- Java 2 V1.4 12
- Java applet 12
- Java client 107
- JS20 4
- Jumbo Frame support 45

L

- L2 switching 2
- L2/3 GbESM Switch Module 42
- LACP 2
- layer 2 forwarding 25
- Layer 2-7 GbESM 56
- layer 3 forwarding 25

- layer 4 switching 48
- layer 4-7 switching 24
- Layer 7 capabilities 52
- Layer 7 switching 51
- LDAP health check 55
- leastconns metric 59
- LED 20
- lines 34
- Link Aggregation 2
- link aggregation 20, 22
- link health checks 56
- load balancing 116
- load balancing metric 62
- load balancing metrics 58
- LS20 4

M

- MAC address 18
- Management Information Base (MIB) 41
- management information base (MIB) 23
- Management Module 7, 17, 22, 29, 31, 117
- Management Module Web interface 32, 42
- management network configuration commands 45
- maximum connections 60
- media access control (MAC) 22
- Microsoft
 - Exchange 4
- Midplane 6, 9
- mini-RMON MIB 2, 23
- minmisses metric 59
- mnet command 31
- modular design 4
- Multiple Spanning Tree Protocol (MSTP) 42

N

- navigation commands 34
- Netscape plug-in 107
- NetVista 6
- Network Address Translation (NAT) 24–25, 44, 72
- Network Attached Storage (NAS) 5
- network management 22
- network monitoring 21
- NIC Teaming 66
- NIC teaming 116
- NNTP server health checks 56
- Nortel GbESM 28
- Nortel Networks L2/7 GbESM 30
- Nortel Networks Layer 2/3 GbESM 4
- Nortel Networks Layer 2-7 GbE Switch Modules for IBM Eserver BladeCenter 2

O

- Open Shortest Path First (OSPF) 2
- operator commands 44
- OS image file 35
- out-of-band 22
- out-of-band management 29

P

- passive cookie mode 58
- password
 - change 39
- Peoplesoft 52
- permanent cookie 57
- persistence 56
- ping 34
- POP3 server health checks 56
- popd 34
- port
 - Ext1 17
- Port Aggregation (802.3-ad) teaming 67
- port statistics 21
- power-on self-test (POST) 20
- priority queues 22
- processor blades 17
- protocols 22
- pting command 32
- public routable IP addresses 24
- pushd 34
- pwd 34

Q

- QoS 43
- Quality of Service (QoS) 44, 67
- quit 34

R

- RADIUS 38
- RADIUS server health checks 56
- random-access memory (RAM) 22
- Rapid Spanning Tree Protocol (RSTP) 42
- real IP service 61
- real server 59, 61
- Real Server configuration parameters 60
- real server group 61
- Real-Time Diagnostics 6
- Redbooks Web site 136
 - Contact us x
- redirect 97
- Remote Authentication Dial-in User Service 38
- Remote Deployment Manager (RDM) 6
- remote desktop client 102
- remote management 21
- remote monitoring (RMON) 23
- response metric 59
- revert 34
- revert apply 34
- rewrite cookie mode 58
- RIPv2 43
- roundrobin metric 59
- routing 42
- Routing Information Protocol 2

S

- SAP 52
- save 34

- save command 34
- Scalable Systems Manager 6
- scale-out 4
- script-based health checks 56
- Secure Shell (SSH) 32
- SERDES Gbit Ethernet interface 9
- SERDES-based Gb Ethernet interface 8
- serial number 18
- serial port 30
- server consolidation 3
- server health checking 54
- Server Load Balancing 25, 42
- Server Load Balancing (SLB) 47
- Server Plus Pack 6
- server processing 61
- ServerGuide 6
- ServerGuide Scripting Toolkit 6
- servers 22
- Service Oriented Architecture (SOA) 76
- shopping cart 57
- Siebel 52
- simple network management protocol (SNMP) 22
- slowage 99
- SMP 4
- SMTP server health checks 56
- SNMP 21
- SNMP Management Information Base (MIB) files 41
- SNMP-based management systems 41
- SNMPv3 43
- Software Distribution Premium Edition 6
- Source IP Address Binding 56
- Spanning Tree 2
- Spanning Tree Protocol 23
- SSL session tracking 56
- standards 22
- static ARP 45
- stations 22
- status LEDs 19
- storage 4
- Storage Area Networks (SAN) 5
- storage solutions 5
- store-and-forward 22
- Switch ASIC 16
- switch information panel 40
- switch IP address 42
- switch maintenance 21
- switch management 21–22
- switch module 7
- switch module console port 32
- switch parameters 21
- switch TCP/IP address 21
- SYN-ACK packet 49

T

- TACACS 38
- Tape Drive Management Assistant 6
- TCP handshaking 51
- TCP SYN 51
- TCP SYN-ACK 50
- Telnet 32

- telnet 34
- Telnet client 12
- telnet interface 21
- telnet remote console 23
- temporary cookie 58
- terminal emulation 32
- terminal emulator 36
- Terminal Server 108
- TFTP server 35–36
- ThinkPad 6
- timezone command 44
- Tivoli 2, 41
- tmout 99
- traceroute 34
- transmission method 22
- traps 21
- Trivial File Transfer Protocol (TFTP) 35
- trivial file transfer protocol (TFTP) 22
- Trunk Failover 65, 68, 108
- trunk hashing 45

U

- UDP-based DNS health checks 55
- unbind request 55
- up 34
- UpdateXpress 6
- URL-based load balancing 24
- user accounts 37
- User guides 21
- UTP Category 3 23
- UTP Category 4 23
- UTP Category 5 23
- UTP Category 5e 23

V

- verbose 34
- virtual console 117
- virtual IP 61
- Virtual IP address (VIP) 24
- virtual local area network (VLAN) 22
- virtual machines 115
- Virtual Router Redundancy Protocol (VRRP) 2, 69
- virtual server 61
- virtual server-based load balancing 24
- Virtual Service configuration parameters 61
- Virtual Service Pool 58
- virtual service pools 24
- VLAN
 - 4095 17
- VLAN tagging 2
- VMware ESX 115
- VRRP 65
- VRRP configuration 69
- VRRP priority 69

W

- WAP gateway health checks 55
- Web browser 21

Web Interface Server 107
Web server 4
Web services 52
Web-based management 23
WebSphere Portal 77
weights 59
who 34
Wireless Session Protocol (WSP) 55
Wireless Transport Layer Security (WTLS) 55

X

XML 52
XpandonDemand 4
xSeries 6



Application Switching with Nortel Networks Layer 2-7 Gigabit Ethernet Switch Module for IBM BladeCenter

Experience the value of using the Nortel GbESM for BladeCenter

This IBM® Redpaper positions the Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter and describes how its integrated switch options enable the consolidation of full Layer 2-7 LAN switching and routing capabilities.

Experience new tools, techniques, and applications to manage and deploy the Nortel GbESM

This Redpaper serves as an update to the IBM @server BladeCenter® Layer 2-7 Network Switching, REDP-3755-00. Here, we provide more discussion on the Nortel Networks Layer 2-7 GbE Switch Modules for IBM @server BladeCenter set of features and services. In particular, we discuss the L2-7 GbESM being a fully-functioning content and load balancing switch with capabilities equivalent to products offered as free-standing appliances. Load balancing using the Nortel GbESM was demonstrated (but not limited to) utilizing the following applications and virtual machines: Citrix MetaFrame/Terminal Services, IBM WebSphere® Application Server/IBM WebSphere Portal, and VMware ESX. However, note these applications were used as examples.

Experience the Server Load Balancing capabilities of the Nortel GbESM

In this Redpaper, we discuss tools, techniques, and applications that help with the management and deployment of the Nortel GbESM in an IBM BladeCenter. We also discuss the management paths and rules for connecting to and accessing the Nortel GbESM.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks