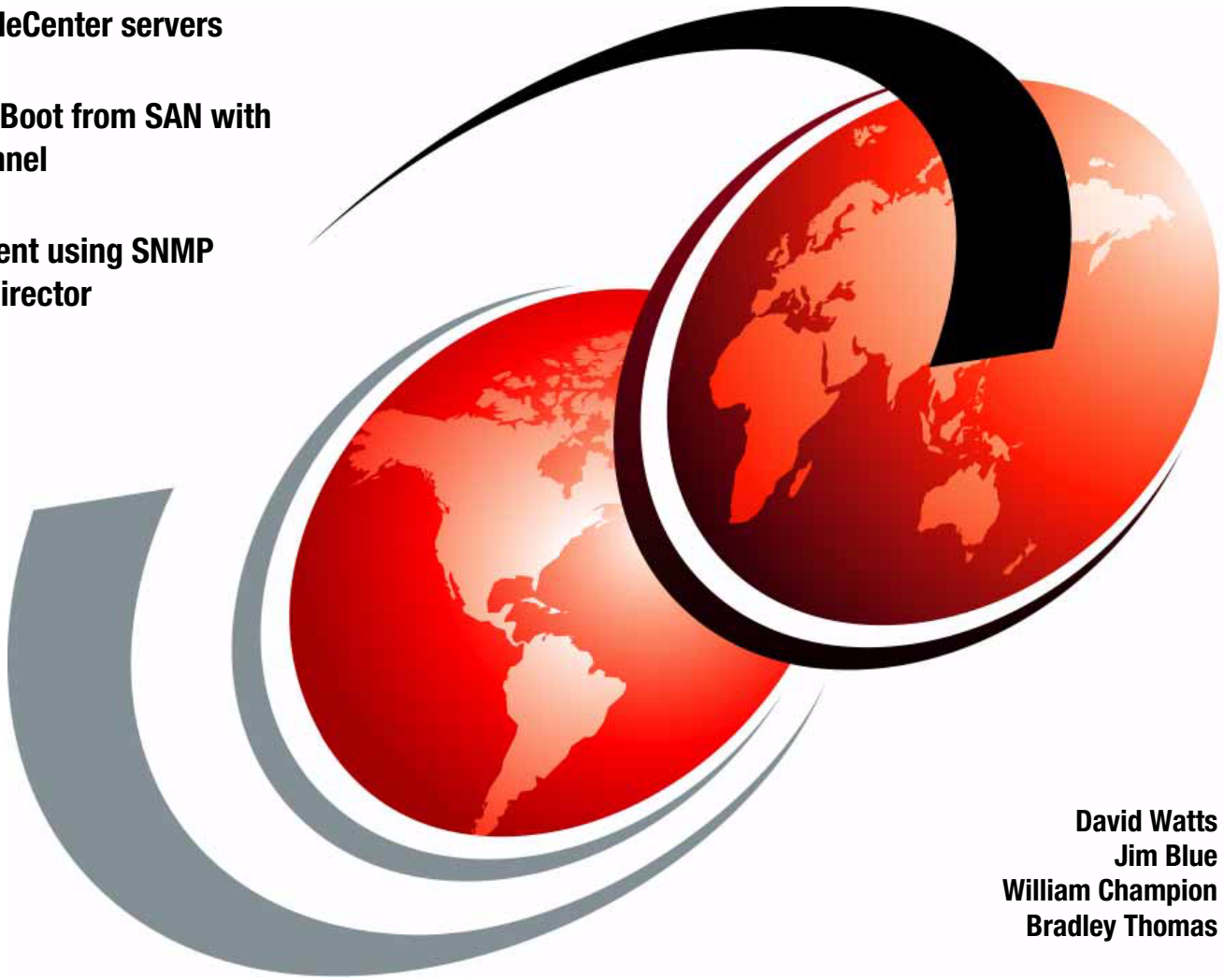
IBM

Implementing Sun Solaris on IBM BladeCenter Servers

- Install and deploy Solaris 10 on the latest BladeCenter servers
- Configure Boot from SAN with Fibre Channel
- Management using SNMP and IBM Director



David Watts
Jim Blue
William Champion
Bradley Thomas

Redpaper

ibm.com/redbooks

International Technical Support Organization

Implementing Sun Solaris on IBM BladeCenter Servers

May 2007

REDP-4269-00

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

First Edition (May 2007)

This edition applies to Sun Solaris 10 11/06 on IBM BladeCenter servers.

Contents

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 illustrate 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.

Trademarks

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

Redbooks (logo) ®
BladeCenter®
DB2®
DS4000™
DS8000™
Informix®
IBM®
NetView®
PowerExecutive™
Predictive Failure Analysis®
Purify®
POWER5™
Rational®
Redbooks®
ServerGuide™
ServeRAID™
System x™
System Storage™
Tivoli Enterprise™
Tivoli Enterprise Console®
Tivoli®
TotalStorage®
Virtualization Engine™
WebSphere®

The following terms are trademarks of other companies:

SAP, and SAP logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

QLogic, and the QLogic logo are registered trademarks of QLogic Corporation. SANblade is a registered trademark in the United States.

Oracle, JD Edwards, PeopleSoft, Siebel, and TopLink are registered trademarks of Oracle Corporation and/or its affiliates.

IT Infrastructure Library, IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

InfiniBand, and the InfiniBand design marks are trademarks and/or service marks of the InfiniBand Trade Association.

Advanced Micro Devices, AMD, AMD Opteron, AMD Virtualization, AMD-V, ATI, HyperTransport, the AMD Arrow logo, and combinations thereof, are trademarks of Advanced Micro Devices, Inc.

Java, JavaScript, JumpStart, JVM, OpenSolaris, Solaris, Sun, Sun BluePrints, SunOS, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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

Intel, Xeon, 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.

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.

Preface

The Solaris™ operating system from Sun™ Microsystems is an important platform for many IBM® customers. Solaris is supported on many of our blade servers and options, as well as servers in our System x™ product line.

This IBM Redpaper describes how to install Solaris on supported BladeCenter® servers, either natively or with the use of a Solaris Installation Server. We describe how to incorporate the latest patches to Solaris from Sun, plus updated drivers for the Ethernet and RAID devices in the blade servers. We explain how to implement Fibre Channel storage and how to configure *boot from SAN*. We also show how to integrate blade servers running Solaris into an IBM Director or SNMP-based management infrastructure.

This paper is aimed toward IBM customers who are already familiar with the use of Solaris on platforms other than the IBM BladeCenter solution.

The team that wrote this IBM Redpaper

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

David Watts is a Consulting IT Specialist at the IBM ITSO Center in Raleigh. He manages residencies and produces IBM Redbooks® publications on hardware and software topics related to IBM System x and BladeCenter servers and associated client platforms. He has authored over 80 books, papers and technotes. He holds a bachelor's degree of Engineering from the University of Queensland (Australia) and has worked for IBM both in the U.S.A. and Australia since 1989. He is an IBM Certified IT Specialist.

Jim Blue is an Advisory Software Engineer in Research Triangle Park, North Carolina, and is currently a member of the SAN Central Support Group. He has 26 years of experience in the areas of IT and networking. He holds a bachelor's degree in Computer Science from North Carolina State University. His area of expertise includes numerous networking protocols. He is a subject matter expert for a number of IBM TotalStorage® certification exams (High-end disk, Open Systems Storage, High-end Tape and Storage Networking Solutions) as well as the Storage Networking Industry Association (SNIA) foundation exam.

William Champion is a Staff Hardware Engineer in Research Triangle Park, North Carolina, and is currently a member of ATS/BISC Americas. He has six years of networking experience working in the IBM Networking Hardware Division. He has managed and designed three large raised floor labs. He currently has the following roles in ATS/BISC: Rack and Power RDS (Regional Designated Specialist) Lead, Linux® RDS, BladeCenter RDS, and Solaris SME (Subject Matter Expert). He holds a bachelor's degree in Electrical Engineering from North Carolina State University.

Bradley Thomas is a Staff Software Engineer in Research Triangle Park, North Carolina, and is currently a a member of the NOS Technology and Support Team. He has seven years of experience in operating system support covering Caldera Linux, SUSE Linux, Red Hat Linux, VMware, and Solaris. He holds bachelor's degree in Computer Engineering from Auburn University. His current focus is specializing in the Solaris and Red Hat operating systems as well as supporting several hardware platforms. Bradley is the author of the many of the IBM installation instructions for Solaris, and Red Hat Linux.

The team (l-r): Jim, Bill, Bradley, and David

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

From the International Technical Support Organization, Raleigh Center:

- Carolyn Briscoe
- Linda Robinson
- Erica Wazweski

From IBM:

- Khalid Ansari
- Eric Brown
- Donald Champion
- Martin Comeau
- Alan Darlington
- Duncan Furniss
- Robert Jakes
- Ishan Sehgal
- Buddy Tapp

From Sun Microsystems:

- Mike Barrett
- Lee Damico
- Brian Dowdy
- Linda Flanagan
- Menno Lageman
- Todd Tornga

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbooks publication dealing with specific products or solutions, while getting hands-on experience with

leading-edge technologies. You will have the opportunity to team with IBM technical professionals, Business Partners, and Clients.

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 IBM Redpapers to be as helpful as possible. Send us your comments about this IBM Redpaper or other Redbooks in one of the following ways:

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

  ibm.com/redbooks

- Send your comments in an e-mail to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

1

Technical overview

In this chapter, we introduce Solaris 10 and the BladeCenter chassis and servers. Topics in this chapter are:

Notice: Notice: This chapter contains Sun Microsystems, Inc. copyrighted materials. Any reproduction or use of the material requires express permission from Sun Microsystems, Inc.

1.1 Solaris 10 overview

The Solaris operating system is designed in a modular fashion so that it can adapt to new processor and hardware platforms while incorporating new features. It is also designed to provide nondisruptive growth and evolution by enabling new services to be added on top of a stable core. The Solaris kernel is dynamic, composed of a core system that is always resident in memory, with services beyond the core loaded as needed. For customers, this means that Solaris software can evolve to accommodate new devices and services without even rebooting — resulting in less downtime and greater agility to meet the changing business needs.

The Solaris kernel is a compact code base that is built to be extended. Unlike other operating environments where the delineation between the operating system and applications is confused, Sun designed the Solaris operating system so that the line is very clear. It is nearly impossible, for example, for a Web browser to *crash* (stop or lock up) the operating system because Solaris software can distinguish between an application and an operating system feature. This clarity enables a smaller code base that results in a more reliable and secure operating environment.

1.1.1 Solaris Containers

Note: This section is an excerpt from the Sun BluePrints™ paper, *Solaris Containers — What They Are and How to Use Them* by Menno Lageman of Sun Microsystems, Inc. The paper can be downloaded from:

`http://www.sun.com/blueprints/0505/819-2679.pdf`

Today, businesses often design their systems with extra capacity to handle occasional peak loads to maximize revenue during periods of high demand. This extra system capacity remains unused during periods of normal demand. By allowing other applications to borrow this unused capacity a more cost-effective solution can be realized. During periods of high demand resources can be dynamically reallocated to important applications. Sharing resources in this way leads to higher resource utilization, reduces capital and system management costs by reducing the total number of systems required. For the consolidation of applications onto fewer systems to be effective, applications must be able to be managed independently. This requires the ability to control resource utilization, isolate faults, and manage security between multiple applications on the same server. In other words, it requires the establishment of virtual server boundaries within the server.

One of the first steps in this direction was the introduction of Dynamic System Domains on large Sun servers. With Dynamic System Domains, a server can be divided into several domains, each running its own copy of the Solaris operating system. The domains provide hardware isolation between the applications so that faults in one domain do not propagate to applications in other domains. Domain boundaries can be dynamically partitioned to adapt to changing resource requirements. Resources can be moved from one domain to another without requiring a restart of the system. This adds flexibility to the data center while maintaining security and isolation from faults in other domains.

Starting with Solaris Resource Manager 1.x in the Solaris 2.6 OS, Sun has gradually enhanced the ability to control resource utilization and separate applications running in a single instance of the Solaris operating system. Several technologies have been added to the Solaris operating system over the years, providing additional capabilities and finer control over resource utilization. Examples of such technologies include the Solaris 9 Resource

Manager and Resource Pools in the Solaris 9 OS. These technologies allow users to create a Solaris Container, an application or service that has one or more resource boundaries associated with it. These resource boundaries can limit CPU or memory consumption, network bandwidth, or even be a processor set. As a result, Solaris Containers are a prime enabler for server consolidation.

With the introduction of Solaris Zones in the Solaris 10 OS, Sun is taking Solaris Containers a step further by allowing servers to be partitioned in sub-CPU granularity. A Solaris Zone is a complete execution environment for a set of software services — a separate, virtual Solaris environment within a Solaris instance. A Zone provides a virtual mapping from software services to platform resources, and allows application components to be isolated from each other even though they share a single Solaris operating system instance. It establishes boundaries for resource consumption and provides isolation from other Zones on the same system. The boundaries can be changed dynamically to adapt to changing processing requirements of the applications running in the Zone.

Solaris Containers can be built using one or more the following technologies. These technologies can be combined to create Containers tailored for a specific server consolidation project.

- Solaris Resource Manager, for workload resource management
- Resource Pools, for partitioning
- Zones, for isolation, security and virtualization

It is important to note that a Solaris Container is not equivalent to a Solaris Zone. Zones technology can be used to create a Container with certain characteristics, such as the isolation provided by the virtual Solaris environment. But it is also possible to create another Solaris Container using Resource Pools technology if the required characteristic of that Container can be met with the features Resource Pools provide. So while a Zone is a Container, a Container is not necessarily a Zone.

1.1.2 DTrace

DTrace is a comprehensive dynamic tracing facility that is built into Solaris and can be used by administrators and developers to examine the behavior of both user programs and of the operating system itself. With DTrace you can explore your system to understand how it works, track down performance problems across many layers of software, or locate the cause of aberrant behavior. It is safe to use on production systems and does not require restarting either the system or applications.

DTrace dynamically modifies the operating system kernel and user processes to record data at locations of interest, called probes. A probe is a location or activity to which DTrace can bind a request to perform a set of actions, like recording a stack trace, a time stamp, or the argument to a function. Probes are like programmable sensors scattered all over your Solaris system in interesting places. DTrace probes come from a set of kernel modules called providers, each of which performs a particular kind of instrumentation to create probes.

DTrace includes a new scripting language called D which is designed specifically for dynamic tracing. With D it is easy to write scripts that dynamically turn on probes, collect the information, and process it. D scripts make it convenient for users to share their knowledge and troubleshooting methods with others. A number of useful D scripts are included in Solaris 10, and more can be found on the following:

- Sun's BigAdmin site: http://sun.com/bigadmin/content/dtrace/
- OpenSolaris™ project site: http://opensolaris.org/os/community/dtrace/

1.1.3 Predictive Self-Healing

Sun has developed a new architecture for building and deploying systems and services capable of *Predictive Self-Healing*. Self-healing technology enables Sun systems and services to maximize availability in the face of software and hardware faults. It facilitates a simpler and more effective end-to-end experience for system administrators, reducing cost of ownership.

The first self-healing features are available as part of the Solaris 10 Operating System. This paper describes Sun's approach to building self-healing systems as well as the new technologies available in the Solaris 10 OS. For more information, visit

`http://sun.com/solaris/10`

Solaris 10 Predictive Self-Healing features:

- Automatic monitoring and diagnosis of CPU, memory, and I/O subsystems
- Automatically take faulty resources offline while the Solaris operating system is running
- Administrator tools to view self-healing logs and results
- Standardized messaging for all self-healing diagnosis results
- Knowledge article Web site linking to online diagnosis messages

Solaris 10 Predictive Self-Healing benefits:

- Improved system and service availability through predictive diagnosis and isolation of faulty components
- Diagnosis of faulty components performed automatically, in some cases reducing analysis time from days to seconds
- Simplified administration model for managing self-healing activities, reducing cost of ownership
- Links to knowledge articles for learning more about problem impacts and repairs, updated in Internet time
- Scalable architecture that can be rapidly adapted to new problems and updated without requiring system downtime

1.1.4 ZFS

Traditional file systems are made up of data structures overlaid onto a disk volume that is managed independently by a separate volume manager. This approach involves configuring storage devices, virtualizing multiple devices into logical volumes, and finally laying out a file system structure onto them. Each layer must be managed consistently with every other layer, making storage difficult to manage and configurations prone to error. Even when everything goes right, the disk space can be underutilized because each file system must be configured from the beginning to handle an anticipated maximum amount of data

Sun's Solaris ZFS fundamentally changes the storage equation by integrating devices, storage, and file system structures into a single structure. By integrating the file system with volume management, the risk of misconfigurations at one layer affecting another layer is virtually eliminated. Sun's file services technology results in more effective use of storage and provides a high degree of reliability and flexibility that can help reduce cost, complexity, and risk.

ZFS Highlights:

- 128-bit file system 16 billion billion times the capacity of 32- or 64-bit file system
- All data is protected by 64-bit checksums, consistency of data is maintained at all times

- Proven and cutting edge technologies combine to optimize performance
- Eliminates the need for a volume manager, GUI automates administrator intent
- POSIX compliant applications run without modification

Solaris ZFS is built to allow file systems to be created on an as-needed basis, so administrators can create a file system per user or a file system per application, as needs dictate. It breaks the bond of a single file system per volume, allowing as many file systems as desired to be allocated from the single pool of storage. This allows more efficient use of storage because different file systems can draw storage from the pool without each one having its own internal fragmentation. This is the perfect complement to Solaris Containers, making it easy for organizations to deploy many Containers with as many file systems as needed.

1.2 BladeCenter overview

Blade servers are thin servers that insert into a single rack-mounted chassis which supplies shared power, cooling, and networking infrastructure. Each server is an independent server with its own processors, memory, storage, network controllers, operating system, and applications. Blade servers came to market around 2000, initially to meet clients' needs for greater ease of administration and increased server density in the data center environment.

When IBM released the IBM BladeCenter in 2002, it quickly changed the industry with its modular design. The IBM BladeCenter provides complete redundancy in a chassis, and enables network and storage integration.

There are four chassis in the BladeCenter family:

- IBM BladeCenter provides the greatest density and common fabric support and is the lowest entry cost option. To eliminate any confusion, this chassis is also known as IBM BladeCenter 8677 or IBM BladeCenter Enterprise.
- IBM BladeCenter H delivers high performance, extreme reliability, and ultimate flexibility for the most demanding IT environments.
- IBM BladeCenter T models are designed specifically for telecommunications network infrastructures and other rugged environments.
- IBM BladeCenter HT models are a combination of the designs of the H and T chassis - designed specifically for telecommunications and rugged environments plus the support of high-speed networking such as 4X InfiniBand®.

All four chassis share a common set of blades and switch modules. The only exceptions to this are:

- The oldest HS20 blade server, machine type 8678 is no supported in the BladeCenter H and BladeCenter T chassis.
- The new high-speed switch modules are not supported in the BladeCenter 8677 and BladeCenter T chassis.

Note: We will be covering the BladeCenter H and BladeCenter HT chassis in detail in this chapter.

1.2.1 BladeCenter H chassis

IBM BladeCenter H delivers high performance, extreme reliability, and ultimate flexibility to even the most demanding IT environments. In 9 U of rack space, the BladeCenter H chassis can contain up to 14 blade servers, 10 switch modules, and four power supplies to provide the necessary I/O network switching, power, cooling, and control panel information to support the individual servers.

The chassis supports up to four traditional fabrics using networking switches, storage switches, or pass through devices. The chassis also supports up to four high-speed fabrics for support of protocols like 4X InfiniBand or 10 Gigabit Ethernet. The built-in media tray includes light path diagnostics, two front USB inputs, and a DVD drive.

Figure 1-1 and Figure 1-2 on page 7 display the front view of an IBM BladeCenter H.

Figure 1-1 BladeCenter H front view

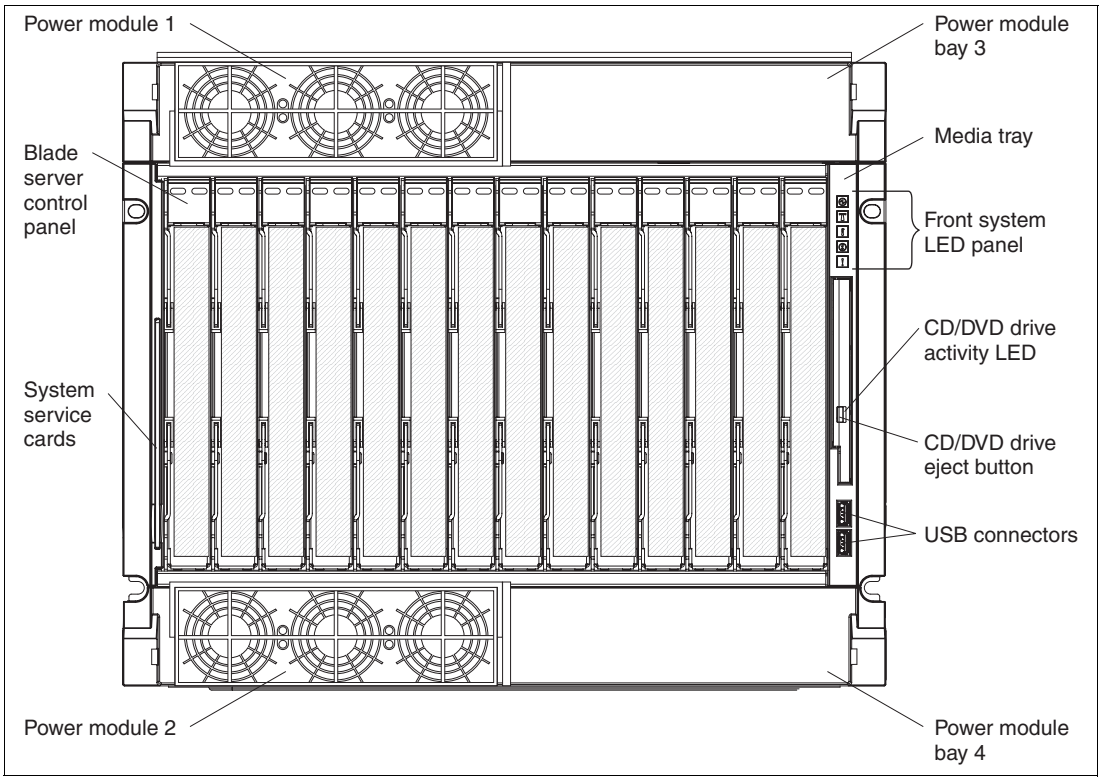


Figure 1-2 Diagram of BladeCenter H front view showing the key features of the BladeCenter H chassis

The following are the key features on the front of the BladeCenter H:

- A media tray at the front right, with a DVD drive, two USB V2.0 ports, and system status LED panel
- One pair of 2,900-watt power modules. An additional power module option (containing two 2,900 W power modules) is available
- Two hot swap fan modules (two extra hot swap fan modules are included with the additional power module option).
- 14 hot swap blade server bays supporting different blade server types:

Figure 1-3 BladeCenter H rear view

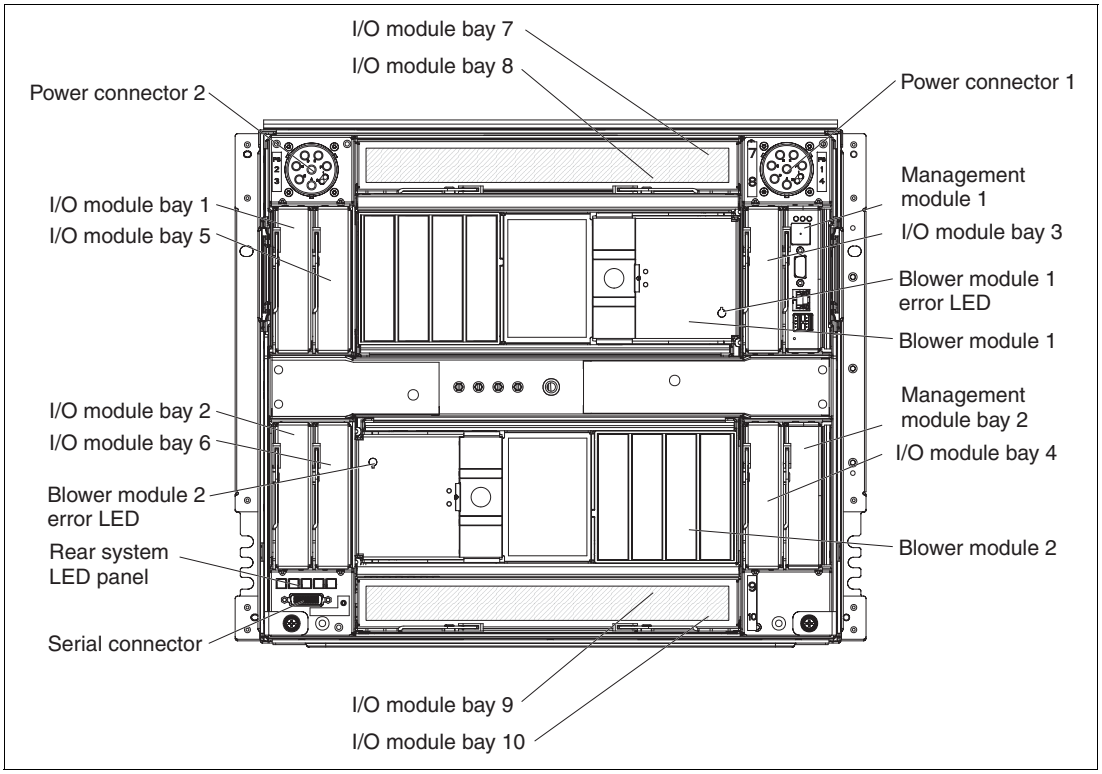


Figure 1-4 Diagram of BladeCenter H rear view showing the key features of the BladeCenter H chassis

The BladeCenter H chassis allows for either 14 single-slot blade servers or seven double-slot blade servers. However, you can mix different blade server models in one chassis to meet your requirements.

The BladeCenter H chassis ships standard with one Advanced Management Module. This module provides the ability to manage the chassis and well as providing the local KVM function. The optional redundant Advanced Management Module provides the IBM BladeCenter H with higher levels of resiliency. This module provides administrators with easy remote management and connectivity to the BladeCenter H chassis.

The BladeCenter has bays for 10 hot-swap I/O modules. You choose these I/O modules based on your connectivity needs. An Ethernet Switch Module (ESM) will be required in I/O module bays 1 and 2, to enable the use of both Ethernet ports on a Blade Server. The I/O modules required in I/O module bays 3 and 4 will depend on the I/O Expansion Card installed in the blade servers. The I/O modules required in the high speed I/O module bays 7, 8, 9 and 10 will depend on the I/O Host Channel Adapter cards installed in the Blade Servers.

1.2.2 BladeCenter HT chassis

The IBM BladeCenter HT is a new 12-server blade chassis designed for high-density server installations, typically for telecommunications use. It offers high performance with the support of 10G Ethernet installations. This 12U high chassis with DC or AC power supplies provides a cost-effective, high-performance, high-availability solution for telecommunication network and other “rugged” non-telco environments. The IBM BladeCenter HT chassis is positioned for expansion, capacity, redundancy, and carrier-grade NEBS level/ETSI compliance in DC models.

BladeCenter HT provides a solid foundation for next-generation networks (NGN) enabling service providers to become on demand providers. Coupled with technological expertise within the enterprise data center, IBM is leveraging the industry know-how of key Business Partners to jointly deliver added value within service provider networks.

Figure 1-5 on page 10 and Figure 1-6 on page 11 show the front view of the BladeCenter HT.

Figure 1-5 BladeCenter HT front view

The BladeCenter HT brings significant new capabilities to IBM's broad ecosystem of hundreds of NGN applications already being deployed on BladeCenter. A key example is the introduction of the Nortel 10 Gb Ethernet Switch Module for BladeCenter, which delivers 10 Gb to each blade server deployed in the BladeCenter H or BladeCenter HT chassis, and six 10 Gb Ethernet uplinks. This capability helps greatly reduce the cost of implementing IPTV and other high bandwidth NGN applications.

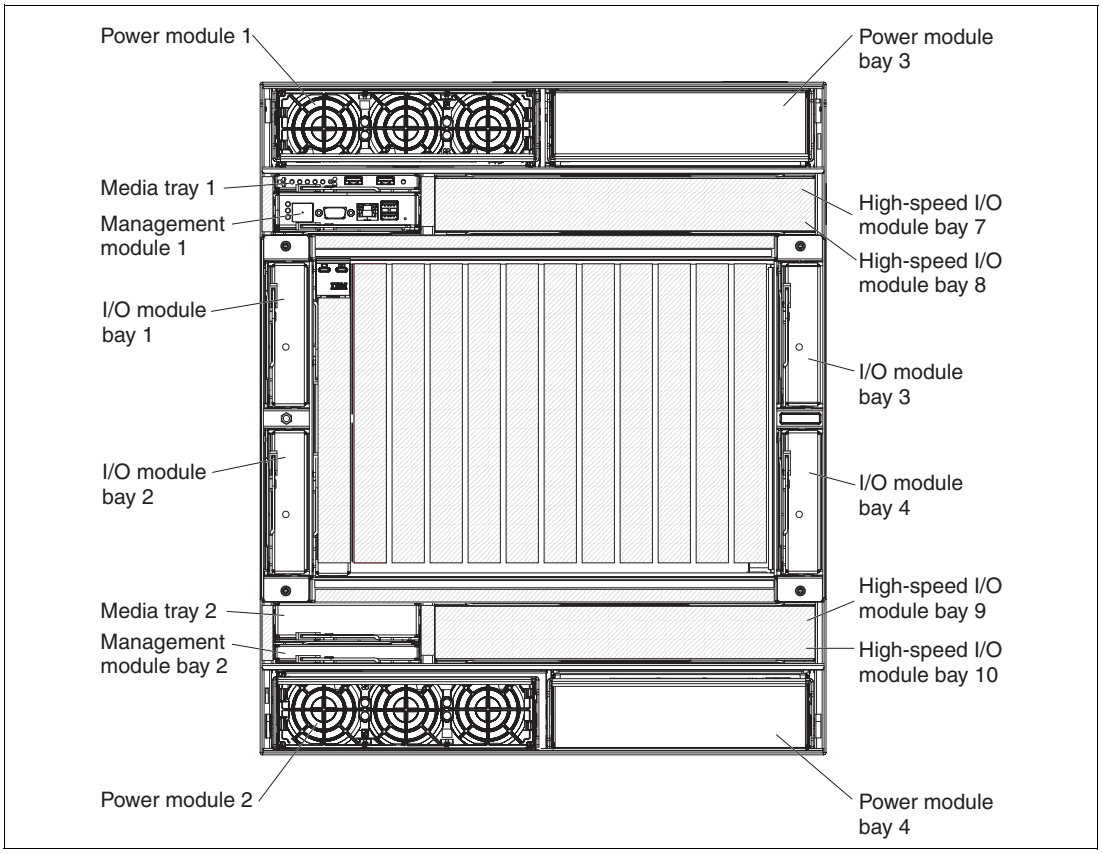


Figure 1-6 The IBM BladeCenter HT chassis

The key features of the BladeCenter HT:

- Support for up to 12 blade servers, compatible with the other chassis in the BladeCenter family
- Four low-speed and four high-speed I/O module bays, compatible with the other chassis in the BladeCenter family
- A media tray at the front with light path diagnostics, two USB 2.0 ports, and optional compact flash memory module support
- Two hot swap management-module bays (one management module standard)
- Four hot swap power-module bays (two power modules standard)
- New serial port for direct serial connection to installed blades
- Compliance with the NEBS 3 and ETSI core network specifications

Figure 1-7 on page 12 and Figure 1-8 on page 13 show the rear view of the BladeCenter HT.

Figure 1-7 BladeCenter HT rear view

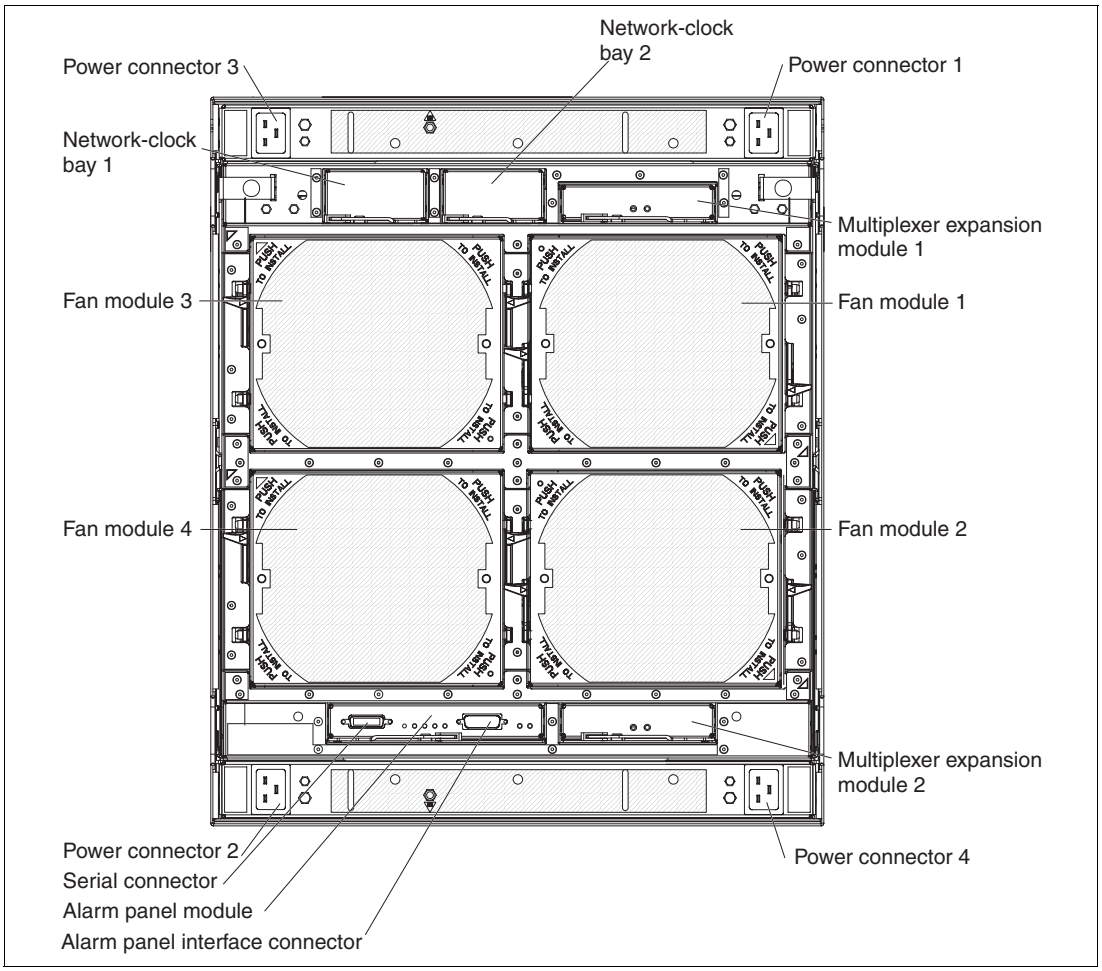


Figure 1-8 Diagram of BladeCenter HT rear view

BladeCenter HT delivers rich telecom features and functionality, including integrated servers, storage and networking, fault-tolerant features, optional hot swappable redundant DC or AC power supplies and cooling, and built-in system management resources. The result is a Network Equipment Building Systems (NEBS-3) and ETSI-compliant server platform optimized for next-generation networks.

The BladeCenter HT applications suited for these servers include the following:

- Network management and security
  - Network management engine
  - Internet cache engine
  - RSA encryption
  - Gateways
  - Intrusion detection
- Network infrastructure
  - Softswitch
  - Unified messaging
  - Gateway/Gatekeeper/SS7 solutions
  - VOIP services and processing
  - Voice portals
  - IP translation database

The BladeCenter T chassis allows for either 12 single-slot blade servers or six double-slot blade servers. However, you can mix different blade server models in one chassis.

The BladeCenter HT chassis ships standard with one Advanced Management Module. This module provides the ability to manage the chassis and well as providing the local KVM function. The optional redundant Advanced Management Module provides the IBM BladeCenter HT with higher levels of resiliency. This module provides clients with easy remote management and connectivity to the chassis.

The BladeCenter HT does not ship standard with any I/O modules. You need to choose these I/O modules depending on your connectivity needs. An Ethernet Switch Module (ESM) will be required in I/O module bays 1 and 2, to enable the use of both Ethernet ports on a Blade Server. The I/O modules required in I/O module bays 3 and 4 will depend on the I/O Expansion Card installed in the blade servers.

1.2.3 Blade servers

IBM BladeCenter servers supports a wide selection of processor technologies and operating systems to allow clients to run all of their diverse work loads inside a single architecture. The slim, hot-swappable blade servers fit in a single chassis like books in a bookshelf, and each is an independent server, with its own processors, memory, storage, network controllers, operating system and applications. The blade server simply slides into a bay in the chassis and plugs into a midplane or backplane, sharing power, fans, diskette drives, switches, and ports with other blade servers.

The benefits of the blade approach will be obvious to anyone tasked with running down hundreds of cables strung through racks just to add and remove servers. With switches and power units shared, precious space is freed up – and blade servers enable higher density with far greater ease.

Table 1-1 shows which blade servers support the installation of Solaris 10. For those that are supported or planned to be supported, IBM supplies installation instructions. See 2.1, “Installation” on page 30.

Table 1-1 Blade servers and Solaris support information

Blade server	Machine type	Processor	Supports Solaris 10
HS20	8678	Intel® Xeon®	Supported
HS20	8832	Intel Xeon	No
HS20	8843	Intel Xeon	Supported
HS20	8678	Intel Xeon	No
HS21 XM	7995	Intel Xeon	Supported
HS21	8853	Intel Xeon	Supported
HS40	8839	Intel Xeon	Supported
LS20	8850	AMD™ Opteron™	Supported
LS21	7971	AMD Opteron	Supported
LS41	7972	AMD Opteron	Supported
JS20	8842	IBM Power	No
JS21	8844	IBM Power	No

We describe the HS21 XM and LS21 in the following sections.

1.2.4 BladeCenter HS21 XM server

The BladeCenter HS21 XM (extended memory) servers are positioned as high-density servers. With one or two dual-core or quad-core Intel processors plus up to 32 GB of DDR2 memory with 8 DIMM sockets., they represent a new approach to the deployment of application servers where performance, capacity, high-availability design, systems management, and easy setup features are combined in an extremely dense package. Reducing an entire server into as little as .5U of rack space does not mean trading away features and capabilities for smaller size.

Integrated dual Gigabit Ethernet controllers are standard, providing high-speed data transfers and offering TCP Offload Engine (TOE) support, load-balancing and failover capabilities. Via optional expansion cards, each blade can also connect to additional Ethernet, Myrinet, Fibre Channel, iSCSI, InfiniBand, and other high-speed communication switches housed in the chassis. Optional 2-port Expansion Cards add additional fabrics to the HS21 XM server as needed. This blade is designed with power management capability to provide the maximum uptime possible for your systems. In extended thermal conditions or power brownouts, rather than shut down completely, or fail, the HS21 XM automatically reduces the processor frequency to maintain acceptable thermal and power levels.

All HS21 XM models also include support one SAS hard disk drive and one USB-based modular flash drive. The optional 30mm Storage and I/O (SIO) Expansion Unit connects to a blade (model-dependent) to provide an additional three 2.5 SAS HDDs with hot-swap support, optional RAID-5 with battery-backed cache, and four additional communication ports.

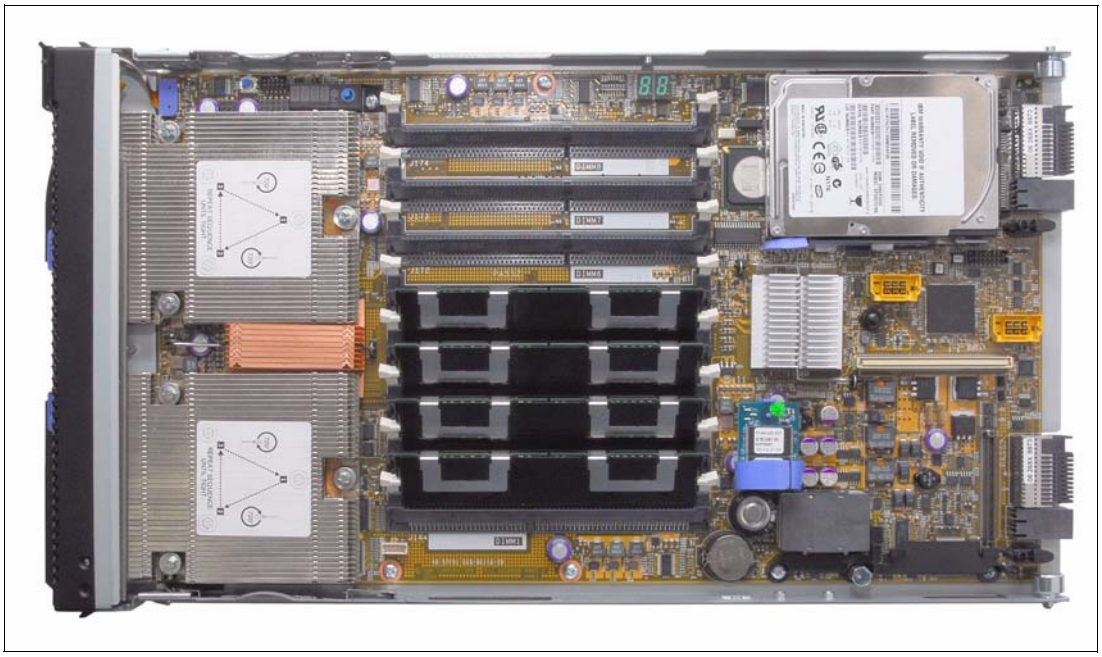


Figure 1-9 The HS21 XM blade server, machine type 7995

Features of the HS21 XM include:

- Blade servers supported in all IBM BladeCenter chassis.
- Two processor sockets supporting either dual-core Intel Xeon 5100 series processors or quad-core Intel Xeon 5300 series processors.

- Up to 32 GB of system memory in 8 DIMM sockets.
- One internal 2.5-inch SAS HDD with support for the Storage and I/O (SIO) expansion unit blade with an additional three 2.5-inch SAS HDD bays and RAID support.
- An optional Modular Flash Drive can be used in place of, or in addition to, the internal SAS HDD, as a boot device.
- Two Gigabit Ethernet ports standard; plus more, using either a 2-port Gigabit Ethernet Expansion Card or a PCI I/O Expansion Unit II.
- Integrated Baseboard Management Controller (BMC) service processor to monitor server availability, perform Predictive Failure Analysis®, etc., and trigger IBM Director alerts.
- Support for IBM PowerExecutive™ 2.0, software designed to take advantage of new system power management features that monitor actual power usage and provide power consumption capping features.
- Concurrent keyboard, video, mouse (KVM) support with addition of the optional IBM BladeCenter Concurrent KVM feature card.
- Three-year, on-site limited warranty.

Table 1-2 Features of the HS21 XM

Feature	
Processor	Intel Xeon 5300 Series quad-core or Intel Xeon 5100 Series dual-core processors
Number of processors (std/max)	1 / 2
Front-side bus	1033 MHz or 1333 MHz
Cache	4 MB or 8 MB L2 cache (shared between both cores)
Memory	8 DIMM slots / 32 GB maximum
Memory Type	PC2-5300 (667MHz) Fully Buffered DDR II ECC
Internal hard disk drives (std/max)	0 / 1 (SAS)
Maximum internal storage	On board: One 2.5” Non Hot Swap SAS HDD On board: One 4 GB modular flash drive Optional: SIO Expansion Unit supports 3 additional 2.5” hot-swap SAS drives
Network	Two ports, Integrated Dual Gigabit Ethernet (Broadcom 5708S), TOE
I/O upgrade	One PCI-X expansion connector and 1x8 PCI-Express expansion connector Optional: PCI Expansion Unit II: Two PCI-X slots Optional: SIO Expansion Unit: One or two PCI-X slots

1.2.5 BladeCenter LS21 server

The BladeCenter LS21 blade servers are high-throughput, two-socket, SMP-capable, AMD Opteron-based blade servers. This two-socket AMD Opteron blade server is well suited to HPC and other applications requiring high performance and high-speed fabrics.

The processors used in these blades are standard- and low-power, full-performance AMD Opteron processors. The standard processors draw a maximum of 95 W. Specially manufactured low-power processors operate at 68 W or less without any performance trade-offs. This savings in power at the processor level combined with the smarter power solution that BladeCenter delivers make these blades very attractive to clients who have limited power and cooling resources.

The BladeCenter LS21 is a single-width blade server that offers these features:

- Up to two high-performance, AMD Opteron dual-core processors.
- System board containing eight DIMM connectors, supporting 512 MB, 1 GB, 2 GB, or 4 GB DIMMs. Up to 32 GB of system memory is supported with 4 GB DIMMs.
- SAS controller, supporting one internal SAS drive (36 or 73 GB) and up to three additional SAS drives with optional SIO blade.
- Two TCP/IP Offload Engine-enabled (TOE) Gigabit Ethernet controllers (Broadcom 5706S) standard, with load balancing and failover features.
- Support for concurrent KVM (cKVM) and concurrent USB / DVD (cMedia) via Advanced Management Module and optional daughter card.
- Support for Storage and I/O Expansion (SIO) Unit.
- Three year on-site limited warranty.

Figure 1-10 IBM BladeCenter LS21

Table 1-3 on page 18 summarizes the features of the LS21.

Table 1-3 Features of the LS21

Feature	Specification
Processor	AMD Opteron Rev F Model 2212, 2212HE, 2216HE and 2218
Number of processors (std/max)	1 / 2
Cache	1 MB L2 per processor core
Memory	8 VLP DIMM slots / DDR2 667 / 32 GB maximum
Internal hard disk drives (standard / maximum)	0 / 1
Maximum internal storage	On board: one 2.5” Non Hot Swap SAS HDD Optional: SIO blade offers support for 3 additional 2.5” hot-swap SAS drives
Network	Two ports, Integrated Dual Gigabit Ethernet (Broadcom 5706S) TOE
I/O upgrade	1 PCI-X expansion connector and 1 PCI-Express expansion connector

1.3 AMD Opteron architecture

BladeCenter servers such as the LS21 and LS41 are based on the AMD Opteron processor. AMD Opteron processors do not use the typical shared front-side bus that is connected to a memory controller used in Intel-based servers. Each processor has its own integrated memory controller and pins on the processor chip to directly connect to a memory bus. So, in AMD Opteron, processor and memory controller logic are integrated into the same piece of silicon, eliminating the need for a separate memory controller part. Hardware vendors simply add a memory bus and memory DIMMs, and they have the core CPU and memory interface.

To keep data coherent between multiple processors, AMD pioneered a new system bus architecture, called *HyperTransport*™ Technology, with the HyperTransport Consortium. Across this interconnect, AMD Opteron processors run a protocol known as coherent HyperTransport, often denoted *cHT*. Three HyperTransport links are available on each AMD Opteron processor of the 2000 and 8000 Series, with the 2000 Series capable of supporting one CPU-CPU connection and the 8000 Series capable of supporting up to three CPU-CPU connections, with indirect connection to up to eight processors. Connections not used for CPU-CPU protocols can be used for non-coherent HyperTransport connections to core logic chipsets.

With a four-processor configuration such as the one implemented in the BladeCenter LS41, the processors are placed at the corners of a square, with each line that makes up the square representing a HyperTransport connection between the processors. See Figure 1-11 on page 19.

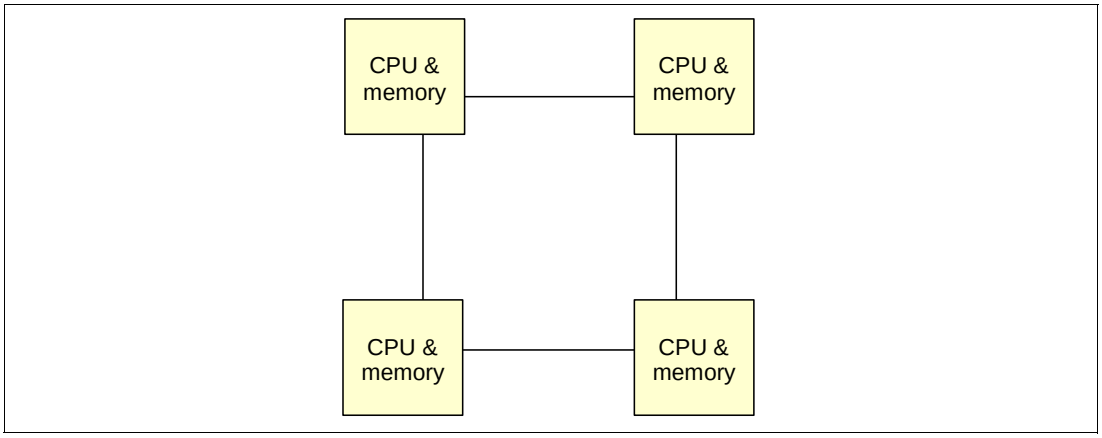


Figure 1-11 A configuration with four AMD processors, each with it's own memory

The third port of the HyperTransport link is not used to interconnect the diagonal processors because it must be used for connection to a PCI I/O bridge, which connects such devices as PCI slots, network, disk storage, mouse, keyboard, video, and so forth.

The remote memory access latency of a processor accessing another processor's memory space makes the AMD Opteron processor configuration a NUMA design. NUMA means that every processor has memory that is *closer* and thus more rapidly accessible, and memory that is *remote* and slower that must be accessed via another AMD Opteron processor.

AMD refers to its processor architecture as *sufficiently uniform memory organization* (SUMO) rather than NUMA. From an architectural standpoint, it still is a NUMA architecture, but the HyperTransport link is fast enough to run software written for SMP systems without very significant performance penalties.

Current operating systems such as the latest versions of Linux and Windows® 2003 SP1 support NUMA and make attempts to minimize remote memory transactions. But in practice, the percentage of remote memory accesses is largely determined by application behavior and by how data is manipulated by users of the application.

Figure 1-12 shows the AMD Opteron architecture with the integrated memory controller and the HyperTransport connectors.

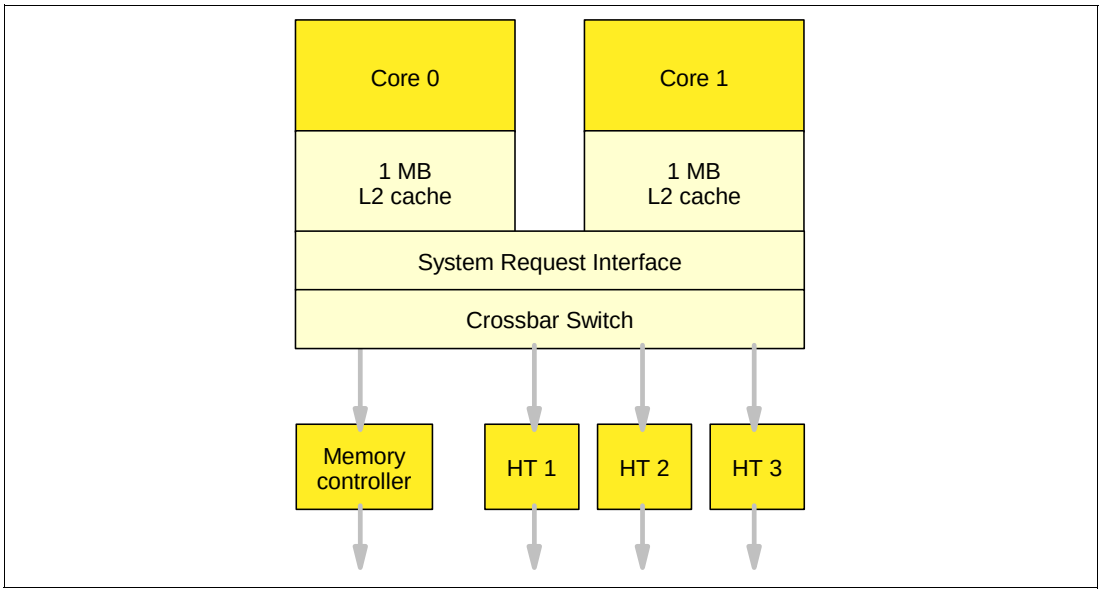


Figure 1-12 CPU architecture of the AMD Opteron CPU with an integrated memory controller

The HyperTransport architecture was initially developed by AMD but is now managed by an open consortium of several big IT companies such as AMD, Apple, Cisco, Broadcom, ATI™, IBM, and many others. HyperTransport is an open standard for a high-speed, point-to-point link system that can be used for connecting a variety of chips.

The HyperTransport technology is used in devices such as network devices, graphics cards, or, as in the case of the AMD Opteron, as a high-speed interconnect for processors. The HyperTransport technology used for interconnecting processors is currently implemented at a speed of 1000 MHz with a bidirectional bandwidth of 4.0 GBps each way that leads to a peak full-duplex capacity of 8.0 GB per second per link. Current AMD Opteron processors incorporate three HyperTransport links, which enables a peak bandwidth of 24 GBps per processor.

You can find more information about the HyperTransport and the HyperTransport Consortium at:

`http://www.hypertransport.org/`

The BladeCenter LS21 and LS41 use the AMD Opteron revision F *Santa Rosa* processor with AMD64 extensions and AMD Virtualization™ (AMD-V™). The following features are incorporated into the Rev F processors:

- All rev F processors are multi-core.
- Support for DDR2 memory is added.
- The current 1 GHz HyperTransport technology is incorporated as the interface to initial Rev F processors.
- PCI Express support is available.
- AMD Pacifica virtualization technology and power management technologies are incorporated.

The major performance increase with Rev F AMD Opteron over the current Rev E AMD Opteron processors is related to the increase to DDR2 memory technology. DDR2 memory technology results in over a 25% increase in memory throughput for random memory

accesses. For a performance analysis of DDR verses DDR2 see the IBM Redbooks publication *Tuning IBM System x Servers for Performance*, SG24-5287.

1.3.1 64-bit extensions: AMD64

The AMD AMD64 architecture extends the well-established x86 instruction set with:

- A set of new 64-bit general purpose registers (GPR)
- 64-bit instruction pointers
- The ability to process data in 64-bit chunks
- Up to 1 TB of address space that physical memory is able to access
- 64-bit integer support and 64-bit flat virtual address space

Even though the names of these extensions suggest that the improvements are simply in memory addressability, the AMD64 are in fact fully functional 64-bit processors.

There are three distinct operation modes available in AMD64:

- 32-bit legacy mode

  The first and, in the near future, probably most widely used mode is the 32-bit legacy mode. In this mode, the AMD64 processors act just like any other IA32-compatible processor. You can install your 32-bit OS on such a system and run 32-bit applications, but you will not be able to make use of the new features such as the flat memory addressing above 4 GB or the additional General Purpose Registers (GPRs). 32-bit applications will run just as fast as they would on any current 32-bit processor.

  Most of the time, IA32 applications will run even faster since there are numerous other improvements that boost performance regardless of the maximum address size. For applications that share large amounts of data there might be performance impacts related to the NUMA-like architecture of multi-processor configurations since remote memory access might slow your application down.

- Compatibility mode

  The second mode supported by the AMD64 is compatibility mode, which is an intermediate mode of the full 64-bit mode described below. In order to run in compatibility mode, you will need to install a 64-bit operating system and 64-bit drivers. If a 64-bit OS and drivers are installed, both AMD Opteron and Intel Xeon processors will be enabled to support a 64-bit operating system with both 32-bit applications or 64-bit applications.

  Compatibility mode gives you the ability to run a 64-bit operating system while still being able to run unmodified 32-bit applications. Each 32-bit application will still be limited to a maximum of 4 GB of physical memory. However, the 4-GB limit is now imposed on a per-process level, not at a system-wide level. This means that every 32-bit process on this system gets its very own 4 GB of physical memory space (assuming sufficient physical memory is installed). This is already a huge improvement compared to IA32, where the operating system kernel and the application had to share 4 GB of physical memory.

  Additionally, compatibility mode does not support the virtual 8086 mode, so real-mode legacy applications are not supported. However, 16-bit protected mode applications are supported.

- Full 64-bit mode (long mode)

  The final mode is the full 64-bit mode. AMD refer to this as *long mode*. This mode is when a 64-bit operating system and 64-bit application are use. In the full 64-bit operating mode, an application can have a virtual address space of up to 40-bits (which equates to 1 TB of addressable memory). The amount of physical memory will be determined by how many DIMM slots the server has and the maximum DIMM capacity supported and available at the time.

Applications that run in full 64-bit mode will get access to the full physical memory range (depending on the operating system) and will also get access to the new GPRs as well as to the expanded GPRs. However it is important to understand that this mode of operation requires not only a 64-bit operating system (and of course 64-bit drivers), but also requires a 64-bit application that has been recompiled to take full advantage of the various enhancements of the 64-bit addressing architecture.

For more information about the AMD64 architecture see:

http://www.x86-64.org/

1.3.2 The benefit of AMD64 64-bit computing

In the same way that 16-bit processors and 16-bit applications are no longer used in this space, it is likely that at some point in the future, 64-bit processors and applications will fully replace their 32-bit counterparts.

Processors using the AMD64 architecture are making this transition very smooth by offering 32-bit and 64-bit modes. This means that the hardware support for 64-bit will be in place before you upgrade or replace your software applications with 64-bit versions. IBM System x now has several models available with the AMD Opteron processors.

The question you should be asking is whether the benefit of 64-bit processing is worth the effort of upgrading or replacing your 32-bit software applications. The answer is that it depends on the application. Here are examples of applications that will benefit from 64-bit computing:

- Encryption applications

  Most encryption algorithms are based on very large integers and would benefit greatly from the use of 64-bit GPRs and ALUs. While modern high-level languages allow you to specify integers above the 2^{32} limit, in a 32-bit system, this is achieved by using two 32-bit operands, thereby causing a significant overhead while moving those operands through the CPU pipelines. A 64-bit processor will allow you to perform 64-bit integer operation with one instruction.

- Scientific applications

  Scientific applications are another example of workloads that need 64-bit data operations. Floating-point operations do not benefit from the larger integer size since floating-point registers are already 80 or 128 bits wide even in 32-bit processors.

- Software applications requiring more than 4 GB of memory

  The biggest advantage of 64-bit computing for commercial applications is the flat, potentially massive, address space.

  32-bit enterprise applications such as databases are currently implementing Page Addressing Extensions (PAEs) and Addressing Windows Extensions (AWEs) addressing schemes to access memory above the 4-GB limit imposed by 32-bit address limited processors. With EM64T and AMD64, these 32-bit addressing extension schemes support access to memory up to 128 GB in size.

  One constraint with PAE and AWE, however, is that memory above 4 GB can only be used to store data. It cannot be used to store or execute code, so these addressing schemes only make sense for applications such as databases, where large data caches are needed.

  In contrast, a 64-bit virtual address space provides for direct access to up to 2 Exabytes (EB), and even though we call these processors 64-bit, none of the current 64-bit

processors actually supports a full 64 bits of physical memory addressing, simply because this is such an enormous amount of memory.

In addition, 32-bit applications might also get a performance boost from a 64-bit AMD64 system running a 64-bit operating system. When the processor runs in compatibility mode, every process has its own 4-GB memory space, not the 2-GB or 3-GB memory space each gets on a 32-bit platform. This is already a huge improvement compared to IA32 where the OS and the application had to share those 4 GB of memory.

When the application is designed to take advantage of more memory, the availability of the additional 1 or 2 GB of physical memory can create a significant performance improvement. Not all applications take advantage of the global memory available. APIs in code need to be used to recognize the availability of more than 2 GB of memory.

Furthermore, some applications will not benefit at all from 64-bit computing and might even experience degraded performance. If an application does not require greater memory capacity or does not perform high-precision integer or floating-point operations, then 64-bit will not provide any improvement.

In fact, because 64-bit computing generally requires instructions and some data to be stored as 64-bit objects, these objects consume more physical memory than the same object in a 32-bit operating environment. The memory capacity inflation of 64-bit can only be offset by an application taking advantage of the capabilities of 64-bit (greater addressing or increased calculation performance for high-precision operations), but when an application does not make use of the 64-bit operating environment features, it often experiences the overhead without the benefit.

In this case, the overhead is increased memory consumption, leaving less physical memory for operating system buffers and caches. The resulting reduction in effective memory can decrease performance.

1.3.3 64-bit memory addressing

The width of a memory address dictates how much memory the processor can address. A 32-bit processor can address up to 2^{32} bytes or 4 GB. A 64-bit processor can theoretically address up to 2^{64} bytes or 16 Exabytes (or 16,777,216 TB), although the AMD Opteron (64-bit) can directly address up to 256 TB (48-bit) and when using PAE up to 128 GB in compatibility mode.

The 64-bit extensions in the processor architectures AMD64 provide a better performance for both 32-bit and 64-bit applications on the same system. These architectures are based on 64-bit extensions to the industry-standard x86 instruction set and provide support for existing 32-bit applications.

1.4 Application migration

Most top tier Independent Software Vendors that offer applications for Solaris on the SPARC platform also offer compatible software for Solaris on the x86 platform. IBM DB2® database for example is offered for Solaris in both the SPARC platform to the Intel/AMD processor platform. In this case you would only need to purchase a license for x86 platform version of DB2 to move your database from the SPARC platform to the x86 platform. This makes such a transition relatively uneventful.

What may be of concern are custom written applications. Some custom written applications may only require being recompiled in order to transition from the SPARC platform to the x86 platform. Other custom applications may require modification or a total rewrite.

Sun provides an Application Binary Interface (ABI) such that when applications are written to that interface, they are assured of compatibility. Currently IBM does not offer any direct assistance to aid in the migration of custom Solaris applications from the SPARC platform to the x86 platform. Sun does offers services to aid customers in migrating custom applications from the SPARC platform to the x86 platform. For further information about application migration services offered by Sun contact your Sun sales representative.

1.4.1 Application Binary Interface

Sun supports a public Application Binary Interface (ABI) that is designed to help properly written applications run on all Sun systems without modification. This programming interface helps ensure compatibility throughout the SPARC product line as well as x86 platforms. It enables Sun to make improvements in the Solaris operating system so long as the improvements conform to the ABI. Because the interface is consistent regardless of the underlying hardware platform, the ABI also enables developers to easily move an application from one architecture to another.

For example, moving an application from the SPARC platform to the x86 platform requires a simple recompile to generate an executable that will exhibit the same behavior on both platforms.

Sun provides a set of compatibility-testing tools in the Solaris Compliance Assurance Toolkit (SolCAT). This toolkit includes the new Solaris Application Scanner, which quickly assesses whether software that runs on previous versions of the Solaris operating system will run on the Solaris 10 OS. The Solaris Appcert tool does a complete job of examining an application's conformance to the Solaris ABI and reports potential release-to-release stability problems.

1.4.2 Solaris Compatibility Assurance Toolkit

Sun currently provides Solaris Compatibility Assurance Toolkit (SolCAT), an extensible tool framework to help ISVs, developers and system administrators test compliance with the Solaris ABI and ease adoption to Solaris 10.

SolCAT currently consists of two complementary tools that examine usage of the Solaris library ABI by the installed applications:

- Solaris Application Scanner that checks if the application uses nonexistent interfaces
- Solaris Appcert that checks if the application uses undocumented and unsupported interfaces

The two tools examine ELF binaries' (Executable and Linking Format) usage of the Solaris library ABI for C, C++, Fortran, and Cobol compiled applications, and so on.

Note: Other aspects of the ABI including interpreted programs (for example, Java™ applications and shell scripts), use of utility commands, configuration files, and device drivers, and the like are out of the scope of these tools.

These tools may be scanned over application binaries to detect issues that might affect the application's compatibility when run on a later Solaris release. Both tools have been updated to test for issues up to and including Solaris 10.

SolCAT can be downloaded from:

http://www.sun.com/software/solaris/programs/abi/solcat.xml

To learn more about Solaris Compatibility Tools, visit

http://www.sun.com/software/solaris/programs/abi/get.xml

Solaris Application Scanner

The Solaris Application Scanner was created in response to feedback from Solaris Appcert users that wanted a tool with the following features:

- Results that were easier to understand than Solaris Appcert's
- A “Go/No-Go” decision for an application
- Faster running time than Solaris Appcert when run over many binaries

The resulting tool focuses on a relatively straightforward check: Does the application directly call any Solaris interfaces or libraries that have gone away in a Solaris release? If so, the application has a very high chance of failing when run on that Solaris release.

Given an application binary with a Solaris Application Scanner “No-Go” status, the only ways the application would not fail at runtime are:

- The application (in principle or in practice) never calls the removed interface.
- The application (using its own support shared objects) provides its own implementation of the removed interface.
- An accidental coincidence exists between the name of the Solaris interface that was removed and one the application provides to itself (using its own support shared objects).

These cases are all relatively rare (and are listed above in order of increasing rarity), and therefore a No-Go result from Solaris Application Scanner should be taken seriously.

Since Sun believes binary compatibility for the Solaris platform is very important, it is extremely rare for public interfaces or libraries to be removed. A lengthy End-Of-Life process is performed before doing so. For private interfaces and libraries (that is, those intended only for Solaris-internal usage: they are not part of the API/ABI), no such constraint exists (because applications are not supposed to be using them). For more information about Sun's EOL process see: Solaris OS Life Cycle.

The Solaris Application Scanner database contains information on both public and private interfaces that have been removed from the Solaris operating system. The private interfaces are included in case an application has been using the private interfaces and has “just happened to work” until the private interface is removed. Because public interfaces are only rarely removed from the Solaris operating system, the vast majority of the Solaris Application Scanner database consists of removed private interfaces.

The Solaris Application Scanner tracks removal of interfaces and libraries from Solaris 2.4 up to and including the Solaris 10 OS.

Appcert Tool for the Solaris OS

The Solaris Appcert tool has been updated to handle some additional potential binary compatibility problems specific to the Solaris 10 OS. In addition, its symbol database has been updated to provide a “snapshot” of the Solaris 10 OS to go along with its existing snapshots for releases from Solaris 2.3 to Solaris 9.

The Solaris Appcert tool takes a somewhat different approach from Solaris Application Scanner: Solaris Appcert looks for and reports potential binary incompatibilities. That is to say, its primary function is to give “heads-up” warnings about vulnerabilities in the application that could lead to failure when run on future Solaris releases. The main check Appcert performs is for the application's direct usage (that is, calling) of private interfaces in the Solaris operating system.

The developer of an application will be interested in removing all Solaris Appcert-reported vulnerabilities from the application (by modifying the application source code and rebuilding and redeploying). Similarly, a system administrator or quality assurance test engineer may decide to perform extra testing on newer Solaris releases of applications that yielded Solaris Appcert warnings.

Some overlap exists in the checking that Solaris Appcert and Solaris Application Scanner perform, in particular: Solaris Appcert does check for usage of a large number of interfaces or libraries that have been removed. These checks are most concerned with public interfaces or libraries that have been removed (although quite a few private interfaces are included as well), while the Solaris Application Scanner has a large database of essentially all interfaces (public or private) that have been removed.

Solaris Appcert has some miscellaneous checks for potential binary incompatibility problems not simply involving private interface usage. The checks added in this Solaris 10 release are:

- Manipulates lightweight processes directly instead of using thread (some _lwp_* interfaces may be removed)
- Non-POSIX libthread fork call (A rare chance exists that an application may need to switch to forkall)
- Potential source code incompatibility from makecontext (source code issue only when rebuilt on the Solaris 10 OS or later)
- SunOS™ 4.x a.out (non-ELF) binaries (warning only, no BCP EOF is planned)
- Usage of deprecated and EOL interface 64-bit ptrace
- Usage of deprecated and EOL interface auditsvc
- Usage of deprecated and EOL interfaces sysmem/asysmem (accidentally introduced into and removed from SVID)
- Usage of deprecated libXinput.so.0 (should migrate to X11 standard libXi.so.1)
- Usage of EOL libldap.so.3 (should migrate to libldap.so.4 or libldap.so.5 standards)
- Usage of EOL libxfn.so
- Usage of EOL libkcs.so.1
- Usage of /usr/lib/libproc.so.1 interfaces (these interfaces remain private)
- Usage of removed private library libdevfsevent.so.1
- Usage of removed private library libdhcp.so.2
- For more information, see End-of-Software Support Statements

For more information about SolCAT, see:.

http://www.sun.com/software/solaris/programs/abi/solcat.xml

1.5 IBM software supported on Solaris

The products below are a subset of IBM Software products that are supported on Solaris 10 running on x86/x64. You can also check the following Web site for supported IBM products:

http://www.sun.com/bigadmin/apps/

Rational® products:

- Rational Software Architect 6.0
- Rational Clearcase Client 7.0.1 (2003.06.00)
- Rational Purify® Plus 7.0

Tivoli® products:

- Tivoli Management Framework Endpoint 4.1.1 FP7 (4.1.1-LCF-0032)
- GSKIT 7.0.3.19
- LDAP Static Client (Directory Server 6.0)
- Tivoli Configuration Manager Endpoint 4.2.3 FP3
- Tivoli Monitoring 6.1
- Tivoli Enterprise™ Console Endpoint 3.9 FP4
- Tivoli Directory Server 6.1
- LDAP Dynamic Client (Directory Server 6.0)
- Tivoli Access Manager for eBusiness 6.0
- Tivoli License Compliance Manager 2.3
- Tivoli Monitoring for Applications (SAP® & Siebel®) 6.1
- Tivoli Monitoring for Databases (DB2, Oracle® & Sybase) 6.1
- Omegamon XE for Messaging 6.0.0.1
- Tivoli Storage Manager 5.4.1, 5.3.5
- Tivoli Storage Manager Client Server version 5.3.3 as a PTF
- Tivoli Storage Manager for Databases (Oracle & DB2) 5.4

WebSphere® products:

- WebSphere Application Server 6.1 (6.0.2)
- WebSphere Application Server, Network Deployment 6.1 (6.0.2)
- WebSphere MQ 6.0
- WebSphere Portal Enable for Multiplatforms 6.0.0.1 (6.0)
- WebSphere Process Server 6.0.2 (6.0.0.2)
- WebSphere MQ Event and Message Broker 6.0.2 (5.3)

Database products:

- DB2 UDB Enterprise Server Edition 8.1 FP15, 9.1 FP3 (8.2.2)
- DB2 UDB WorkGroup Server Edition 8.1 FP15, 9.1 FP3 (8.2.2)
- DB2 UDB Express Edition 8.1 FP15, 9.1 FP3 (8.2.2)
- DB2 Content Manager 8.3.3 (8.3.2)
- DB2 Records Manager 4.1.3.6 (4.1.3)
- DB2 CommonStore for SAP 8.2
- Informix® IDS 11.10

2

Installation

This chapter describes the installation of Solaris onto a blade. In addition, we discuss how to create a Solaris Installation Server and necessary updates to perform an installation onto an LS21 based server.

Topics in this chapter are:

2.1 Installation

The installation of Solaris 10 11/06 on IBM BladeCenter servers is basically the standard Solaris installation procedure with a few exceptions:

- For some blade servers, you will need to disable **Reboot on NMI**. See Table 2-1 for blades that require this setting.
- For blade servers, such as the LS21, LS41, HS21, and HS21 XM servers (see Table 2-1), you will need to apply certain Solaris patches to the Solaris 10 11/06 code base, either by integrating those patches into the installation media or with the use of the Solaris Installation Server. We will cover the later in this chapter as this method is better suited for handling the deployment of multiple blades.

We cover these in the following sections.

Table 2-1 Blade Servers and installation information

System	Support (first supported release)	Installation instructions	Disable reboot on NMI	11/06 and patches required to install
HS20 type 7981	Supported (1/06+)	MIGR-64047	Required	No
HS20 type 8832	No	Not Applicable	Not Applicable	Not Applicable
HS20 type 8843	Supported (1/06+)	MIGR-62921	Required	No
HS20 type 8678	No	Not Applicable	Not Applicable	Not Applicable
HS21 XM type 7995	Supported (11/06+)[a]	Planned	Required	Required
HS21 type 8853	Supported (11/06+)[a]	Planned	Required	Required
HS40 type 8839	Supported (1/06+)	MIGR-5070449	Required	No
LS20 type 8850	Supported (1/06+)	MIGR-63018	Not Required	No
LS21 type 7971	Supported (11/06)[a]	Planned	Not Required	Required
LS41 type 7972	Supported (11/06+)[a]	Planned	Not Required	Required
JS20 type 8842	No	Not Applicable	Not Applicable	Not Applicable
JS21 type 8844	No	Not Applicable	Not Applicable	Not Applicable

a. Requires 11/06 with patches or later

2.1.1 Disabling Reboot on NMI

If the blade server that you are installing to requires you to disable Reboot on NMI (see Table 2-1), perform the following steps before beginning the installation.

1. Turn on the blade server.
2. When the prompt `Press F1 for Configuration/Setup` is displayed, press F1.
3. Select **Advanced Setup** then **Baseboard Management Controller (BMC) Settings**, and select **Reboot System on NMI**.
4. Press the Right Arrow or Left Arrow key until **Disabled** is selected.
5. Press Esc twice.
6. Select **Save Settings**.

These steps are required due to a non-maskable interrupt (NMI) that is seen during installation.

Tip: Once the operating system installation is complete, you can re-enable the Reboot on NMI setting.

2.1.2 Patches to the 11/06 version

Blade servers such as the LS21, LS41, HS21, and HS21 XM will need additional patches to Solaris 10 11/06. For the list of servers that need patches, see Table 2-1 on page 30.

To install the patches on these servers, an installation using either an Installation Server (see 2.3, “Use of a Solaris Installation Server” on page 39) or to create an updated DVD image. Directions for how to perform this, is located in chapters of the Sun document “Solaris 10 Installation Guide for IBM BladeCenter Servers” that can be found at:

http://docs.sun.com/app/docs/doc/819-5455

In the Chapter 3 of this document, there are instructions for performing an updated installation through an Installation Server under the heading “How to Create an x86 Solaris 10 11/06 OS Installation Server”. You should then perform the steps outlined in the next section “How to Perform an Installation of the Solaris 10 11/06 OS on the IBM BladeCenter LS21 Server” to complete the installation onto the LS21. There are also instructions for performing an installation using an updated DVD media in the subsection “How to Create a DVD ISO Image”.

The required patches can be downloaded from Sun at:

http://sunsolve.sun.com

You will need the following patches:

- 123840-04.zip
- 118855-36.zip
- 125034-01.zip

If you will be using a BladeCenter Storage and I/O (SIO) Expansion Unit (sometimes referred to as a BSE3) attached to your system, you will also need to download:

- 125325-01

Tip: The above document “Solaris 10 Installation Guide for IBM BladeCenter Servers” gives the current list of required patches in the Troubleshooting section.

For all of the above patches, the version number of the patch may have been updated, so the number after the dash may be higher. Ensure you use the most recent patch.

You will also need to download the updated Broadcom driver. At the time of writing, we were using the 3.0.0 driver, and will reference this version during the course of this paper. The driver can be obtained from:

http://www.broadcom.com/support/ethernet_nic/netxtremeii.php

For the BladeCenter LS41, if the bnx driver is Version 3.0.0, or a version prior to Version 3.0.0, you must also download the latest version of the Broadcom bcme driver from the Broadcom Web site.

Directions for downloading and installing the Broadcom bcme driver can be found on the Broadcom Web site at:

http://www.broadcom.com/support/ethernet_nic/netxtreme_server.php

Tip: These patches were incorporated in Solaris 10 8/07.

2.1.3 Installing Solaris 10

Installation instructions using Solaris 10 1/06 are documented for the following blade systems.

- HS20 type 7981:

  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-64047
- HS20 type 8843:

  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-62921
- LS20 type 8850:

  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-63018
- LS40 type 8839:

  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-5070449

Other servers as specified in Table 2-1 on page 30 are planned, but at the time of writing they were not published.

These systems were test using Solaris 10 1/06 as a minimum level. The initial version of Solaris 10 had various incompatibilities with the IBM BladeCenter server that were resolved in the 1/06 update. As a result, Solaris 10 1/06 is the first supported version.

2.2 Boot from SAN

This document illustrates the detailed configuration procedure necessary to implement boot from SAN (storage area network) function using the IBM BladeCenter modules with the Sun Solaris 10 operating system and the IBM DS4000™ storage system.

Important: Boot from SAN running Sun Solaris on a blade server is not an IBM supported configuration.

2.2.1 Boot from SAN overview

Boot from SAN allows for deployment of diskless servers in an environment where the boot disk is located on the RAID (redundant array of independent disks) capable storage server connected to the SAN. The server (Initiator) communicates with the storage device (target) through the SAN using the Fibre channel host bus adapter (HBA). The software configuration parameters in FC BIOS of the HBA allow the server to identify and map the boot disk on the SAN. The boot from SAN function allows for consolidation of the server and storage hardware and facilitates a central point of management.

The system downtime is greatly minimized in case a critical component such as a processor or host bus adapter fails and needs to be replaced. The system administrator only needs to swap the hardware and reconfigure the FC HBA BIOS, fabric zoning plus change the host

port definitions on the storage server. The system image still exists on the logical drive and the server is operational once the hardware swap and configuration change is completed.

Because the FAStT storage subsystem is configured with dual redundant controllers for fault tolerance, high availability and load balance, it is critical for the initial install of the operating system that only a single and unique path from the Blade to the logical drive on the storage subsystem is configured. As such, the secondary path via the second controller must be disconnected or isolated by excluding the second port on the server blade's FC daughter card and the FAStT Controller B in the initial zone configuration. The second path is included in the active zone configuration after the operating system is successfully installed and verified that the blade server successfully boots from the logical drive on the SAN.

2.2.2 Topology overview

The number of potential equipment combinations to use for use in boot from SAN scenarios is quite large, so we used a relatively simple configuration for our proof of concept work in the lab. Our environment for performing the blade server booting Solaris from a SAN-attached disk consisted of the following:

- HS20 blade for installation
- HS20 blade as a network installation server
- BladeCenter with dual QLogic™ Fibre Channel switches
- DS4300 Storage Subsystem with 15 GB logical volume allocated to the server blade

A simple network diagram of our environment is shown in Figure 2-1.

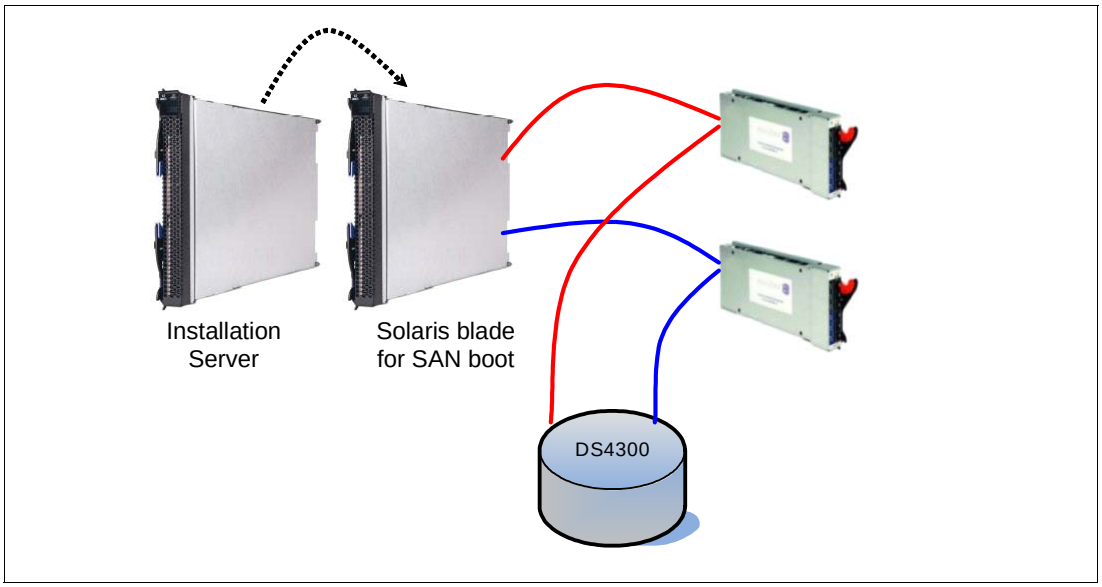


Figure 2-1 Network diagram of Solaris boot from SAN environment

2.2.3 Configuration overview

The process to implement boot from SAN has a number of steps that must be undertaken in a specific sequence to ensure consistent results. The basic steps are shown below with some steps being discussed in detail and other steps briefly covered.

The implementation steps in order are:

1. Solaris host configuration for the primary path
2. Switched fabric configuration for the primary path
3. DS4000 storage configuration for the primary path
4. Solaris Host adapter BIOS configuration for primary path
5. Solaris operating system installation
6. Verification of primary path
7. Switched fabric configuration for the secondary path
8. DS4000 storage for the secondary path
9. Solaris host adapter BIOS configuration for the secondary path
10. Failover verification

The steps which involve switched fabric configuration, DS4000 storage configuration and Solaris operating system installation will not be covered in great detail. The following sections provide details for each implementation step.

Step 1 - Solaris host configuration for the primary path

The following steps illustrate the procedure for booting a blade server from an IBM DS4000 storage system via the SAN, using Solaris 10. The first action involves disabling any internal disk drives before configuring the primary path across the SAN. The following steps are necessary for this task:

1. Power on the blade server and interrupt the boot sequence to enter the server's BIOS by pressing F1 key when prompted.
2. Select **Devices and IO ports** from the main menu.
3. Select **IDE configuration** from this menu.
4. Disable the Primary IDE and Secondary IDE connection.
5. Press Esc to exit out of this menu and save the changes.

The next step is to configure the HBA port BIOS to be used as the primary path. The steps to configure the BIOS are:

1. Power on or restart the Solaris host blade
2. Press <Ctrl+Q> or <Alt+Q> to enter the QLogic BIOS utility when prompted
3. Select the first Fibre Channel adapter port
4. Select **Configuration Settings ... Host Adapter Settings** and then change the Host Adapter BIOS setting to **Enabled**. The default value is Disabled. This is shown in Figure 2-2 on page 35.

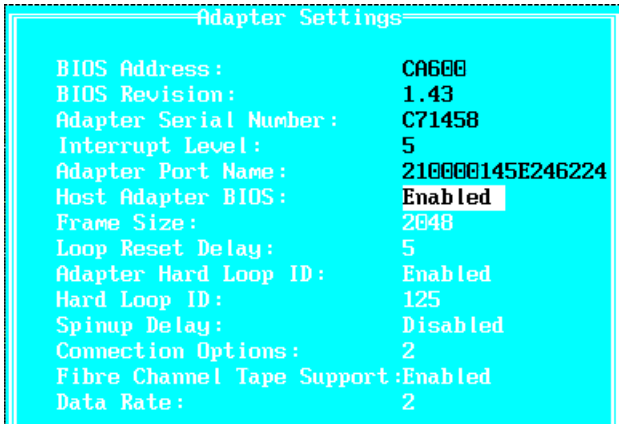


Figure 2-2 Host Adapter BIOS enabled

5. Record the World-Wide Port Name (WWPN) of this host bus adapter. It will be needed while configuring LUN masking on the storage device as well as being needed for modifying the fabric's zoning configuration.

Step 2 - Switched fabric configuration for the primary path

The switched fabric configuration actions are based on a number of conditions which are assumed to be already present. These conditions are:

- The necessary fiber cable connections for DS4000 controllers, storage expansion units, BladeCenter switch module and any external devices that will be connected to the SAN are properly connected.
- All the switches in the fabric are configured with unique Domain ID and IP address
- All the switches in the fabric are running latest and compatible firmware version
- The switches have merged into the fabric successfully
- The fabric NameServer database has all of the host HBA ports and the relevant DS4000 storage ports are Online as F ports

Fabric zoning for the blade server connections to the DS4000 was modified with the addition of two new zones. The first zone has zone members which are the server's HBA port associated with the primary path and a DS4000 host-side port which connects to the preferred controller for the boot partition, LUN ID 0, as shown in Figure 2-3.

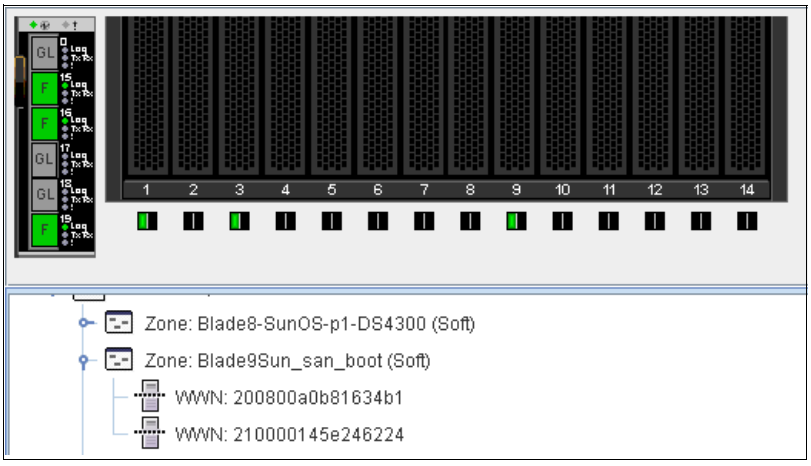


Figure 2-3 Zoning implementation

To minimize potential fabric disruptions due to zoning changes, we added both of the new zones to the active zoning configuration in the fabric at the same time. However, to ensure that only the primary path was active, we disconnected the optical fiber cable between the external switch and the DS4000 host-side port to the controller that is not the preferred owner of the boot partition, LUN ID 0. This secondary path will be enabled in a later step after the primary path has been fully configured and the Solaris operating system has been installed.

Step 3 - DS4000 storage configuration for the primary path

Prior to implementing this step, you will need to know the world-wide port name (WWPN) for the Solaris server's HBA ports. In addition to this information, you must decide which HBA port will be associated with the primary path while the other server HBA port is associated with the secondary path.

To configure the DS4000 storage system, we performed the following actions:

1. Create a logical drive to be used as the operating system (C drive) disk for the blade server.
2. Complete the Host Group, Host and Host Port definitions. For clarity and simplicity, we defined the host ports in the order of primary path first and secondary path next.
3. Define the Storage Partition to map the logical drive with LUN ID 0 to the host port in the Host group which is associated to the primary path.
4. Delete the Access LUN (ID =31), if it was automatically assigned to the host group while defining the storage partition for the logical drive.

Be sure that the server HBA port associated with the primary path has a communication path to the DS4000 controller which is the preferred controller for the boot LUN, LUN ID 0.

Note: Record the WWPN of the primary path into the storage system. This information will be needed in the next step.

Step 4 - Solaris host adapter BIOS configuration for primary path

At this point, the storage configuration process for the primary path is complete. Ensure that the secondary path from the Solaris host to the storage device is not connected. This disconnect can be accomplished by one of the following methods:

- Disable fabric connection to the storage port along the secondary path.
- Disable the fabric connection to the Solaris host's secondary path.
- Disable the zone definition of the secondary path.

The next step is to configure the host HBA BIOS to point to the boot LUN, LUN ID 0, on the storage device. The actions to complete this step are:

1. Reboot the Solaris host being configured for remote boot and enter the QLogic BIOS utility by pressing Ctrl+Q function keys.
2. Ensure that the BIOS for the adapter is still set to Enabled.
3. Exit the menu by pressing Esc key.

Note: To verify that the Host Adapter was able to properly communicate with the DS4000, we used the Scan Fibre Channel Loop option which resulted in the DS4000 logical volume being automatically discovered.

4. Select **Selectable Boot Settings** from the main menu.

5. Set the field BOOT Device to the controller's World Wide Port Name. If the Host Adapter has already scanned for available devices, you will be presented with a listing of device WWPNs to select from.
6. To enable and map the boot device, set the Boot LUN to reflect LUN 0. The result will appear similar to our example shown in Figure 2-4.
7. Save all changes and exit from the BIOS menu by pressing Esc.

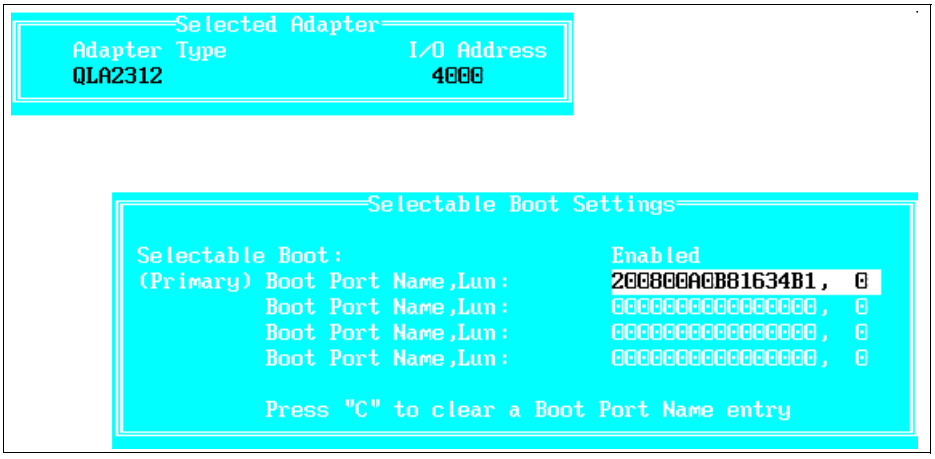


Figure 2-4 Boot setting configuration

Step 5 - Solaris operating system installation

Install Solaris 10 as described in 2.1, “Installation” on page 30. We used a network installation server on a different blade server during the installation, as shown in Figure 2-5.

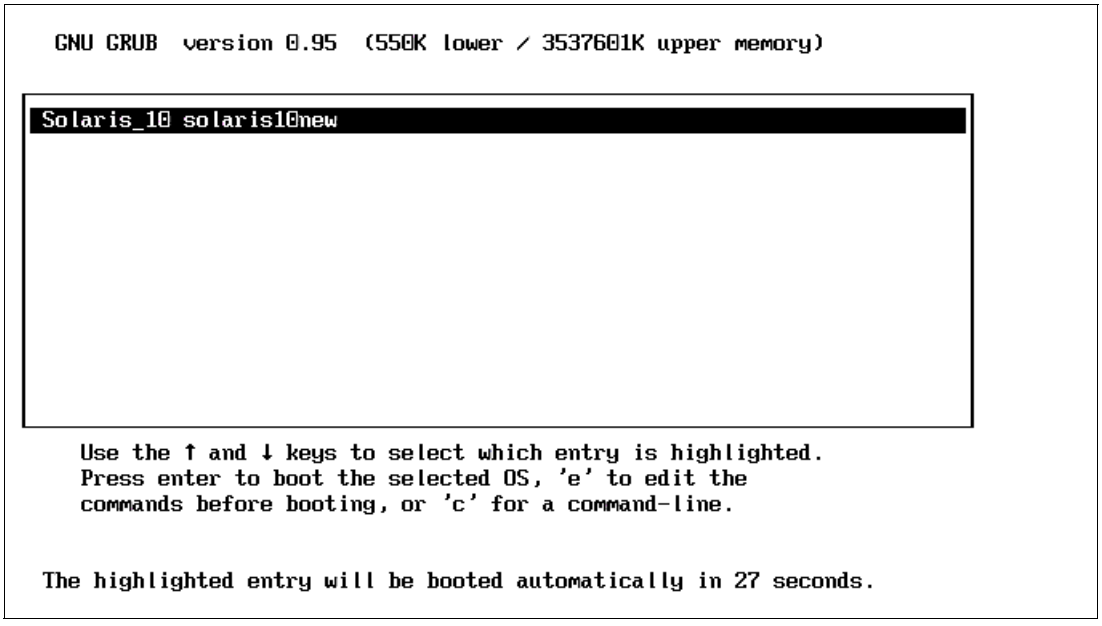


Figure 2-5 Start of Solaris installation from install server

Step 6 - Verification of primary path

Reboot the Solaris host blade to verify that it successfully boots from the logical drive on the primary path. When we begin the first Solaris boot from SAN, there was a minor delay due to this time being the initial boot for loading and configuring the appropriate drivers, as shown in Figure 2-6 on page 38.

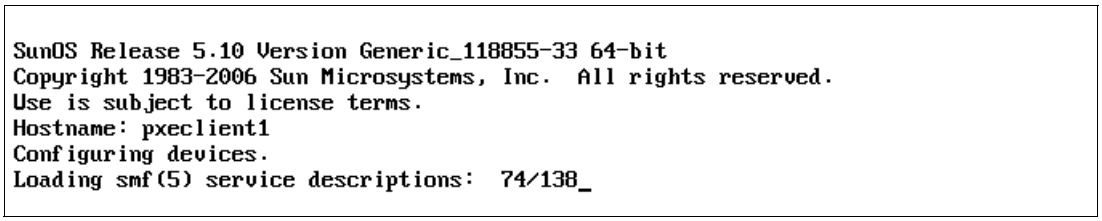


Figure 2-6 Initial loading of Solaris from SAN-attached disk

Step 7 - Switched fabric configuration for secondary path

Once the primary path to the boot partition has been successfully tested, the secondary path needs to be enabled as a failover path for redundancy. If the zone definition has already been added to the active zoning configuration, then the only action needed is the activation of this path between the server's secondary HBA port and the DS4000 controller which is not the preferred owner for the boot LUN, LUN ID 0.

In our test scenario, we enabled the end-to-end connectivity by simply enabling the external switch port to the DS4000 host-side port which goes to the non-preferred DS4000 controller.

Step 8 - DS4000 storage configuration for the secondary path

If the second Solaris host HBA port has not been defined and mapped to the boot LUN, LUN ID 0, then you need to add the second port on the HBA under the same host group used for the primary path configuration. The host type should be the same as for the first host port. This step provides redundant paths to the storage device which can be used by the multipathing utility after the Solaris operating system is running on the host.

Step 9 - Solaris host adapter BIOS configuration for secondary path

Now that the primary path is operational and the secondary path has been configured within the SAN and storage device, the second Solaris host HBA port is ready for configuration. The following actions are necessary to complete this step:

1. Power on or reset the blade server and enter the QLogic BIOS utility by pressing Ctrl+Q when prompted.
2. Select the second HBA port on the Solaris host
3. From the Basic Settings panel, verify and configure that the BIOS on this adapter is set to Enabled.
4. From the Configurable Boot Settings panel, set the **BOOT Device** field to the storage devices host side port which is associated with the secondary path.
5. Set the Boot LUN to reflect LUN 0.
6. Save all changes and exit from the BIOS menu by pressing Esc.

At this point the setup process for the secondary path is complete.

Step 10 - Failover verification

In order to prove that the Solaris host blade can successfully utilize the secondary path in case of a primary path fails or becomes unavailable, simply disable the primary path. This action can be accomplished by either disabling the switch port connection to the Solaris host primary path HBA port or the fabric's connection on the primary path to the storage device. Depending on the multipathing application in use, verify that the secondary path is now active.

2.3 Use of a Solaris Installation Server

An Installation Server allows you to perform a remote installation to one or more blade servers across the network. You do not have to be physically near the blade server, nor do you necessarily need installation media.

The time you will save in doing multiple installations is well worth the time it takes to set up the Installation Server system. In addition, if you plan to install Solaris 10 on servers such as the BladeCenter LS21 or HS21, the use of an Installation Server will make it easier to incorporate any necessary patches (see 2.1.2, “Patches to the 11/06 version” on page 31). In fact, our procedure documented here uses the Installation Server to create patched installation media for use with such servers.

Depending on how you want to perform the installation, there are different steps that you will need to perform. The steps build on each other depending on the complexity of the installation environment you want to have within your business. The choices include:

- Install from code over the network but start and run the process locally

  For the most basic configuration, where you are installing to a blade by booting from optical media in the BladeCenter chassis, but using the other portions of your media remotely, then you will need to set up two components:

  – An Install Server
  – A Boot Server

  This will speed up your installations by having the file copy portion of the installation occurring over the network, which greatly speeds up the installation. The boot media can either be in the optical drive in the BladeCenter chassis or in the optical drive of a remote management system where you are connected to the chassis’s management module.

  Note: For servers such as the BladeCenter LS21 or HS21, you will need to patch the Install Server (as described in 2.3.3, “Patching the Install Server for LS21/HS21 installations” on page 42) and then create a new bootable DVD to boot the servers from. These DVDs are the bare minimum of steps that will need to be performed to have an installation complete.

  We describe this procedure starting with 2.3.1, “Setting up an install server” on page 40.

- Boot and install from the network but manually complete the installation process

  If you do not want to boot from optical media locally at the BladeCenter chassis, you can also configure a DHCP server and configure PXE remote booting. This not only allows running systems to use DHCP, but will allow the systems that you want to install to PXE boot to the Install Server.

  This removes the requirement of being at the physical system to perform the installation, but you will still need to provide input during the installation to configure the system (perhaps remotely using the Remote Control feature of the management module interface)

  We describe this procedure in 2.3.5, “Creating a DHCP and PXE server” on page 46.

- Fully unattended and remote installation

  With the addition of a Jumpstart Server, you can do a complete unattended installation. You pre-configure the installation options and the Jumpstart Server answers all installation questions based on your parameters. Depending on the number of blades you want to deploy, a Jumpstart server could greatly improve your efficiency.

  We will not be covering the use of the Jumpstart Server in this paper.

For our installation server, we performed an installation of Solaris 10 11/06 into a separate IBM System x server. We then created an installation server using the documentation found in *Solaris 10 11/06 Installation Guide: Network-Based Installations*, available from:

http://docs.sun.com/app/docs/doc/819-6395

Note, this document is updated each time an additional update to the Solaris 10 OS is released and is located under the Release and Installation Collection for that updated version.

For servers such as the LS21 and HS21 (see Table 2-1 on page 30), additional patches are necessary, and those instructions can be found at *Solaris 10 1/06 Installation Guide for IBM BladeCenter Servers* which can be found at the following URL, under the Troubleshooting Installation Issues on IBM BladeCenter Servers (Tasks) subsection:

http://docs.sun.com/app/docs/doc/819-5455

An installation server, at a minimum, will require an Install Server and a Boot Server. We also cover the optional DHCP server. We will not be covering the creation of a Name Server nor how to create a Jumpstart Server in these instructions:

- Install Server (required)
- Boot Server (required)
- DHCP Server (optional)
- Name Server (optional)

2.3.1 Setting up an install server

For the install server, you must use the DVD based media. Our testing shows that you will need at least 3 GB of disk space. Make sure that you have enough space for your images before they are uploaded. In order to create a burnable DVD later in the process, we had to create the install server using DVD media.

The steps we are going to describe can also be found within *Solaris 10 11/06 Installation Guide: Network-Based Installations* beginning on page 69.

http://docs.sun.com/app/docs/doc/819-6395

1. On the x86 system that is to become the install server, become *superuser* or assume an equivalent role.

   The system must include a DVD-ROM drive and be part of the site's network and naming service. If you use a naming service, the system must also be in the NIS, NIS+, DNS, or LDAP naming service. If you do not use a naming service, you must distribute information about this system by following your site's policies.

2. Insert the Solaris DVD into the system's drive.
3. Create a directory to contain the boot image.

   ```
   # mkdir -p install_dir_path
   ```

   `install_dir_path` specifies the directory where the DVD image is to be copied.

4. Change to the Tools directory on the mounted disc:

   ```
   # cd /cdrom/cdrom0/Solaris_10/Tools
   ```

5. Copy the disc in the drive to the install server's hard disk by using the **`setup_install_server`** command:

   ```
   # ./setup_install_server install_dir_path
   ```

`install_dir_path` specifies the directory where the DVD image is to be copied.

Note: The setup_install_server command indicates whether you have enough disk space available for the Solaris Software disc images. To determine available disk space, use the **`df -kl`** command.

6. Decide if you need to make the install server available for mounting.

   If the install server is on the same subnet as the system to be installed or you are using DHCP, you do not need to create a boot server. Proceed to step 11.

   If the install server is not on the same subnet as the system to be installed and you are not using DHCP, continue to the next step.

7. Verify that the path to the install server's image is shared appropriately.

   ```
   # share | grep install_dir_path
   ```

   `install_dir_path` specifies the directory where the DVD image is to be copied.

   – If the path to the install server's directory is displayed and anon=0 is displayed in the options, proceed to step 11.

   – If the path to the install server's directory is not displayed or you do not have anon=0 in the options, continue to the next step.

8. Make the install server available to the boot server by adding this entry to the /etc/dfs/dfstab file.

   ```
   share -F nfs -o ro,anon=0 -d "install server directory" install_dir_path
   ```

9. Verify that the nfsd daemon is running by listing the status of services:

   ```
   # svcs -l svc:/network/nfs/server:default
   ```

   If the nfsd daemon is not online, start it with the following command:

   ```
   # svcadm enable svc:/network/nfs/server
   ```

10. Share the install server with the following command:

    ```
    # shareall
    ```

11. Change directories to root (/).

    ```
    # cd /
    ```

12. Eject the Solaris DVD.

2.3.2 Setting up a boot server

Next we will create a boot server. Note that we will go back and update the images we just added to our boot server in order to perform an installation to an LS21. The steps we are going to describe can also be found within *Solaris 10 11/06 Installation Guide: Network-Based Installations* beginning on page 74.

http://docs.sun.com/app/docs/doc/819-6395

1. On the system you intend to make the boot server for the subnet, log in and become superuser or assume an equivalent role.

   The system must have access to a remote Solaris 10 11/06 disc image, which is normally the install server. If you use a naming service, the system should also be in a naming service. If you do not use a naming service, you must distribute information about this system by following your site's policies.

2. Mount the Solaris DVD from the install server.

   ```
   # mount -F nfs -o ro server_name:path /mnt
   ```

`server_name:path` is the install server name and absolute path to the disc image.

3. Create a directory for the boot image.

   `# mkdir -p boot_dir_path`

   `boot_dir_path` specifies the directory where the boot software is to be copied.

4. Change to the Tools directory on the Solaris DVD image.

   `# cd /mnt/Solaris_10/Tools`

5. Copy the boot software to the boot server.

   `# ./setup_install_server -b boot_dir_path`

   `-b` specifies to set up the system as a boot server

   `boot_dir_path` specifies the directory where the boot software is to be copied

   Note: The **`setup_install_server`** command indicates whether you have enough disk space available for the images. To determine available disk space, use the **`df -kl`** command.

6. Change directories to root (/).

   `# cd /`

7. Unmount the installation image.

   `# umount /mnt`

2.3.3 Patching the Install Server for LS21/HS21 installations

Now we update the Install Server that we made to incorporate the patches that are needed to install onto servers such as the LS21 and HS21. The following comes from the *Solaris 10 Installation Guide for BladeCenter Servers* under the Troubleshooting section “How to Create an x86 Solaris 10 11/06 OS Installation Image”.

http://docs.sun.com/app/docs/doc/819-5455

Some steps have been updated to properly reflect the driver at time of writing this paper.

Patch the Install Server as follows:

1. Change directories to the Solaris_10/Tools directory on the Solaris 10 11/06 DVD or the network installation image.

   `# cd install-dir-path/Solaris_10/Tools`

   `install-dir-path` specifies the directory where the image is to be copied, for example, `/export/solaris-image`.

   Note: Be aware that the root (/) level directory for the operating system image varies, depending on whether you are installing the Solaris operating system from DVD or CD media, or from a network installation server. The examples in this procedure show how to create an operating system image on a network installation server.

2. Run the **`setup_install_server`** command to create a new installation image.

   `# ./setup_install_server new-install-dir-path`

   `new-install-dir-path` specifies the path on the Solaris 10 11/06 for x86 system in which to create the new installation image.

Note: This command creates a new installation image on the Solaris 10 11/06 for x86 system.

3. Create a writable copy of the miniroot using the following command:

   ```
   # /boot/solaris/bin/root_archive unpackmedia new-install-dir-path
   miniroot-dir-path
   ```

   `miniroot-dir-path` specifies the path to the directory to contain the unpacked boot archive, for example, `/export/sol10-mod-miniroot`.

4. From the SunSolve Web site, download the following patches:

   – 123840-04

   – 118855-36

   – 125034-01

   If you will be using a BladeCenter Storage and I/O (SIO) Expansion Unit (sometimes referred to as a BSE3) attached to your system, you will also need to download:

   – 125325-01

   For all of the above patches, the version number of the patch may have been updated, so the number after the dash may be higher. Ensure you use the most recent patch.

   Note: Some patches are restricted and require a Sun Service Plan or Solaris Subscription to gain access. To obtain a Sun Service Plan, Solaris Subscription, or to get information about downloading and applying recommended patches, go to the SunSolve Web site at:

   `http://sunsolve.sun.com/`

5. Download the latest solaris-x.x.x bnx driver from the Broadcom Web site. We used the solaris-3.0.0 bnx driver. Directions can be found on the Broadcom Web site at:

   `http://www.broadcom.com/support/ethernet_nic/netxtremeii.php`

6. For the BladeCenter LS41, if the bnx driver is Version 3.0.0, or a version prior to Version 3.0.0, you must also download the latest version of the Broadcom bcme driver from the Broadcom Web site.

   Directions for downloading and installing the Broadcom bcme driver can be found on the Broadcom Web site at:

   `http://www.broadcom.com/support/ethernet_nic/netxtreme_server.php`

7. Create a new temporary directory Tmp, then copy the patches and the bnx driver to the new Tmp directory.

   ```
   # mkdir new-install-dir-path /Solaris_10/Tmp
   ```

8. Change directories to the directory where the patches and the bnx driver are located. Unzip the patches and the bnx driver.

   ```
   # cd new-install-dir-path /Solaris_10/Tmp
   # unzip 123840-04.zip
   # unzip 118855-36.zip
   # unzip 125034-01.zip
   # unzip solaris-2.8.5.zip
   ```

9. For the BladeCenter LS41 server, unzip the bcme driver.

   ```
   # unzip sol86-version.zip
   ```

10. Set the PKG_NONABI_SYMLINKS variable to true.

```
# export PKG_NONABI_SYMLINKS=true
```

11. Apply the three patches to the miniroot.

```
# patchadd -C miniroot-dir-path new-install-dir-path/Solaris_10/Tmp/patch-id
```

patch-id specifies the patch ID that you want to apply, for example, 123840–04

12. Repeat the **patchadd** command for each of the patches that you want to apply.
13. Install the bnx driver in the miniroot:

```
# pkgadd -d new-install-dir-path/Solaris_10/Tmp/Server/solaris/GLDv2/BRCMbnx.pkg \
-R miniroot-dir-path#
```

14. If you are using a BladeCenter Storage and I/O (SIO) Expansion Unit in your configuration you will need to add the following to your miniroot.
    a. Edit the file with the command:

    ```
    vi miniroot-dir-path/etc/driver_aliases
    ```

    b. Add the following line anywhere in the file:

    ```
    aac "pci9005,286"
    ```

15. For the BladeCenter LS41 server, if you are installing the Broadcom bcme driver, follow these steps:
    c. Remove the existing bge driver for the Solaris operating system from the miniroot:

    ```
    # rm -f miniroot-dir-path/kernel/drv/bge
    # rm -f miniroot-dir-path/kernel/drv/bge.conf
    # rm -f miniroot-dir-path/kernel/drv/amd64/bge
    ```

    d. Use a text editor to edit the driver_aliases file. Remove all instances of “bge” in the driver_aliases file from the miniroot.

    For example:

    ```
    # vi miniroot-dir-path/etc/driver_aliases
    ```

    e. Install the bcme driver in the miniroot.

    ```
    # pkgadd -d new-install-dir-path/Solaris_10/Tmp/sol86/BRCMbcme.pkg \
    -R miniroot-dir-path
    ```

16. Modify the miniroot's repository.db:

> **Important:** The SVCCFG_REPOSITORY variable *must* point to the location of the unpacked miniroot's repository.db. The repository.db is located in the directory /etc/svc, under the unpacked miniroot. Failure to export this variable results in the modification of the live repository, which prevents the system from booting.

```
# svccfg
svc:> repository miniroot-dir-path/etc/svc/repository.db
svc:> select /system/device/local
svc:/system/device/local> setprop start/exec=:true
svc:/system/device/local> select /milestone/single-user
svc:/milestone/single-user> setprop start/exec=:true
svc:/milestone/single-user> select /system/filesystem/usr
svc:/system/filesystem/usr> setprop start/exec=:true
svc:/system/filesystem/usr> exit
```

17. Repack the miniroot using the following command:

```
# /boot/solaris/bin/root_archive packmedia new-install-dir-path
miniroot-dir-path
```

2.3.4 Creating an updated DVD image and installing onto an LS21

Now that we have a patched Install Server image, there are two methods that can be used to perform an installation:

- Use the patched Install Server to burn a custom DVD that can be used to perform an install. We cover this method in this section.
- Use the patched Install Server as a basis for a Network based PXE boot install. We'll cover this method in 2.3.5, "Creating a DHCP and PXE server" on page 46.

The following comes from the *Solaris 10 Installation Guide for BladeCenter Servers* under the Troubleshooting section "How to Create a DVD ISO Image" and then from portions of "How to Perform an Installation of the Solaris 10 11/06 OS on the IBM BladeCenter LS21 Server".

http://docs.sun.com/app/docs/doc/819-5455

To create a new DVD disk with the patched operating system, do the following:

1. Change directories and run the **mkisofs** command to create a bootable iso9660 file system installation image.

   ```
   # cd /new-install-dir-path
   # mkisofs -o /tmp-dir/solaris-image.iso -b boot/grub/stage2_eltorito \
   -c .catalog -no-emul-boot -boot-load-size 4 -boot-info-table \
   -relaxed-filenames -N -allow-leading-dots -l -r -J -d -D -V \
   SOL_10_U3MOD /new-install-dir-path
   ```

   tmp-dir specifies the directory that is used to temporarily store the DVD image. Note that you need a minimum of 4 GB of free disk space.
2. Copy the ISO image to a system with a DVD writer (or attach a USB one to your system).
3. Create the DVD with the command:

   ```
   # cdrw -iC /tmp-dir/solaris-image.iso
   ```

   Tip: If you have two or more optical drives attached to your system (for example, an internal one and a USB attached drive), then use the -l (list) parameter to list the drives available:

   ```
   # cdrw -l
   ```

   Then use the -d (device) parameter to specify which device to use:

   ```
   # cdrw -d cdrom2 -iC /tmp-dir/solaris-image.iso
   ```

Now we can install the system using the updated DVD media. Here are the differences from a standard installation:

1. When the installation begins, select the **Manual Reboot** option.

   Note: During the installation, you might see the following error message:

   ```
   eeprom: syntax error in /boot/solaris/bootenv.rc line 23
   ```

   You can safely ignore this message.

2. After the installation is finished, before the system reboots, type the appropriate key stroke combination to exit to a command shell.
3. Add the patches and the Broadcom Ethernet device driver package to the newly installed disk.

```
# patchadd -R /a /cdrom/Solaris_10/Tmp/123840-04
# patchadd -R /a /cdrom/Solaris_10/Tmp/118855-36
# patchadd -R /a /cdrom/Solaris_10/Tmp/125034-01
# pkgadd -R /a -d /cdrom/Solaris_10/Tmp/Server/solaris/GLDv2/BRCMbnx.pkg
```

4. If you are using a BladeCenter Storage and I/O (SIO) Expansion Unit in your configuration you will need to add the following to your miniroot.
   a. Edit the miniroot with the command:

      ```
      vi miniroot-dir-path/etc/driver_aliases
      ```

   b. Add the following line anywhere in the file:

      ```
      aac "pci9005,286"
      ```

5. After adding the patches, reboot the system.

2.3.5 Creating a DHCP and PXE server

In this section we describe how to configure a DHCP and PXE server to eliminate the need for boot media.

The steps in this section are based on a feature article within the BigAdmin System Administration Portal titled *Adding Device Drivers to the New Boot Install Image in the Solaris 10 OS for x86 Platforms* available from:

http://www.sun.com/bigadmin/features/articles/device_driver_install.html

This will create a promiscuous PXE and DHCP server. This configuration is set up so that any server on the same subnet that makes a PXE request will be answered by this server and granted a DHCP address that you specify during the course of the installation. If you are setting up a standard DHCP server, or have another DHCP server on the same subnet, you will need to further refine your DHCP server configuration to address this. We will not be covering possible configuration changes that you can make since it's beyond the scope of this paper.

For addition details, see the *Solaris DHCP Administration Guide* available from:

http://docs.sun.com/app/docs/doc/806-5529

To DHCP and PXE, perform the following steps:

1. If a DHCP server is already configured, you can unconfigure it using the dhcpconfig command with the unconfigure flag.

   ```
   # dhcpconfig -Ux
   ```

   This will give us a clean slate to begin with. If you already have a DHCP server on the same subnet, then you can skip this step, but you'll need to modify the script below to not clash with your current DHCP server. Instructions for how to do that would be beyond the scope of this paper:
2. Create a file with the following information to create a generic DHCP server with PXE enabled:

Example 2-1 Script to configure a DHCP server with PXE

```
#!/bin/sh
dhcpconfig -D -r SUNWbinfiles -p /var/dhcp
dhcpconfig -N <network> -m <netmask> -t <routerip>
dhtadm -A -s SrootOpt -d 'Vendor=SUNW.i86pc,1,ASCII,1,0'
dhtadm -A -s SrootIP4 -d 'Vendor=SUNW.i86pc,2,IP,1,1'
dhtadm -A -s SrootNM -d 'Vendor=SUNW.i86pc,3,ASCII,1,0'
dhtadm -A -s SrootPTH -d 'Vendor=SUNW.i86pc,4,ASCII,1,0'
dhtadm -A -s SswapIP4 -d 'Vendor=SUNW.i86pc,5,IP,1,0'
dhtadm -A -s SswapPTH -d 'Vendor=SUNW.i86pc,6,ASCII,1,0'
dhtadm -A -s SbootFIL -d 'Vendor=SUNW.i86pc,7,ASCII,1,0'
dhtadm -A -s Stz -d 'Vendor=SUNW.i86pc,8,ASCII,1,0'
dhtadm -A -s SbootRS -d 'Vendor=SUNW.i86pc,9,NUMBER,2,1'
dhtadm -A -s SinstIP4 -d 'Vendor=SUNW.i86pc,10,IP,1,1'
dhtadm -A -s SinstNM -d 'Vendor=SUNW.i86pc,11,ASCII,1,0'
dhtadm -A -s SinstPTH -d 'Vendor=SUNW.i86pc,12,ASCII,1,0'
dhtadm -A -s SsysidCF -d 'Vendor=SUNW.i86pc,13,ASCII,1,0'
dhtadm -A -s SjumpsCF -d 'Vendor=SUNW.i86pc,14,ASCII,1,0'
dhtadm -A -s Sterm -d 'Vendor=SUNW.i86pc,15,ASCII,1,0'
dhtadm -A -s SbootURI -d 'Vendor=SUNW.i86pc,16,ASCII,1,0'
dhtadm -A -m PXEClient:Arch:00000:UNDI:002001 -d \
':BootFile="nbp.SUNW.i86pc":BootSrvA=<serverip>:'
dhhtadm -A -m SUNW.i86pc -d \
':SinstNM="<server>":SinstIP4=<serverip>:\
SinstPTH="/export/install":SrootNM="<server>":\
SrootIP4=<serverip>:\
SrootPTH="/export/install/Solaris_10/Tools/Boot":'
```

3. Make the following changes to the contents of the above script:
   a. In the third line, replace the following:
      - <network> should be replaced with the network address (for example, 192.168.100.0)
      - <netmask> is your netmask (for example, 255.255.255.0)
      - <routerip> is the IP address of your router (for example, 192.168.100.1)
   b. In the last line of the script, replace <server> in two places with the host name of the install server.
   c. Replace <serverip> in three places with the IP address of the install server.
   d. Modify the SinstPTH parameter in the last line to properly reflect the directory that you used in the creation of the installation server.
   e. Modify the SrootPTH parameter to point to the directory where the /Tools/Boot directory is located.
4. Now that the script has been made, use the **chmod** command to make it executable and run the script as root.
5. Edit the /etc/hosts file and add one or more client entries with an IP address. For two clients, add the following to /etc/hosts:

```
192.168.100.101 pxeclient1
192.168.100.102 pxeclient2
```

Note: This is a read only file, so you will need to either change the permissions on the file before editing (and change it back after the edits are complete) or to force the write.

6. Add the addressed that you put into the /etc/hosts file (Step 5 on page 47) into the DHCP server client table using the pntadm command:

   ```
   # pntadm -A 192.168.100.101 -m <server> -h pxeclient1 <network>
   # pntadm -A 192.168.100.102 -m <server> -h pxeclient2 <network>
   ```

   Note that <server> and <network> have the same meaning for the server and network macros specified by the last line of the PXE configuration script in Figure 2-1 on page 47.

   When you run each command, you will receive the message:

   ```
   Operation succeeded, but hosts table was not updated.
   ```

   This is warning message can be safely ignored. It occurs because the /etc/hosts file is write protected. We manually updated the hosts table in Step 5 on page 47.

7. Send a -HUP signal to the in.dhcp process by issuing the following command.

   ```
   # pkill -HUP in.dhcpd
   ```

   This signal will force the DHCP server to reread its configuration files.

8. To configure the boot files for TFTP, the Solaris operating system provides a command to simplify the creation and copying of all the files to the /tftpboot directory. Simply run the command from the /export/install/images/Solaris_10/Tools directory:

   ```
   # ./add_install_client -d SUNW.i86pc i86pc
   ```

You should now be able to test your client and boot PXE on the Solaris operating system.

From here, an installation onto any server is straight forward. If you boot a blade server that is connected to the same subnet as your Installation Server, and the blade server is configured for PXE based booting, upon completing POST the blade will begin looking for a PXE server. It will find your newly created DHCP/PXE server, and boot from it. The configuration in the above script will point the blade to begin using the installation from the Installation server.

There are a couple of steps that you will need to do once you have started the installation on the client (note that these are the same steps required if you created a custom DVD and used that as your installation media):

1. When the installation begins, select the **Manual Reboot** option.

   Note: During the installation, you might see the following error message:

   ```
   eeprom: syntax error in /boot/solaris/bootenv.rc line 23
   ```

   You can safely ignore this message.

2. After the installation is finished, before the system reboots, type the appropriate key stroke combination to exit to a command shell.
3. Add the patches and the Broadcom Ethernet device driver package to the newly installed disk.

   ```
   # patchadd -R /a /cdrom/Solaris_10/Tmp/123840-04
   # patchadd -R /a /cdrom/Solaris_10/Tmp/118855-36
   # patchadd -R /a /cdrom/Solaris_10/Tmp/125034-01
   # pkgadd -R /a -d /cdrom/Solaris_10/Tmp/Server/solaris/GLDv2/BRCMbnx.pkg
   ```

4. If you are using a BladeCenter Storage and I/O (SIO) Expansion Unit in your configuration you will need to add the following to your miniroot.

   a. Edit the miniroot with the command:

      ```
      vi miniroot-dir-path/etc/driver_aliases
      ```

b. Add the following line anywhere in the file:

   `aac “pci9005,286”`

5. After adding the patches, reboot the system.

3

Management

Important differentiators in the IBM BladeCenter product families is the systems management features, both hardware based and software based. These features make the servers easier to manage, and provide comprehensive alerting and task-based management.

Good overall server management is the key for reducing overall total cost of ownership for entry-level environments up to large high-end enterprise environments.

In this chapter, we explore ways to manage the IBM BladeCenter and Solaris using the following management tools:

- Advanced Management Module
- IBM Director
- ServerGuide™ Scripting Toolkit
- Altiris Deployment Solution
- Advanced Setup Utility
- Sun Management Center
- Sun Connection

Going forward, IBM and Sun are working together to increase the integration of IBM and Sun management tools. In the future, look for enhanced integration of management offerings from IBM and Sun.

3.1 Advanced Management Module

The IBM BladeCenter provides a secure, out-of-band management path that provides several alternatives for accessing and controlling the chassis subsystems. Each chassis contains up to two Advanced Management Modules (AMMs) to provide redundancy in the event that the active AMM must be removed temporarily for service.

The AMM is the chassis focal point for out-of-band management. Each AMM has a 100 Mbps Ethernet link for remote access. User authentication is accomplished via local user records (within the AMM) or a remote LDAP server. Each user can be authorized to perform a subset of operations within a chassis (role-based authentication). For example, a user may be authorized to have access to chassis and blade level of information but restricted from making changes or having access to the storage or networking subsystems.

3.1.1 Chassis management

A single AMM controls the chassis at any one time, with the second AMM available to assume chassis control capability in the event the active MM is removed or otherwise disabled. The AMM communicates with the individual blades and chassis subsystems via redundant internal buses.

The hardware systems management architecture for the BladeCenter chassis is structured around a multi-tiered management concept involving a management module(s) that provides support for all chassis components, and service processors (SPs) that work in conjunction with the management module(s) to provide processor blade management for the blades on which they reside (Figure 3-1).

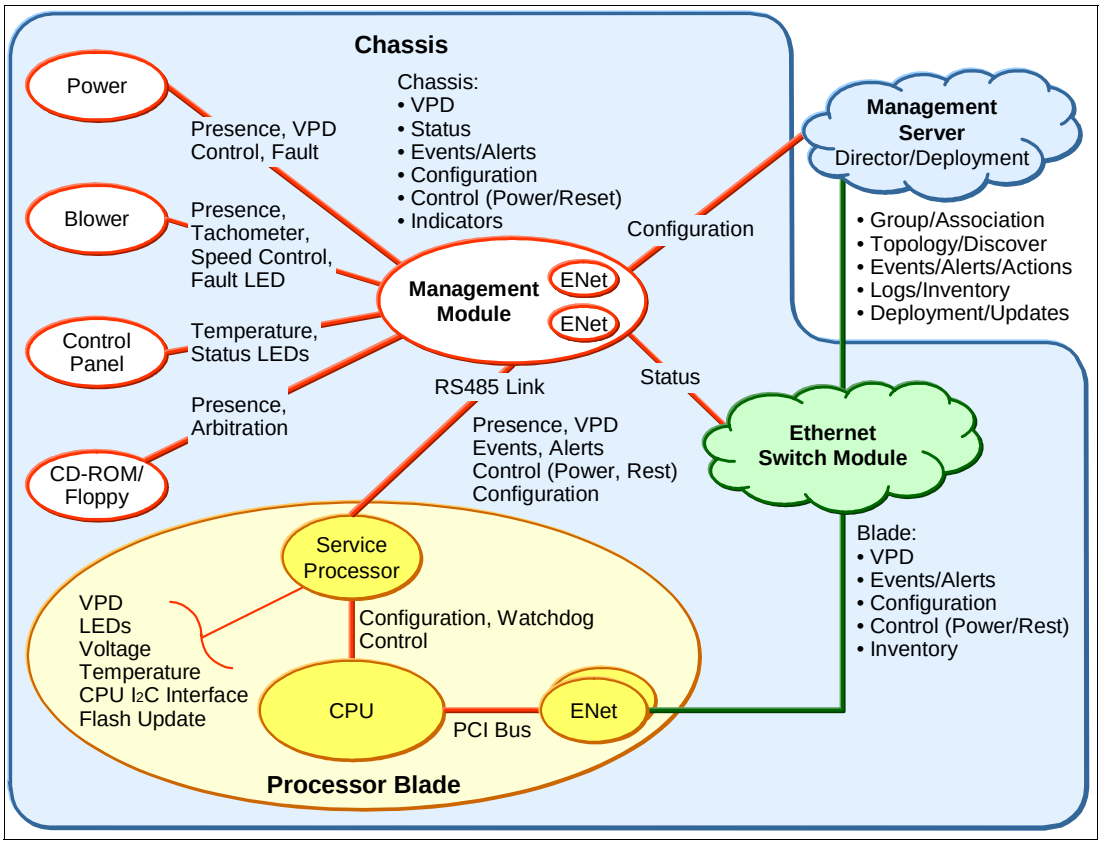


Figure 3-1 Hardware systems management architecture

Chassis level components including power modules, fans, KVM, and optical media. Ethernet and Fibre Channel switches communicate with the management module over I2C busses to provide various levels of control and status. Fans, which have unique signals, are directly connected to the management module.

In addition, the management module supports an Ethernet connection to each switch slot for the purpose of fabric configuration and management. Service processors on each of the processor blades communicate with the management module over the RS485 bus using a command protocol to provide control and status, in addition to supporting normal processor management function.

An external Ethernet link on the management module provides connectivity for remote management, including full console capability (keyboard, video and mouse) with keystroke selection of the target processor blade. PS/2 and video ports on the management module(s) enables local console (keyboard, video and mouse) access to individual processor blades with keystroke selection.

Functions provided by the management module include:

- Chassis configuration - Installed configuration discovery and dynamic configuration change alerts.
- Chassis cooling - Temperature sensing, fan speed control, fan fault indicator, thermal and fan fault alerts.
- Power control - Power module status alerts and reset control, processor blade remote and local power control, and switch module power reset.
- Processor blade initialization - Network compatibility verification, power control authorization, and chassis identification.
- Switch module initialization - Ethernet and Fibre Channel switch configuration
- Media selection - Arbitration for processor blade access to optical and diskette drives.
- Remote console - Remote keyboard, video and mouse access with keystroke target processor blade selection over external Ethernet connection.
- Local console - Local keyboard, video and mouse USB access with keystroke target processor blade selection.
- Display panel - LED control for front and rear display panel.

Functions provided by the processor blade service processor include but are not limited to:

- Power control - Local processor blade power control and power state indication.
- Media control - Request and enable/disable CDROM/Floppy access.
- Keyboard and Mouse control - Control local processor access to keyboard/mouse USB bus.
- Video control - Control local processor access to video bus.
- Thermal sensing - Monitor thermal status and post alerts.
- Management module interface - Communications and command process with management module.

The management module supports hot-plug capability such that if the management module becomes disabled either as the result of a failure on the management module or by removal from the chassis, there will be no disruption of ongoing normal operation of chassis components. Additionally, upon becoming re-enabled, the management module will not disrupt the then current operation or configuration of the chassis components without policy intervention.

Redundant and fail-over operation provides two distinct capabilities; redundant bus support, and management module fail-over.

Redundant bus capability allows the active management module to switch the bus being used to communicate with the chassis components when an I2C or RS485 bus communications problem has been detected.

Fail-over operation is provided when a chassis contains two management modules. When operating in this mode, one management module operates as the active manager and the other is in standby mode. Each management module monitors the state of the other management module within the chassis, and upon detecting a failure of the active module the standby module will take over as the active management module. If the active management module detects a failure of the standby management module, an alert will be posted indicating fail-over operation is inactive. In the event of a fail-over of the active management module, there will be no disruption of the then current operation or configuration of the chassis components. However, some operations, such as Serial Over LAN, may be momentarily disrupted.

Management Modules are capable of detecting the presence, quantity, type, and revision level of each blade, power module, blower, and mid-plane in the system and can detect invalid or unsupported configurations (for example, processor blades with two Ethernet ports connected to Fibre Channel switches.) Also, the MMs play a central role in the power administration of the server blades. Blades can be throttled back to a reduced performance level if the MM detects conditions that could be harmful to the overall chassis.

3.1.2 Management module login

To connect to the Management Module, open a Web browser and enter the Management Module's IP address (default IP address is 192.168.70.125/24). You will be prompted for a user ID and password (default admin userid=USERID and password=PASSW0RD where the O in PASSW0RD is a zero) as in Figure 3-2.

Figure 3-2 AMM login

After you log into the AMM you will be asked to choose an inactivity session timeout value before continuing as in Figure 3-3 on page 55.

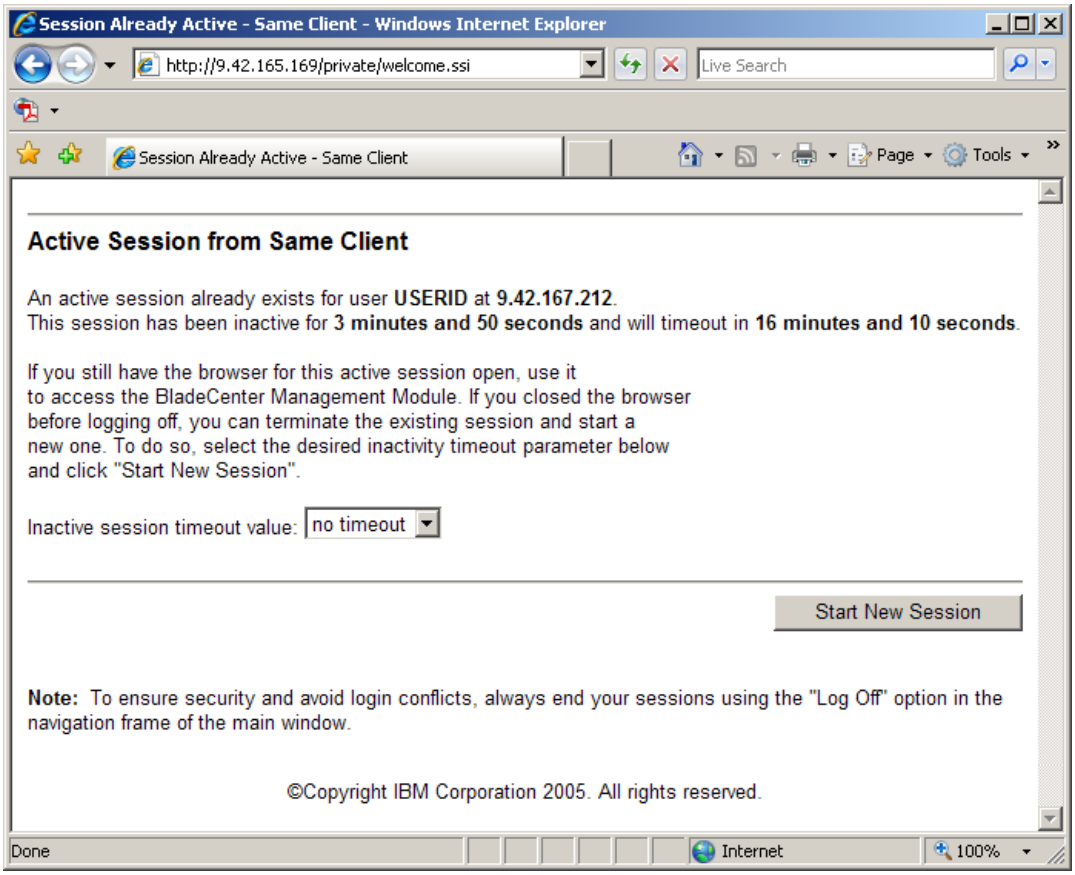


Figure 3-3 AMM inactivity session timeout value selection

Now you will be presented with the AMM main window as in Figure 3-4 on page 56.

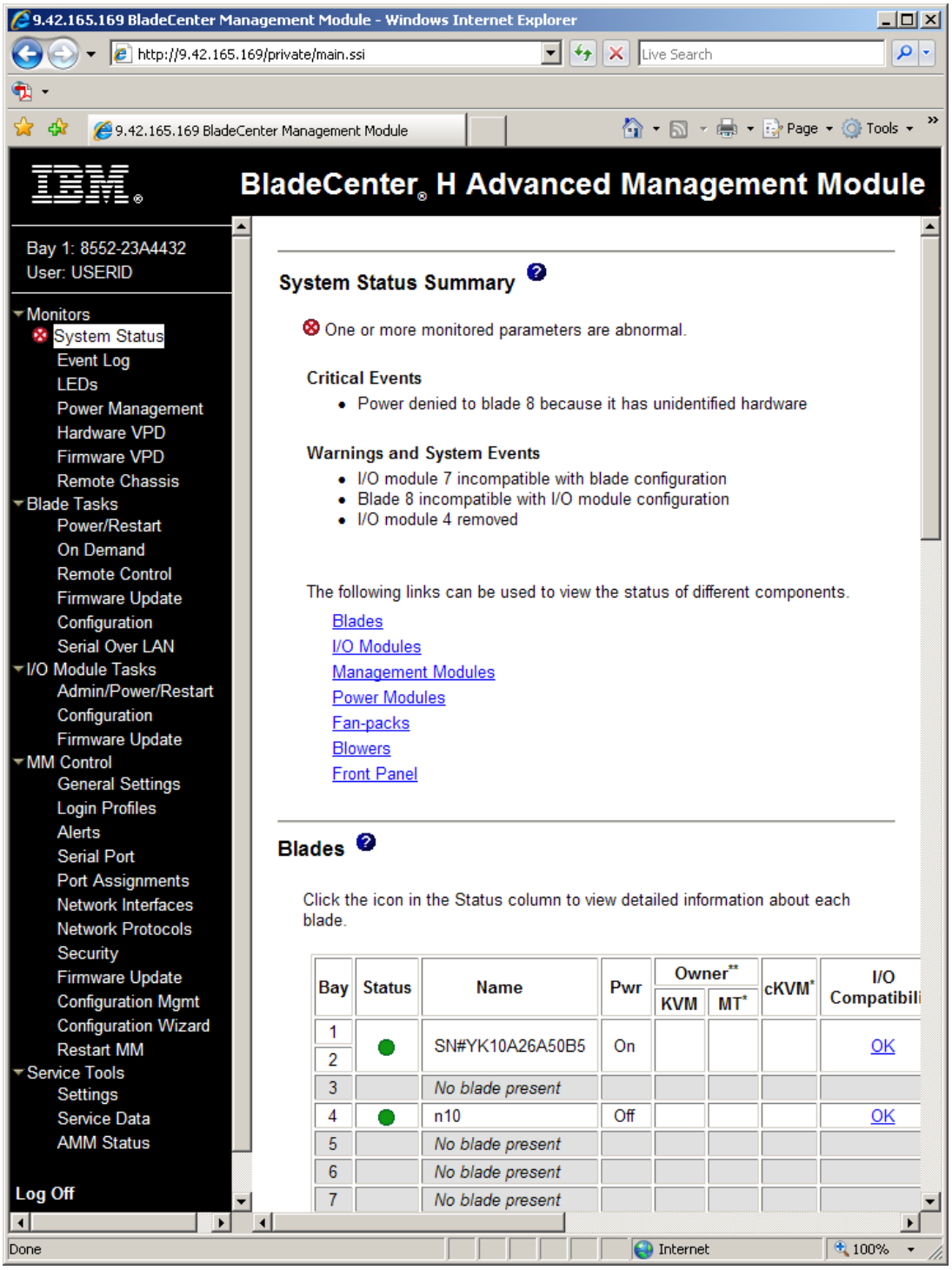


Figure 3-4 Advanced Management Module main window

Remote client software requirements

The management module supports the following Web browsers for remote (client) access:

- Microsoft® Internet Explorer® 5.5 or later (with latest Service Pack installed)
- Firefox 1.07 or later

The client Web browser that you use must be Java-enabled, must support JavaScript™ version 1.2 or later, and must have the Java Virtual Machine (JVM™) plug-in version 1.4.2_08 or later. A suitable JVM plug-in is available from:

http://www.java.com

Note: Even though Solaris does not come with a supported Web browser, it does come with the Mozilla Web browser. Mozilla 1.7 or higher can be used to perform the Remote Control function to an IBM blade server. The limitation is that Remote Disk feature does not work.

3.1.3 Advanced Management Module Command-Line Interface

The IBM BladeCenter management-module command-line interface (CLI) provides direct access to BladeCenter management functions as an alternative to using the Web-based user interface. Using the command-line interface, you can issue commands to control the power and configuration of the management module and other components that are in a BladeCenter unit.

The command-line interface also provides access to the text-console command prompt on each blade server through a serial over LAN (SOL) connection. See the *IBM BladeCenter Serial Over LAN Setup Guide* for information about SOL and setup instructions.

http://www.ibm.com/support/docview.wss?uid=psg1MIGR-54666

You access the management-module CLI by establishing a Telnet connection to the IP address of the management module or through a Secure Shell (SSH) connection. You can initiate connections from the client computer using standard remote communication software; no special programs are required. Users are authenticated by the management module before they can issue commands. You enter commands one at a time; however, you can use command scripting to enter multiple commands. The interface does not support keyboard shortcuts, except for the special key sequence that terminates an SOL session (pressing Esc then "(").

For further details on AMM CLI, see the *IBM BladeCenter Management Module: Command-Line Interface Reference Guide* that is available from:

http://www.ibm.com/support/docview.wss?uid=psg1MIGR-5069717

3.2 IBM Director

IBM Director is an easy-to-use, integrated set of tools used to manage systems on your network. With IBM Director you can view and track the hardware configuration of remote systems in detail and monitor the usage and performance of critical components, such as processors, disks and memory. IBM Director provides a convenient way to manage multiple IBM BladeCenter chassis's and corresponding components. By using the SNMP browsing capabilities in IBM Director you can also monitor and manage Solaris running on the IBM BladeCenter.

IBM Director is provided at no additional charge for use on IBM systems.

Extensions to IBM Director are available for customers who want additional capabilities from a consistent, single point of management.

Here is list of some of the IBM Director extensions that can be useful in a Solaris environment:

- Remote Deployment Manager - can be used to remotely deploy firmware. Firmware deployment is non-concurrent (requires a reboot). Solaris is not currently supported for operating system deployment, however. Fee-based.
- Altiris Deployment Solution for IBM Servers - a third party application that can be used to remotely deploy firmware and Solaris. Firmware deployment is non-concurrent (requires a reboot). Fee-based.
- IBM PowerExecutive - used to monitor and track power consumption. IBM PowerExecutive allows you to trend actual power consumption and corresponding thermal load. Free to download and use.

IBM Director can also be integrated with other popular system management products via its upward integration modules. Listed below are the system management tools supported with upward integration modules:

- Tivoli Management Framework
- Tivoli NetView®
- HP OpenView
- Microsoft SMS
- CA Unicenter
- Microsoft Operations Manager (MOM)
- Altiris
- BMC

IBM Director documentation can be found at the following link:

http://www.ibm.com/systems/management/director/resources/

The IBM Redbooks publication *Implementing IBM Director 5.20*, SG24-6188 is particularly useful for installation and configuration of IBM Director and is available from:

http://www.redbooks.ibm.com/abstracts/sg246188.html

3.2.1 Discovery

The IBM Director management server has the ability to discovery SNMP devices. The device can be a PC host, a printer, a router, or any other device with a listening SNMP agent.

The method for discovery of SNMP devices in IBM Director parallels the method for discovery of native Director Agents. It is not a “ping spray” of IP devices in the local subnet to find agents listening on port 161 but instead uses the IP addresses configured in the Discovery Preferences for SNMP devices as seed addresses and uses the values in the ipNetToMediaNetAddress column of the ipNetToMediaTable variable of the seed addresses MIB as the candidate devices to discover.

The default seed address is that of the server local IBM Director Server host system. However, the best practice is to configure a more practical device as the seed address such as a router or a heavily used server such as an http server that is likely to have a lot of values in its ipNetToMediaTable variable. If IBM Director Server has just been installed, it is likely that there will be few addresses in its ipNetToMediaTable variable, and hence selecting **Discovery** → **SNMP Devices** will discover few devices out of the box. To see the values of the ipNetToMediaTable variable, drag and drop the SNMP Browser task on the Director Server host and drill down to iso.org.dod.internet.mgmt.mib-2.ip.ipNetToMediaTable. ipNetToMediaEntry.ipNetToMediaNet. In this case, either configure a more practical seed

address, or perform a discovery of Director Agents which will broadcast to the local subnet. IP addresses that it reaches will be added to the IpNetToMediaTable variable.

In addition to performing a manual discovery, by default IBM Director provides the ability to add SNMP devices as managed objects on demand. The SNMP Discovery Preferences dialog contain an option that, by default, adds any SNMP devices that sends it a trap to its list of managed objects. This can be turned off if desired by simply unchecking the box.

3.2.2 SNMP browser

The Solaris 10 operating environment has a variety of SNMP interfaces. By default one will find a distribution of the net-snmp agent installed on every installation of Solaris. There is also a more feature-rich solution available at no charge called Sun Management Center (Sun MC).

Using the Sun MC Agent that comes with Sun MC solution you can use the SNMP browser in IBM Director to monitor and manage Solaris systems. The Sun MC Agent is on the Sun Management Center software DVD. The Sun MC DVD ISO image can be downloaded from:

`http://www.sun.com/software/swportfolio/get.jsp`

Once you download the ISO image and mount the ISO file system, to install the Sun MC Agent, use the following command:

```
disk1/sbin/es-inst -a
```

Sun MC installation instructions are in the *Installation and Configuration Guide* and the *User's Guide*. The Sun Management Center Document Collection is available from:

`http://docs.sun.com/app/docs/coll/810.8`

The Sun MC Agent is an SNMP agent that can integrated with the SNMP browser in IBM Director. The SNMP browser does not ship with any Solaris-specific MIBs and will have to be compiled. See 3.2.4, "Compiling MIBs" on page 62 for details on compiling the Solaris MIBs.

The SNMP Browser in IBM Director is used to browse and configure the attributes of SNMP devices such as IBM blade severs running Solaris. You can use the SNMP Browser for SNMP-based management, troubleshooting, or monitoring the performance of SNMP devices.

If you have ever opened the SNMP Browser against a managed system running the SNMP service, you probably saw something similar to Figure 3-5 on page 60. When you have clicked to expand the browser tree often enough to finally reach the end of the line, the information presented hardly seems worth the effort.

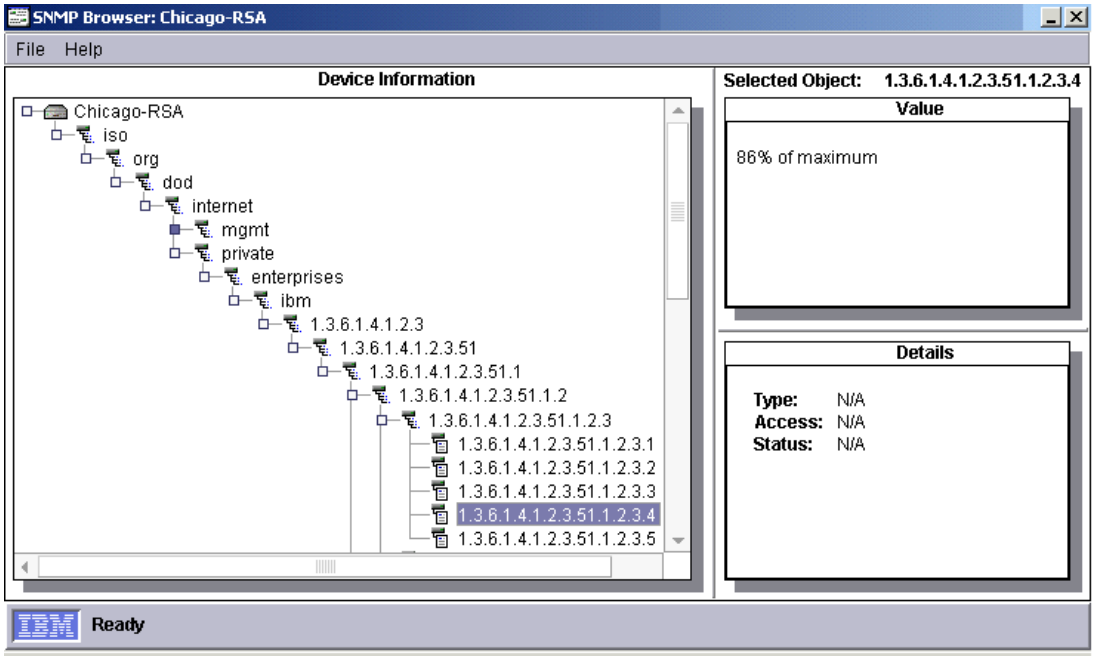


Figure 3-5 SNMP browser showing dotted decimal representation of management data

It is easy to determine that *something* is at 86% of maximum, but unless you happen to know what 1.3.6.1.4.1.2.3.51.1.2.3.4 means, you cannot be sure whether you need to archive files to free up space on a hard drive or consider purchasing an additional processor for this system.

Now take a look at Figure 3-6 on page 61. This is exactly the same view of the SNMP browser targeting the same managed system (actually, a Remote Supervisor Adapter) as in Figure 3-5. This time, we see very specific information about the managed device. The selected item is *fan4speed*, which is much more understandable than 1.3.6.1.4.1.2.3.51.1.2.3.4.

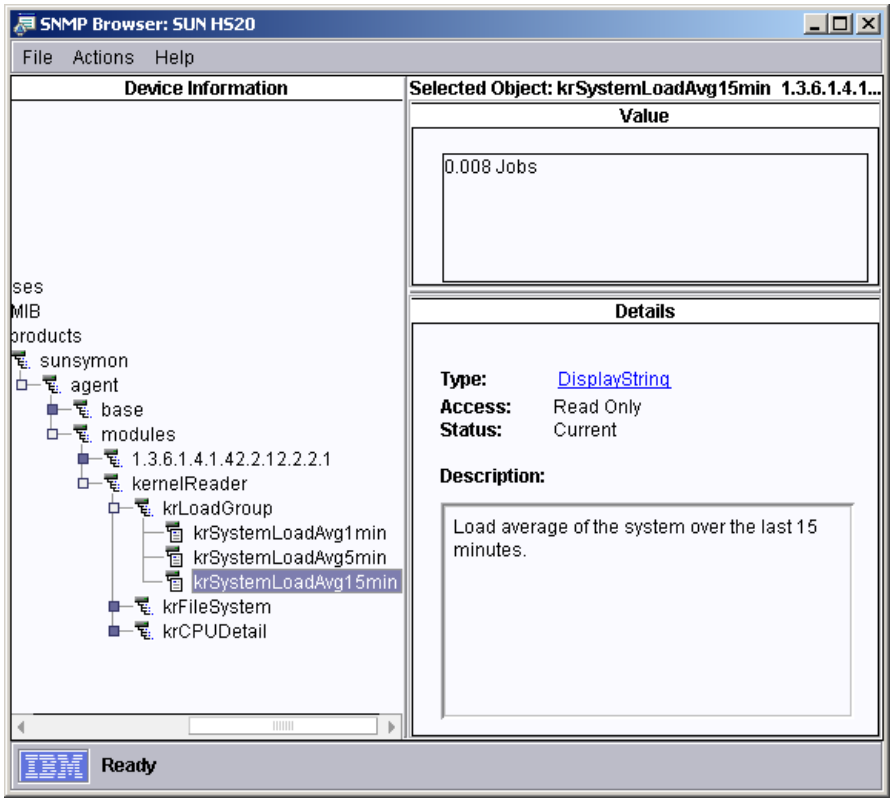


Figure 3-6 SNMP browser after the appropriate MIB has been compiled

In fact, the entire SNMP tree is much more understandable. We quickly and easily recognize specific parameters that are of interest, and the Details pane provides additional information about the parameter selected.

All we have to do to make the SNMP browser more useful is to integrate the appropriate Management Information Base (MIB) file into IBM Director and load it into memory. In IBM Director 5.x, this requires two steps for the SNMP dotted decimal notation to be translated into the more readable format seen in Figure 3-6:

1. Compile the MIB
2. Load the MIB into memory on the management server

The MIB file is used to translate the raw SNMP dotted decimal notation into readable text. This is especially useful for SNMP devices that do not have IBM Director Agent installed, such as network hubs, routers, switches, printers, and uninterruptible power supplies.

In addition, SNMP can be an excellent way to manage systems, especially non-IBM systems, onto which you would rather not install IBM Director Agent. An example might be an HP server that already has the HP Systems Insight Manager agents installed. You might want to manage this system through SNMP, rather than buying a license to install IBM Director Agent on such a system. There is no license fee associated with managing devices with SNMP.

3.2.3 Managing MIBs

IBM Director Server installs with several MIB files located in the Director\proddata\snmp directory.

The SNMPServer.properties file located in the Director\data\snmp directory can be set to automatically recompile any MIBs placed in the Director\proddata\snmp directory each time IBM Director Server service is restarted. To enable this function, open SNMPServer.properties and change the following line to true:

```
snmp.recompileOnStartup=true
```

In previous versions of IBM Director, the order of MIB compilation was often critical for a given set of MIBs. However, the current release seeks out MIB dependencies, then compiles and loads them in the proper order. Of course, if a prerequisite MIB is missing (not in the Director\proddata\snmp directory), then compilation of the dependent MIB fails.

For a MIB to do its job and be useful in the SNMP and Event Log tasks, it must be compiled and loaded into management server memory. In previous versions of IBM Director, all compiled MIBs were loaded into memory automatically. This included all MIB files in the Director\proddata\snmp directory, if the auto-recompile function was enabled.

However, having many MIBs loaded uses up management server resources. In order to reduce the impact of this situation, IBM Director 5.10 separates the compile process from the memory load process. As before, when you compile a MIB file in IBM Director, its mibdata file is loaded into memory automatically. However, you can now choose to unload compiled mibdata files from memory.

The loadedmibs.dat file located in the Director\data\snmp\ directory contains the list of mibdata files most recently requested to be loaded. This is persistent data. The most recent list of loaded mibdata files is stored with each change and reloaded when the management server is restarted. If a mibdata file that has been compiled elsewhere (for example, on another IBM Director Server) is copied into the Director\data\snmp directory, it will not be loaded into memory nor will it be added to the loadmibs.dat file.

The bottom line is that you can have as many compiled MIBs stored on your management server as you want without impacting performance. It is now possible to load any set of mibdata files into memory when you need them, then unload them when you are finished with them.

3.2.4 Compiling MIBs

The SNMP browser does not ship with any SUN specific MIBs and will corresponding have to be compiled. For information about the MIBs that are supported in Solaris, see the *Solaris System Management Agent Administration Guide* that is available from:

http://docs.sun.com/app/docs/doc/817-3000

See the SNMP Browser section for details on installing the Sun MC Agent. Once the Sun MC Agent is installed the MIB files can be found in the /opt/SUNWsymon/util/cfg directory on the system. The following MIBS files from /opt/SUNWsymon/util/cfg directory need to be compiled:

- kernel-reader-mib.mib
- process-details-mib.mib
- process-monitor.mib

Compiling a MIB file creates a binary data file named <MIB definition name>.mibdata in the Director\data\snmp directory. The process of manually compiling a MIB file for IBM Director is as follows:

1. In the IBM Director Console window, click **Tasks** → **SNMP Browser** → **Manage MIBs**.
2. In the MIB Management window, click **File** → **Select MIB to Compile**.

3. In the Select MIB to Compile window (Figure 3-7), specify the directory and file name of the MIB file you want to compile, then click **OK**.

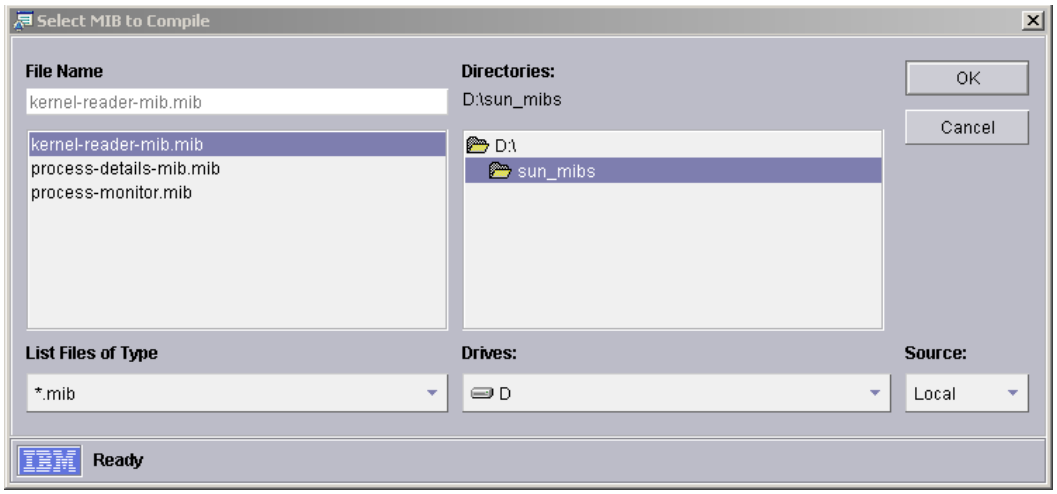


Figure 3-7 Select MIB to Compile window with VMware Root MIB selected

4. You are returned to the MIB Management window, as shown in Figure 3-8. This window provides status information while the MIB compiles and lists any errors encountered in compiling the MIB file.

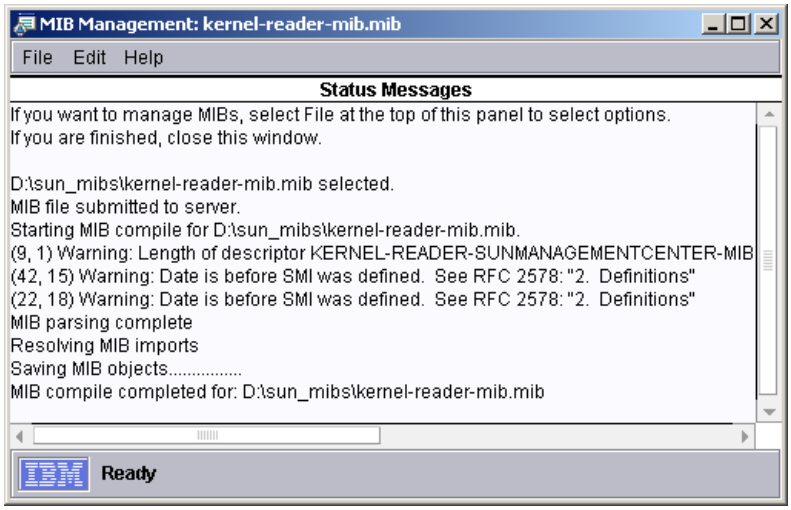


Figure 3-8 MIB Management window shows status of MIB compilation and any errors if they occur

5. After the MIB compilation completes, close the window and return to the SNMP Browser to see more human-readable information for the device you have compiled.

3.2.5 Loading MIBs into memory

As discussed in the previous section, it is now possible to reduce memory utilization on the management server by loading only required mibdata files into memory, while unloading all others. To load or unload mibdata files into management server memory manually, use the following steps.

Tip: When you compile a MIB, it is automatically loaded into memory, so you do not have to load it manually using these instructions.

1. On the IBM Director Console window, click **Tasks** → **SNMP Browser** → **Manage MIBs**.
2. In the MIB Management window, click **File** → **Select MIB to Load**.
3. In the Load MIBs window (Figure 3-9), select the MIBs you want to load from the Available MIBs column. Move them to the Loaded MIBs column by double-clicking individual items or by using Shift-click or Ctrl-click to select multiple items, then clicking **Add**.
4. When you are finished adding MIBs to load, click **OK**.

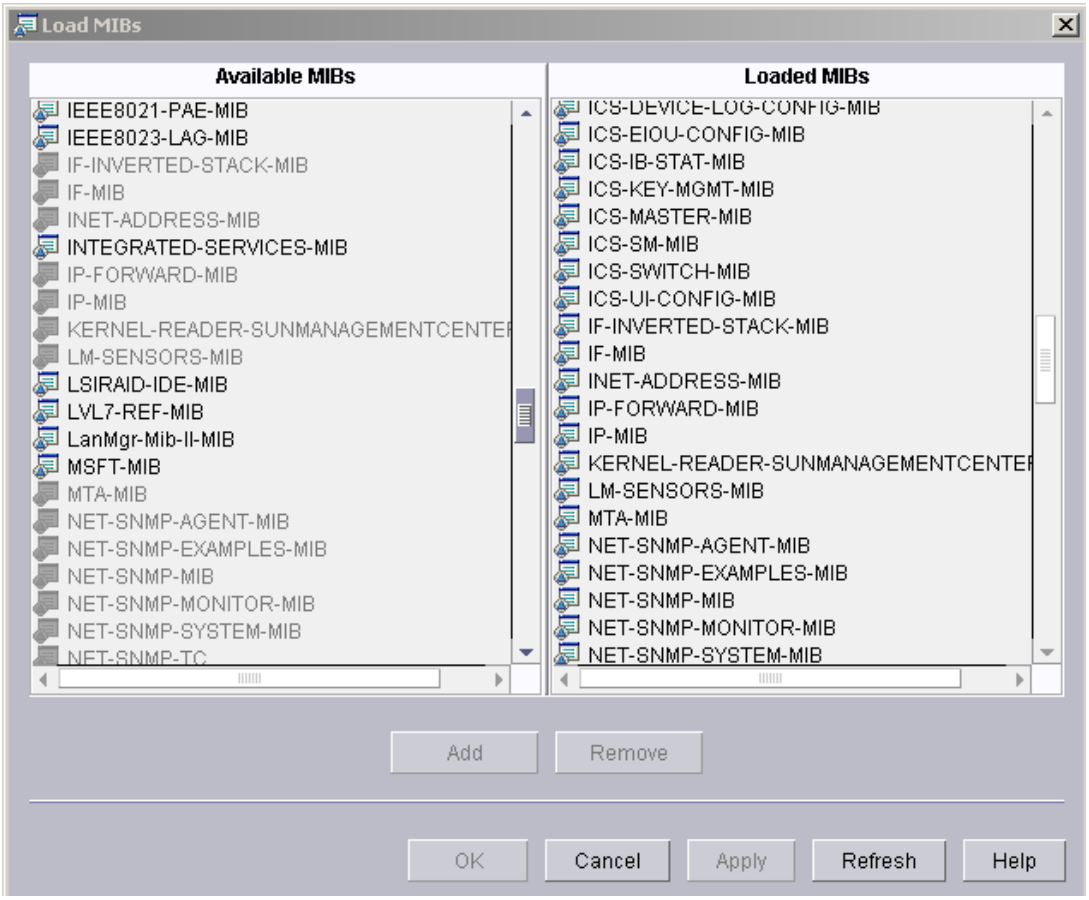


Figure 3-9 Selecting which MIBs to load into management server memory

Any changes made in the Load MIBs window are written to a binary file loadedmibs.dat and stored as persistent data on the management server. Each time the management server is subsequently restarted, it reloads into memory all MIB data files that were most recently specified.

3.2.6 Traps

The Director Server can be used as an SNMP trap listener by configuring the IBM Director management server as a Trap Destination for SNMP devices. This configuration can be done locally through the operating system's SNMP service dialog or remotely using the SNMP Configuration task in IBM Director. This task supports the mass configuration of trap

destination parameters and community names for use against groups of agents. Simply right-click the task and select **Create Profile** to begin this process.

When the Director Server is running, it assumes control of the Microsoft SNMP Trap Service on the local host by default. This means that no other trap listeners running on the system can receive traps.

To turn off this behavior, open the file TWGServer.prop and add the line, then recycle the Director Server application.

```
twg.snmp.trap.listener=false
```

Traps are published to the Director Server by compiling MIBs containing trap definitions. Trap definitions can be recognized by the TRAP-TYPE qualifier in SNMP v1 MIBs or the newer NOTIFICATION-TYPE qualifier in SNMP v2 MIBs. Director 3.x supports SNMP v1 MIBs, Director 4.x supports both SNMP v1 and SNMP v2 MIBs. Alternatively, like other IBM Director events, the trap will be published the first time it is received by the IBM Director management server if the MIB has never been compiled. The traps will appear in the Event Filter Builder under the SNMP node. Follow the iso.org.dod.internet qualifiers to the desired trap definitions.

3.2.7 Monitors

The Resource Monitors task in Director can be applied to managed objects in the SNMP Devices group. By default, only variables under the MIB II branch of the MIB are visible. To change the root OID to some other value, open the file classes\com\tivoli\twg\monitors\RmonSubSys.properties, and comment out the existing line:

```
Root = 1.3.6.1.2.1
```

Replace this line with some other root OID, for example:

```
Root = 1.3.6.1
```

Both individual and group thresholds can be configured for variables in an agent's MIB. Recordings a MIB variable values can also be created.

3.3 Deploying Solaris with the ServerGuide Scripting Toolkit

The ServerGuide Scripting Toolkit is best described as collection tools and sample scripts that can be used to deploy operating systems to IBM System x servers in a predictable and repeatable way.

IBM has released a white paper outlining how to configure the hardware and install the Solaris 10 operating system on IBM BladeCenter servers with the ServerGuide Scripting Toolkit. The white paper titled *IBM ServerGuide Scripting Toolkit; Hardware Configuration and Installation of the Solaris 10 OS on IBM Blade Servers* can be downloaded from:

http://www.ibm.com/support/docview.wss?uid=psg1MIGR-66510

The white paper covers the following:

- Using the ServerGuide Scripting Toolkit to update the system BIOS prior to an operating system installation.
- Using the ServerGuide Scripting Toolkit to configure RAID prior to an operating system installation.

- Using the ServerGuide Scripting Toolkit to configure fibre prior to an operating system installation.
- Performing a Solaris 10 OS JumpStart™ installation.
- Creating and deploying Solaris images using Altiris Deployment Solution. For more details on Altiris Deployment Solution see the Altiris Deployment Solution section.

3.4 Altiris Deployment Solution

Altiris's Deployment Solution is a tool that can be used to remotely install BIOS updates, configure RAID, and configure fibre adapters. In addition it can be used in conjunction with the ServerGuide Scripting Toolkit to deploy Solaris images to IBM BladeCenter servers remotely. For more information about Deployment Solution, see the IBM Redbooks publication, *Deployment using Altiris on IBM System x and BladeCenter Servers*, SG24-7303, available from:

http://www.redbooks.ibm.com/abstracts/sg247303.html

In order for Altiris to deploy Solaris images remotely, a Solaris network install server is required. For more details, see the document *Solaris 10 Installation Guide for IBM BladeCenter Servers* that is available from:

http://docs.sun.com/app/docs/doc/819-5455

3.5 Advanced Setup Utility

The Advanced Settings Utility (ASU) enables you to modify your firmware settings from the command line on multiple operating system platforms. Using the utility, you can modify user preferences and configuration parameters in the BIOS and the service processor firmware without the need to restart the server to access BIOS Setup via the F1 key.

In addition, the Advanced Settings Utility supports scripting environments through its batch processing mode. The utility retrieves and modifies user settings from the supported firmware types using its command-line interface. The utility does not update any of the firmware code.

The base ASU utility and User's Guide can be downloaded from the following location:

http://www.ibm.com/support/docview.wss?uid=psg1MIGR-55021

3.5.1 Using the ASU definition files

The ASU requires a definition file (patch) for each firmware type. The application of the definition file modifies the ASU utility so it will work with the specific hardware. You cannot use the ASU utility until a definition file for that firmware type is applied.

An ASU definition file simply informs the ASU where the settings are located for a single BIOS version so it knows how to apply the settings. The definition file adds data to the end of the utility executable. You can either add or remove the definition file from the ASU, and you can add any number of definition files.

To add a definition to the ASU utility, download the appropriate definition, extract the .def file, and issue the following command to add the definition:

```
./asu patchadd <definition file>.def
```

When ASU runs, it automatically scans the patches that have been applied to it and determines if the applicable definition file exists for the setting that you want. If an applicable definition file exists, the utility applies the setting. If the definition file does not exist, ASU returns an error.

3.5.2 Using the ASU command

You must install the applicable definition files to update ASU files before using the **asu** command, and you must type the ASU commands from the directory in which the utility files are located. The syntax of the ASU command is:

```
./asu [command] [setting] [value]
```

In the following list of commands, *setting* is the parameter that you want to view or change, and *value* is the value that you are placing on the parameter. If *value* contains spaces, enclose the value string with quotation marks.

ASU available commands and their syntax are listed in Table 3-1.

Tip: The -v option lets you specify verbose mode, which provides more detail in the output.

Table 3-1 Available ASU commands and their syntax

ASU command	Function	Syntax
(no command)	Show all the ASU commands and options.	No parameters
batch <filename>	Execute multiple commands.	`batch <filename>` Where <filename> is the text file containing various ASU commands.
comparedefault	Compares current values to default values for one or all settings.	`comparedefault [all | <CMOS setting>] [-v]`
dump	Show RAW CMOS settings.	dump
help	View the online help for one or all settings. For BIOS settings, the help information is the same help that you access when you press F1 during setup.	`help [all | <CMOS setting>]`
loaddefault	Loads default values for one or all settings.	`loaddefault [all | <CMOS setting>] [-v]`
patchadd	Add support for a particular server or device.	`patchadd <.def file> [<patched program>]` Where <.def file> is a CMOS definition file taken from a BIOS build, and <patched program> is the filename of the patched program to write.
patchextract	Extracts a definition file from the utility to a patch file. You can patch the extracted definition file to another version of the utility.	`patchextract <patch #> <.def file>` Where <patch #> is the index output by the **patchlist** command, and <.def file> is the file you want to create.
patchlist	Displays the definition files that are currently applied.	No parameters

ASU command	Function	Syntax
patchremove	Removes a definition file.	patchremove <patch #> [<unpatched program>] Where <patch #> is the index output by the **patchlist** command and <unpatched program> is the filename of the unpatched program to write.
rebootrsa	Restart the service processor. This is useful when you must restart a Remote Supervisor Adapter to bring into effect any configuration changes.	No parameters
resetrsa	Reset the RSAI/RSAII back to defaults.	No parameters
replicate	Use the output of one or more **show** commands to set multiple settings at the same time.	replicate <filename> Where <filename> is a file created by piping output from the **show** command.
set	Change the value of a setting.	set <CMOS setting> <value> [-v] Where <value> is a string shown from showvalues.
show	Display the current value of one or all settings.	show [all | <CMOS setting>] [-v]
showdefault	Display the default value for one or all settings.	showdefault [all | <CMOS setting>] [-v]
showvalues	List all possible values for one or all settings. This is useful for finding the value parameter used for the **set** command.	showvalues [all | <CMOS setting>] [-v]
version	Display the version and build date of the utility.	No parameters

3.6 Sun Management Center

For the most feature rich experience on BladeCenter running Solaris, use a combination of IBM and Sun infrastructure management tools. Many data centers have used a combination of IBM Director, IBM Tivoli, Sun Management Center, Solaris Container manager, and Sun Connection to achieve an operational best practice and IT Infrastructure Library® (ITIL®) compliance.

BladeCenter administrators will be able to passively monitor the most important application and operating system specific fault attributes across the Solaris environment without impacting the business propose of the system through the Sun Management Center (Sun MC) product line. In addition, through Sun Connection they will be able to control the system ensuring that their business is always 100% protected and complaint with the environment's approved configurations, Sun suggested patch clusters, Mitre CAN security advisories, or any other predetermined software configuration guidelines. Combine those features with the raw hardware management of IBM Director, and the data center will inherit many cost savings during the day-to-day tasks of the IT administration staff.

The Sun MC software platform is an enterprise management solution that is based on open standards such as SNMP, Java, and Tcl. The software has a rich set of features that enables the complete modeling of Solaris software solutions. Administrators will use this application to help ensure all Solaris and the services they provide are meeting service level agreements.

Administrators will be able to perform remote system management, monitor performance, and software faults. It has an easy-to-use Java or Web interface, it includes a database, a standards-based agent, a fully documented API set that allows for the imputing and extracting of content, and a fast deployment method.

Figure 3-10 shows and example of the Sun MC interface. This is an example of the product's report and trending feature. There are also metric alarms and event notifications features available.

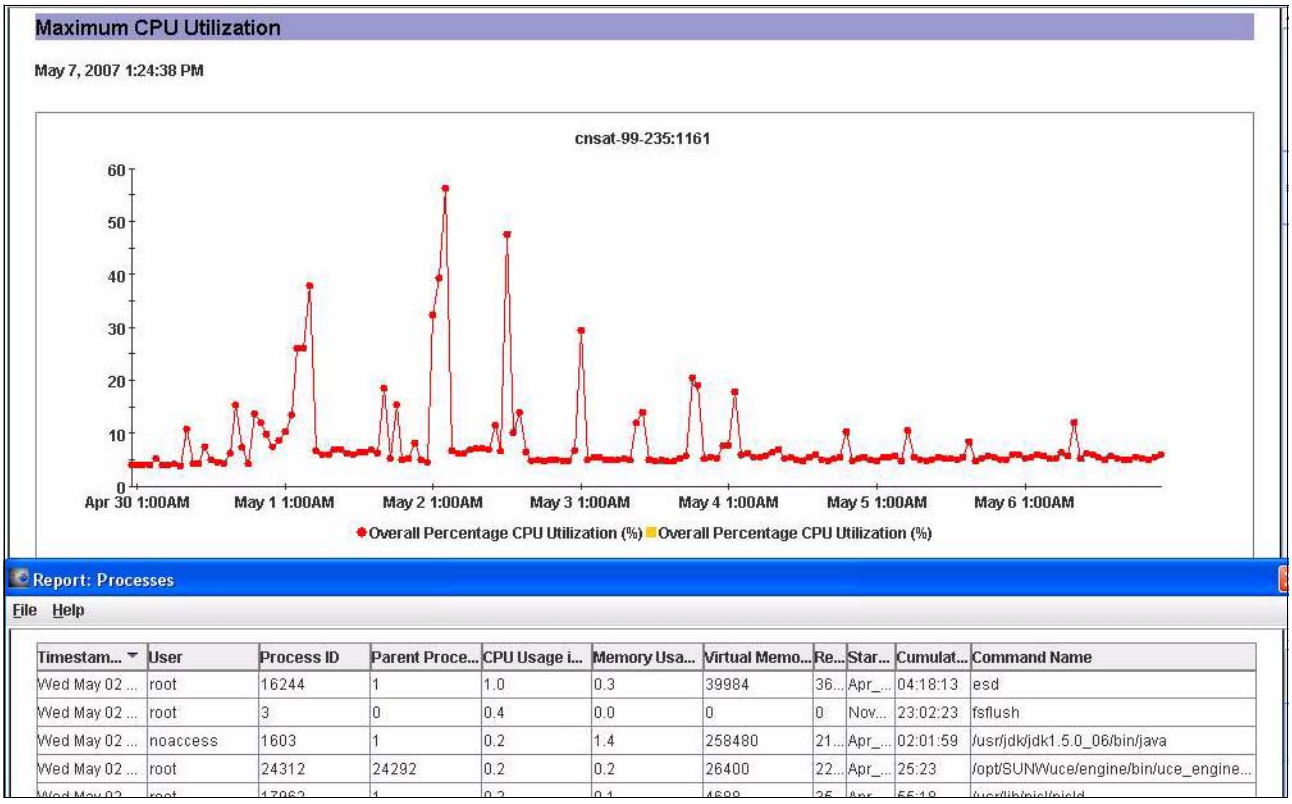


Figure 3-10 Sun Management Console

Solaris 10 specific features are made visible to the administrator, including attributes and alerts from the Solaris Fault Management Architecture (FMA), Service Management Facility (SMF), and other new Solaris 10 attributes. These attributes and their default alarm rules represent Solaris knowledge and awareness that is simple to integrate into higher tier monitoring products such as IBM Tivoli and Director.

Figure 3-11 on page 70 shows a small sample of the over one hundred attributes visible in the Sun MC solution.

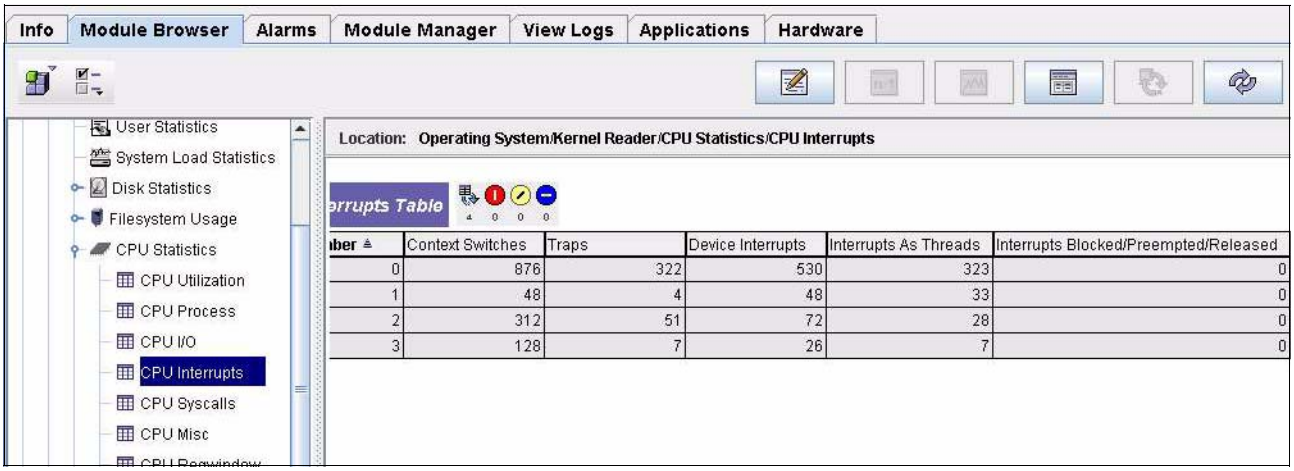


Figure 3-11 Sun Management Console

Solaris Container Manager, as shown in Figure 3-12 and Figure 3-13 on page 71, provides active control over all Solaris resource technologies such as projects, Inodes, zones, and pools. It also does proactive alerting when utilization goals or operational status of these components are not meet. Figure 3-12 shows the host to virtual entity/resource relationship building, a trend of zone specific utilization, and active provisioning, boot control, and migration of the Solaris zone constructs.

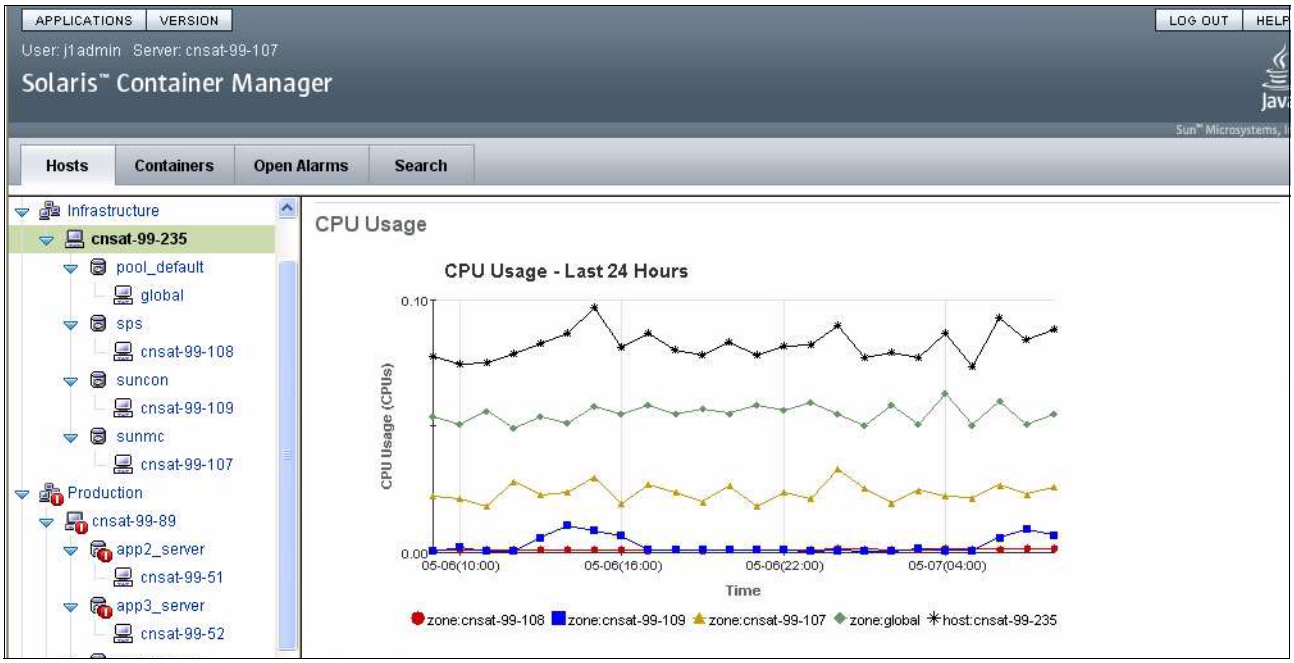


Figure 3-12 Solaris Container Manager

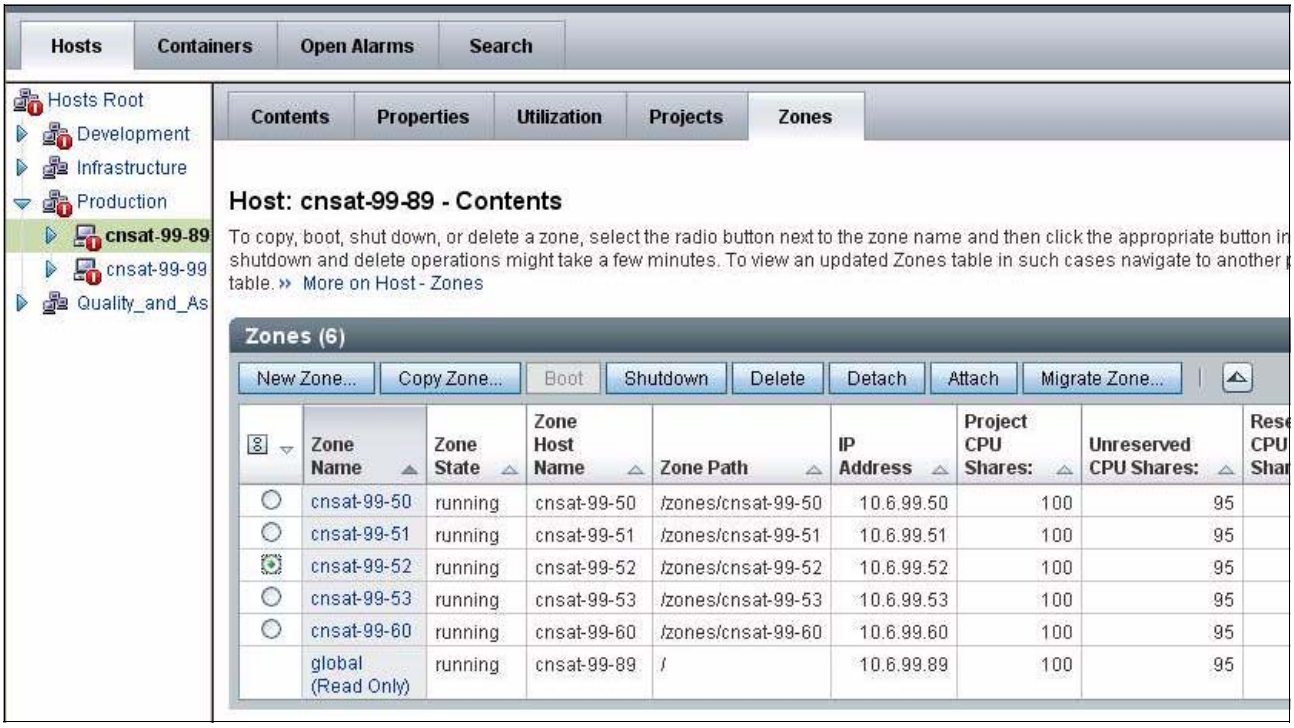


Figure 3-13 Solaris Container Manager

3.7 Sun Connection

BladeCenter administrators can use Sun Connection to help automate patch life cycle management and maintenance. With this method, information is generated from the inspection of software packages and patches down to the binary level instead of relaying on the traditional patch readme or dependency declarations. This “active dependency” information is published as a Web service, operated through a toolset that automates installation, simulation, rollback, compliance checking, and reporting. These capabilities can be used for change management, configuration management, and provisioning.

Figure 3-14 on page 72 shows how by working with Sun to establish the most efficient software configuration, you can experience extreme time and cost savings. Sun Connection on Solaris offers the ability to deploy custom patch baselines. It offers rollback, clone, and other provisioning methods to insure compliancy with application test specification of the development staff or the security requirements of the system administrators.

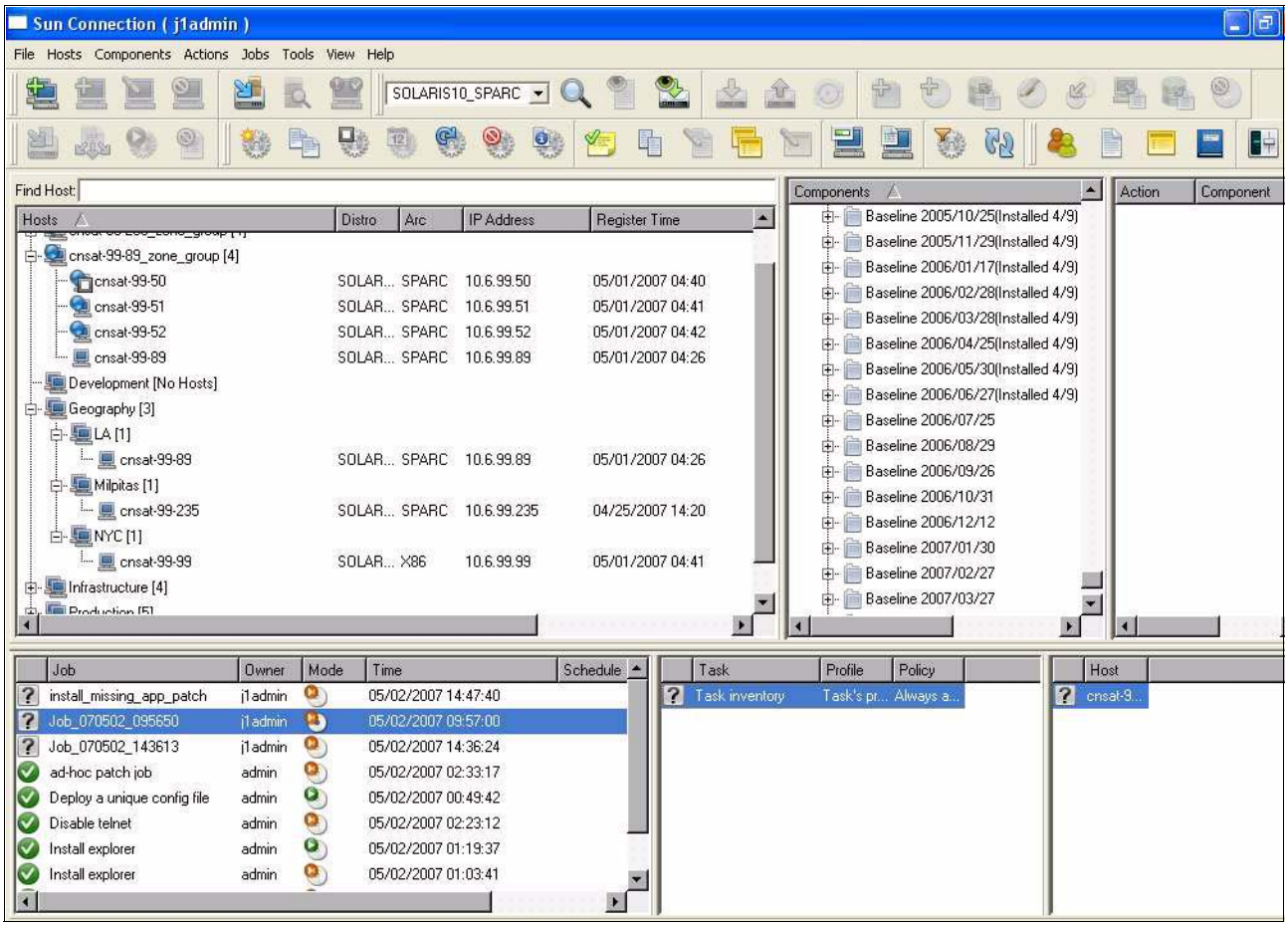


Figure 3-14 Sun Connection on Solaris

To learn more about Sun Connection, visit:

`http://sun.com/service/sunconnection`

To learn more about Sun Management Center, visit:

`http://docs.sun.com/app/docs/coll/810.8`

4

Storage

Server blades running Sun Solaris have a number of disk storage resources available, both on-board the blade server as well as externally attached storage via Fibre Channel. In this chapter, we describe some of these options.

First, we discuss the use of ServeRAID™ adapters for improved resiliency and redundancy for local disk storage. Next we discuss external storage provided by enterprise and non-enterprise disk storage systems, the IBM DS8000™ Storage Subsystem family and the IBM DS4000 Storage Subsystem family, respectively.

4.1 LSI RAID and ServeRAID

ServeRAID adapters offer a number of features to give you flexibility in configuring the disk subsystem. RAID (redundant array of independent disks) is a technology that groups several disk drives into an array that can be defined as one or more logical volumes. Each logical volume then appears to the operating system as a single physical drive. The ServeRAID controller is able to transfer data to and from these multiple disk drives in parallel, thereby yielding much higher data transfer rates than that of a single disk. In addition, some RAID configurations will tolerate the failure of one, or even two, disks without the loss of data.

ServeRAID adapters allow the RAID arrays and logical volumes to span multiple SCSI channels within a single adapter. This ability allows for larger logical drive capacities and potentially greater performance levels as the I/O requests can be distributed across SCSI channels. There are several different RAID levels to use when creating the logical volumes. Each different RAID level provides different performance, fault-tolerant, and disk space efficiency characteristics.

ServeRAID adapters provide RAID support at the hardware level. RAID functionality could be provided by the operating system or by third-party utilities. However, hardware RAID has the following advantages over software RIAD:

- All RAID calculations are performed by the adapter and place no load on the system processor.
- All read and write accesses to redundant information, such as mirrored data, are transparent to the operating system.
- Hot-pluggable disk drives are not supported by software RAID.
- Software RAID recovery procedures after a disk drive failure are more complex.

You can download the latest ServeRAID driver code and find additional documentation from the following page:

`http://www.ibm.com/support/docview.wss?uid=psg1MIGR-61707`

This page includes links to the README file, change history and an ISO image of the IBM ServeRAID Application CD. The ServeRAID Application CD is a non-bootable CD containing tools and utilities for use with IBM ServeRAID adapters and controllers. For Solaris 10, ServeRAID Manager 8.40, or higher, is required to configure, administer and monitor RAID controllers.

The base Solaris operating system includes support for SCSI-based ServeRAID controllers. Unfortunately, the native functionality does not support the SAS-based ServeRAID controllers in BladeCenter server, meaning that drivers will need to be patched into the operating system through the use of the Installation Server.

Sun has plans to incorporate the latest ServeRAID driver for SAS-based controllers in the next release of Solaris 10. Until then, it is possible to incorporate patches into a Solaris 10 10/06 installation to enable booting from a ServeRAID-based blade server. However, at the time of writing, they were not yet released but were expected to be incorporated into the Troubleshooting Installation Issues section of document *Solaris 10 1/06 Installation Guide for IBM BladeCenter Servers* which can be found at:

`http://docs.sun.com/app/docs/doc/819-5455`

4.2 Fibre Channel

Solaris-based BladeCenter modules have the ability to connect to external storage resources via Fibre Channel connections available to the IBM BladeCenter chassis. Fibre channel port adapters are needed on the server module and the external connection ports out of the chassis using either Fibre Channel switch modules or optical pass-thru module (OPM) on the back side of the chassis. IBM offers a variety of storage options and we will briefly cover the following topics:

4.2.1 Attaching to DS4000 storage

IBM DS4000 Storage Servers support the large and growing data storage requirements of business-critical application. These systems are scalable and have redundant controllers for reliability and enhanced data access. The DS4000 supports either direct attachment of up to four hosts or Fibre Channel switches for a significantly larger number of SAN-attached hosts.

Prior to attaching a DS4000 to the Solaris blade server, you should get the DS4000 code and firmware to the latest levels to take full advantage of the constant improvements and bug fixes.

The latest code levels are available by selecting the Product Support link from the appropriate DS4000 model at:

`http://www.ibm.com/servers/storage/disk/ds4000`

To be notified of important product updates, you can register at the IBM Support and Download Web site:

`http://www.ibm.com/support/us/`

In the Personalized Support box of the Web page, click **My Support**. On the next page, if you have not already done so, register to use the site by clicking **Register now**.

Perform the following steps to receive product updates:

1. After you have registered, type your user ID and password to log into the site.
2. When the My support page opens, click **Add products** for a pull-down menu and select **Storage**.
3. In the subsequent pull-down menus, you can select menu topics for specific products for which you would like to receive information. Then click **Add products** to save your selections and return to the My Support page.
4. On the My Support page, click the **Edit profile** tab, and click **Subscribe to email**.
5. Place a check in the box for any topics that you may be interested in as well as the **Please send these documents by weekly email**.

Introduction

The DS4000 storage disk subsystem consists of a storage server enclosure and one or more storage expansion enclosures. The storage server is a rack-mountable enclosure, or a drawer, which has two RAID controller units, redundant power supplies and fans for high availability. Each RAID controller unit has battery-backed cache to boost performance and improve reliability.

Each DS4000 Storage Subsystem contains several removable components, called customer replaceable units (CRUs), that can be accessed from either the front or the back of the unit. These CRUs include the battery unit, RAID controllers, fans, power supplies, communications modules and mini-hubs for Fibre Channel connectivity.

Total storage capacity can be scaled with the use of external storage expansion enclosures continuing Fibre Channel (FC) or Serial Advanced Technology Attachment (SATA) hard drives. Each expansion enclosure is attached via fiber optic cabling using the drive side mini-hubs. The IBM DS4000 series product's interoperability matrix contains the latest information about maximum capacity, operating system support and much more. This document is available from:

http://www.ibm.com/servers/storage/disk/ds4000/pdf/interop-matrix.pdf

On the host side, each RAID controller has at least one mini-hub for Fibre Channel connectivity. Depending on the model-type of DS4000, each controller has a maximum of up to eight host ports (two GBICs per mini-hub). One the drive side, each RAID controller is attached to four Fibre Channel loops, which are normally used as two redundant loops. Each redundant loop attaches to the disk expansion enclosures.

Requirements for installation

Connecting a DS4000 storage system to a Solaris host may require optional licensing or a Sun Solaris Host Kit. Refer to the latest readme file for your specific model of DS4000 storage system for requirements. The latest readme files can be found from:

http://www.ibm.com/servers/storage/support/disk

If Solaris host is to be used for configuring the DS4000 storage system, then certain minimum levels of firmware are strongly recommended. Refer to the latest readme file for the suggested firmware level by DS4000 model.

Installing host software

A failover driver monitors I/O paths. If a component failure occurs in one of the Fibre Channel paths, the failover driver reroutes all I/O to another path. Solaris host systems require one of the following failover drivers:

- Solaris Multiplexed I/O (MPxIO)
- Veritas Volume Manager with Dynamic Multipathing (DMP)

The functionality of MOxIO and DMP are very similar in that they assume a primary path to each volume and do not move to the alternate path until there is a problem with primary path access. Both of these failover drivers are substitutes for each other and are not complementary products. Concurrent use of MPxIO with DMP on multiple DS4000 storage systems is not supported.

We discuss each of these in the following sections.

Solaris Multiplexed I/O (MPxIO)

Multiplexed I/O (MPxIO) is a Sun Solaris multipath driver architecture. This failover driver enables storage arrays to be accessed through multiple host controller interfaces from a single instance of the storage array. MPxIO helps protect against storage array outages because of controller failures. If one controller fails, MPxIO automatically switches to an alternate controller.

MPxIO is fully integrated within the Solaris 10 operating system., and does not need to be installed separately. For the latest Solaris kernel patches and most recent updates to information about using MPxIO, refer to the document *Solaris Fibre Channel Storage*

Configuration and Multipathing Administration Guide (Sun publication number 819-0139) available for downloads or an HTML document from:

http://docs.sun.com/app/docs/doc/819-0139

Note: It is recommended that you install the regular kernel jumbo patch, because there are dependencies between the various patches that make up the driver stack.

When MPxIO is enabled, devices in the /dev and /devices trees are named differently from their original names. For example. a device name with MPxIO disabled would be:

```
/dev/dsk/c1t1d0s0
```

Once MPxIO is enabled, the device name would look similar to:

```
/dev/rdsk/c1t201400A0B811804Ad31s0
```

Therefore, you must configure applications that directly access devices to use the new names whenever the MPxIO configuration is enabled or disabled. In addition, the /etc/vfstab file and the dump configuration also contain references to device names. When using the **stmsboot** command to enable or disable MPxIO, /etc/vfstab and the dump configuration are automatically updated with the new device names.

Veritas Volume Manager with Dynamic Multiplathing (DMP)

The Dynamic Multipathing (DMP) feature of the Veritas Volume Manager provides a path failover mechanism. In the event of the loss of one connection to the disk array, DMP selects the next I/O path for the I/O requests dynamically. This allows the host system to continue to access critical data over the other functioning connections to the disk array. DMP allows you to have up to 256 LUNs.

Veritas VxVM provides Dynamic MultiPathing (DMP) support for disk arrays in the form of Array Support Library (ASL) software packages. ASL packages can be obtained from:

- The VxVM release CD-ROM in the veritas_enabled directory
- The Veritas Technical Support Web site:

  http://support.veritas.com

  Perform a Knowledge Base Search and enter **asl** followed by particular DS4000 storage system in your configuration.

Veritas Volume Manager 4.1, or higher, is supported with Solaris 10. The appropriate ASL for DS4000 storage systems, called the SMIBMASL package, is necessary to enable Solaris to recognize and correctly work with the DS4000 storage systems. Prior to DMP installation the following prerequisites should be verified:

- HBAs are installed on the Solaris host and parameter settings in the HBA configuration file are modified appropriately.
- HBA bindings are configured.
- Fabric zoning is created and enabled for the Solaris partition.
- Storage is mapped to the Solaris partition.

Once the prerequisites have been verified, the installation process can begin. An overview of the installation process is:

1. Prepare the host for Veritas DMP installation.
2. Install Veritas Volume Manager packages for DMP.

3. Install the SMibmasi software package.
4. Configure Veritas Volume Manager.
5. Start the Veritas Enterprise Administrator.

Preparations on the host typically involve manually defining the SAN-attached targets and LUNs in the /kernel/drv/sd.conf file. Each target represents a controller to a disk subsystem and each LUN represents a logical drive. If additional target or LUN definitions are added to the /kernel/drv/sd.conf file, once any edits are saved, you should reboot the Solaris host.

Additional information and detailed configuration and installation instructions can be found in the document *Configuring IBM TotalStorage FAStT for use with Veritas Dynamic Multi-Pathing DMP* that is available from:

`http://www.ibm.com/support/docview.wss?uid=psg1MIGR-57485`

Creating direct-attached and SAN-attached configurations

DS4000 storage systems are supported in direct-attached Solaris configurations or in a SAN environment through switches in Solaris configurations. In a direct-attached configuration, one or two Solaris servers can be connected to a single DS4000 storage system. Each Solaris server has two HBAs where each HBA is connected to a different controller.

For SAN-attached configurations, there are several requirements to follow:

- Multiple HBAs within the same server must be isolated from each other if they are connected to the same switch that is connected to the same DS4000 controller port.
- Each HBA and controller port must be in its own fabric zone, if they are connecting through a single switch.

When connecting Fibre Channel host bus adapters in host servers to DS4000 Storage systems in a Fibre Channel switch environment, IBM recommends that you establish one-to-one zones between the HBA and DS4000 host ports. Figure 4-1 on page 79 illustrates the SAN attached environment and zoning used for this publication.

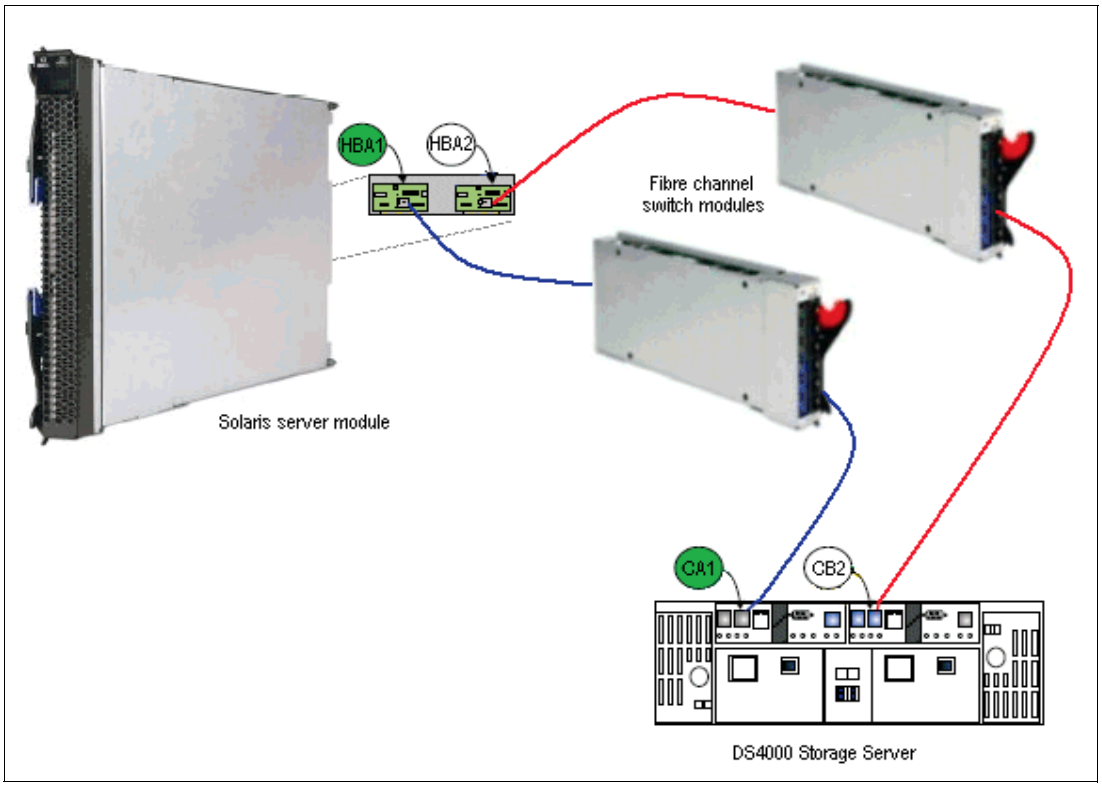


Figure 4-1 DS4000 SAN environment and zones

4.2.2 Attaching to DS8000 storage

The IBM System Storage™ DS8000 series provides extensive support for Sun Solaris 10 on a variety of platforms. It also supports Veritas Cluster Server and Sun Cluster. The most up-to-date information about support configurations, including information about support host bus adapters, SAN switches, and Multipathing technologies in the Interoperability Matrix can be found at:

`http://www.ibm.com/systems/storage/disk/ds8000/interop.pdf`

A great deal of useful information is available in the *IBM System storage DS8000 Host Systems Attachment Guide*, IBM publication number SC26-7917, that is available from:

`http://www.ibm.com/support/docview.wss?uid=ssg1S7001161`

At this Web site, you should specify **`System Storage`** and then select any DS8000 model to go to the product support page. On the product support Web page, select the **`Install/use`** tab to find the link to the referenced document.

Introduction

The IBM System Storage DS8000 is a high performance, high capacity series of disk storage that is designed to support continuous operations. DS8000 systems use the IBM POWER5™ server technology that is integrated with the IBM Virtualization Engine™ technology. DS8000 models consist of a storage unit and one or two management consoles, two being the recommended configuration. The DS8000 had a graphical user interface (GUI) or a command-line interface (CLI) to logically partition storage and configure advanced copy functions.

Models of the DS8100 feature a dual two-way processor complex and support for one expansion frame. Models of the DS8300 feature a dual four-way processor complex and support for up to two expansion frames. Some models also provide support for two logical partitions (LPARs) in one storage unit. Additional information about available features and specifications can be found at:

http://www.ibm.com/systems/storage/disk/ds8000

The IBM System Storage hardware management console (HMC) is the focal point for configuration, copy services management, and maintenance activities. The HMC is a dedicated workstation that monitors the DS8000 system. There can be a maximum of two HMCs with the first management console physically located inside the DS8000. The second HMC can be either internal or external to the DS8000 system.

Requirements for installation

There are a number of general requirements that should be verified before attaching a Solaris host to a DS8000 storage system. These items are:

- Check the DS8000 Interoperability Matrix at for the most up-to-date information and details about the Solaris host and adapter cards.

  http://www.ibm.com/systems/storage/disk/ds8000/interop.pdf

- Refer to the most current listing of supported host bus adapters (HBAs), firmware and device driver information:

  http://www.ibm.com/servers/storage/support/config/hba

- Review the I/O adapter features of the *IBM System Storage DS8000 Introduction and Planning Guide*, publication number GC35-0515, for detailed HBA configuration rules, cabling information, and load balancing for optimal performance. This document is available from:

  http://www.ibm.com/support/docview.wss?uid=ssg1S7001073

- Define the host and HBA port configurations on the DS8000 storage system.

For Fibre Channel attachment, LUNs have an affinity to the world-wide port name (WWPN) of the host's HBA port. In a switched fabric, an individual HBA port could have access to multiple DS8000 storage system ports. In these situations, the storage system can be configured to allow the Solaris host to use some or all of the accessible Fibre Channel ports on the storage unit which requires the use of a multipathing utility.

Table 4-1 provides the minimum revision level that is require for each Solaris system element.

Table 4-1 Minimum patch levels and system elements for multipathing

Revision Level	System Element
108528-03	kernel update
109524-02	ssd driver
109657-01	isp driver
108974-03	sd, uata drivers

Installing host software

As with other operating systems, you should use multiple paths between the DS8000 and your Solaris server. Multiple paths help to maximize the reliability and performance of your operating environment. The DS8000 supports three multipathing technologies on Solaris and those are:

- Solaris Multiplexed I/O (MPxIO)
- System Storage Multipath Subsystem Device Driver (SDD)
- Veritas Volume Manager with Dynamic Multipathing (DMP)

Sun Solaris contains native multipathing software called the StorEdge Traffic Manager Software (STMS). STMS is commonly known as MPxIO. IBM provides the Subsystem Device Driver (SDD) as a part of the DS8000 at no extra charge. The SDD software and supporting documentation are available from this IBM Web site:

http://www.ibm.com/servers/storage/support/software/sdd

Next, IBM provides the System Storage Multipath Subsystem Device Driver (SDD) s a part of the DS8000 at no extra charge. Finally, IBM DS8000 supports Veritas Volume Manager (VxVM) Dynamic Multipathing (DMP), which is a component of the Veritas Storage Foundation Suite.

Note: SDD does not support clustered Sun servers.

The multipathing technology that you should use depends predominantly on your operating environment and, of course, your business requirements. There are a few limitations depending on your operating system version, your host bus adapters and whether you use clustering. Additional information and details are available in the *IBM System Storage DS8000 Host Systems Attachment Guide*, publication number SC26-7917.

One difference between the multipathing technologies is whether they suppress the redundant paths to the storage. MPxIO and DMP both suppress all paths to the storage except for one, and the device appears to the application as a single-path device. However, SDD allows the original paths to be seen, but creates its own virtual device (called a vpath) for applications to reference.

Solaris Multiplexed I/O (MPxIO)

Multiplexed I/O (MPxIO) is a Sun Solaris multipath driver architecture. This failover driver enables storage arrays to be accessed through multiple host controller interfaces from a single instance of the storage array. MPxIO helps protect against storage array outages because of controller failures. If one controller fails, MPxIO automatically switches to an alternate controller.

MPxIO is fully integrated within the Solaris 10 operating system., and does not need to be installed separately. For the latest Solaris kernel patches and the most recent updates to information about using MPxIO, refer to the Sun document *Sun Solaris Fibre Channel and Storage Multipathing Administration Guide,* document number 819-0139, that is available from:

http://docs.sun.com/app/docs/doc/819-0139

Note: It is recommended that you install the regular kernel jumbo patch, because there are dependencies between the various patches that make up the driver stack.

When MPxIO is enabled, devices in the /dev and /devices trees are named differently from their original names. For example. a device name with MPxIO disabled would be:

```
/dev/dsk/c1t1d0s0
```

Once MPxIO is enabled, the device name would look similar to:

```
/dev/rdsk/c1t201400A0B811804Ad31s0
```

Therefore, you must configure applications that directly access devices to use the new names whenever the MPxIO configuration is enabled or disabled. In addition, the /etc/vfstab file and the dump configuration also contain references to device names. When using the **stmsboot** command to enable or disable MPxIO, `/etc/vfstab` and the `dump` configuration files are automatically updated with the new device names.

Due to a problem found in two-node SunCluster implementations on a storage unit, the following restriction must be observed. When setting up a two-node SunCluster 3.x environment with MPxIO for multipathing, it is mandatory that 1-to-1 zoning be used. That is, each HBA must be in a zone that contains only one server HBA port and only one storage port. This zoning and port masking must be implemented prior to assigning any DS8000 LUNs to the SunCluster host and prior to attempting any SunCluster configuration.

Note: Refer to Sun's Web site at `http://www.sun.com` for up-to-date documentation about this SunCluster implementation restriction.

Subsystem Device Driver (SDD)

SDD provides multipath configuration environment support for a Solaris host system that is attached to DS8000 storage devices. It provides enhanced data availability, dynamic load balancing across multiple paths, and automatic path failover protection. Each SDD vpath device represents a unique physical device on the DS8000 storage system. Each physical device is presented to the Solaris operating system as a disk device which can represent up to thirty two different paths to the same LUN.

By distributing the I/O workload across multiple active paths, SDD provides dynamic load balancing. If a data path should fail, SDD automatically switches the affected I/O operations to another active data path. The SDD failover protection feature minimizes any disruptions in I/O operations by detecting a path failure, notifying the host system of the path failure and then the selection of an alternate data path.

SDD requires a minimum of two independent paths that share the same logical unit to use the load balancing and path failover protection feature up to a maximum of 600 physical LUNs from different DS8000 storage systems to a single Solaris host. SDD supports both 32-bit and 64-bit applications on 64-bit Solaris 10.

For SDD to operate correctly, ensure that the proper Solaris patches are install on your Solaris operating system. Go to the following Web site for the latest information about Solaris patches:

`http://sunsolve.sun.com`

Before installing SDD, make sure that you have root access to your Solaris host system and that all required hardware and software is ready.

The latest SDD package and readme files can be downloaded from the SDD Web site:

`http://www.ibm.com/servers/storage/support/software/sdd`

The installation process is started with the **pkgadd** command with the **-d** option which points to the directory that contains the SDD package. After SDD installation, vpath devices may not be configured and a reboot of the Solaris server is needed. The SDD installation package will determine if a reboot is required and displays a message about this requirement. Vpath devices will be automatically configured after reboot. SDD vpath devices are found in the /dev/rdsk and /dev/dsk directories such that the vpath device is named according to the SDD instance number.

To verify that SDD has been successfully installed, enter the **datapath query device** command. If the command executes, SDD is installed.

Veritas Volume Manager with Dynamic Multipathing (DMP)

The Dynamic Multipathing (DMP) feature of the Veritas Volume Manager (VxVM) provides a path failover mechanism. In the event of the loss of one connection to the disk array, DMP selects the next I/O path for the I/O requests dynamically. this allows the host system to continue to access critical data over the other functioning connections to the disk array.

Veritas VxVM provides Dynamic MultiPathing (DMP) support for disk arrays in the form of Array Support Library (ASL) software packages. ASL packages can be obtained from:

- The VxVM release CD-ROM in the veritas_enabled directory
- The Veritas Technical Support Web site, http://support.veritas.com. Perform a Knowledge Base Search and enter **asl** followed by particular DS4000 storage system in your configuration.

Veritas Volume Manager 4.1, or higher, is supported with Solaris 10. The SMibmasl.pkg package is needed to enable Solaris to recognize the DS4000 machine type. Prior to DMP installation the following prerequisites should be verified:

- HBAs are installed on the Solaris host and parameter settings in the HBA configuration file are modified appropriately.
- HBA bindings are configured.
- Fabric zoning is created and enabled for the Solaris partition.
- Storage is mapped to the Solaris partition.

Once the prerequisites have been verified, the installation process can begin. An overview of the installation process is:

1. Prepare the host for Veritas DMP installation.
2. Install Veritas Volume Manager packages for DMP
3. Install the SMibmasi software package.
4. Configure Veritas Volumen Manager
5. Start the Veritas Enterprise Administrator.

Preparations on the host typically involve manually defining the SAN-attached targets and LUNs in the /kernel/drv/sd.conf file. Each target represents a controller to a disk subsystem and each LUN represents a logical drive. If additional target or LUN definitions are added to the /kernel/drv/sd.conf file, Once any edits are saved, be sure to reboot the Solaris host.

Creating a SAN-attached configuration

DS8000 storage systems are supported in direct-attached Solaris configurations or in a SAN environment through switches in Solaris configurations. In a direct-attached configuration, multiple Solaris servers can be connected to a single DS8000 storage system. Each Solaris server has two HBAs where each HBA is connected to a different DS8000 port.

For SAN-attached configurations, there are several requirements to follow:

- Multiple HBAs within the same server must be isolated from each other if they are connected to the same switch that is connected to the same DS4000 controller port.
- Each server HBA port and storage port must be in its own fabric zone if they are connecting through a single switch.

When connecting Fibre Channel host bus adapters in host servers to DS8000 Storage systems in a Fibre Channel switch environment, IBM recommends that you establish one-to-one zones between the individual server HBA ports and DS8000 host ports. Figure 4-2 displays a simple configuration.

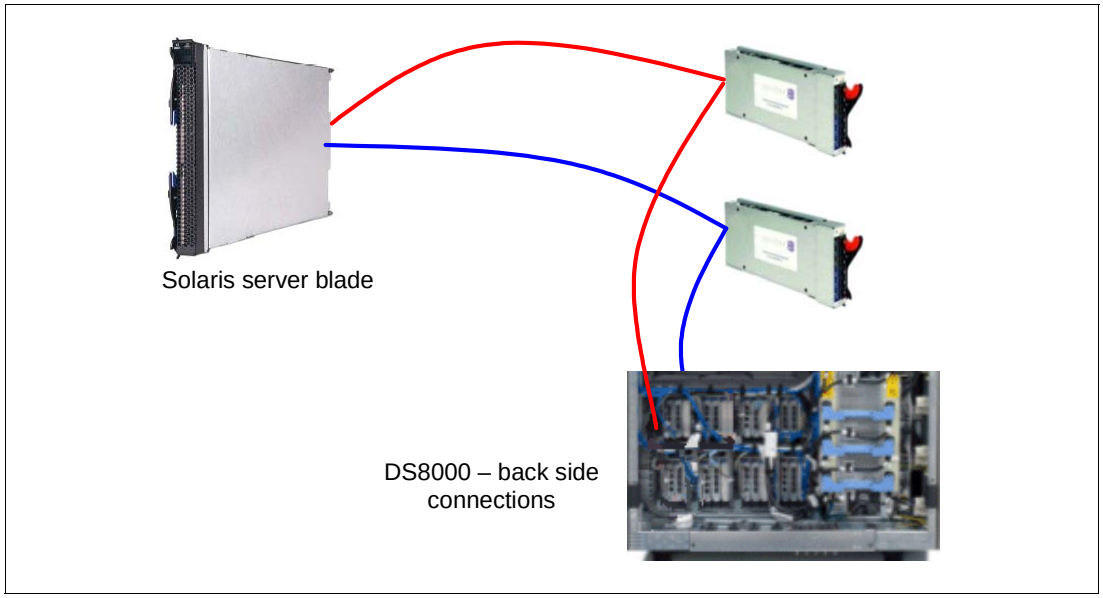


Figure 4-2 DS8000 SAN environment and zoning

Abbreviations and acronyms

ABI	Application Binary Interface
AC	alternating current
AMD	Advanced Micro Devices™
AMM	Advanced Management Module
API	application programming interface
ASL	Array Support Library
ASM	Advanced System Management
ASU	Advanced Settings Utility
BCP	Bulk Copy Program
BIOS	basic input output system
BMC	baseboard management controller
BUI	Browser User Interface
CA	Computer Associates
CD-ROM	compact disc read only memory
CLI	command-line interface
CPU	central processing unit
CRU	customer replaceable unit
DC	domain controller
DHCP	Dynamic Host Configuration Protocol
DIMM	dual inline memory module
DMP	Dynamic Multipathing
DNS	domain name system
ECC	error checking and correcting
ELF	Executable and Linking Format
EOF	end of file
EOL	end of line
ESM	Ethernet switch modules
ETSI	European Telecommunications Standard Industry
FC	Fibre Channel
FMA	Fault Management Architecture
FMD	Fault Management Daemon
GB	gigabyte
GBIC	Gigabit Interface Converter
GUI	graphical user interface
HBA	host bus adapter
HDD	hard disk drive
HMC	Hardware Management Console
HPC	high performance computing
HT	Hyper-Threading HyperTransport
HTML	Hypertext Markup Language
HW	hardware
I/O	input/output
IBM	International Business Machines Corporation
IBM	International Business Machines Corporation
ID	identifier
IDE	integrated drive electronics
IP	Internet Protocol
IPTV	Internet Protocol Television
ISO	International Organization for Standardization
IT	information technology
ITSO	International Technical Support Organization
KVM	keyboard video mouse
LAN	local area network
LDAP	Lightweight Directory Access Protocol
LED	light emitting diode
LUN	logical unit number
MB	megabyte
MIB	management information base
MIO	Memory and I/O
MM	Management Module
MOM	Microsoft Operations Manager
NEBS	network equipment building system
NGN	next-generation networks
NIS	Network Information Service
NMI	non-maskable interrupt
OID	object identifiers
OS	operating system
PC	personal computer
POSIX	Portable Operating System Interface
PXE	Pre-boot-execution
RAID	redundant array of independent disks
RDAC	Redundant Disk Array Controller

RDM	Remote Deployment Manager
RSA	Remote Supervisor Adapter
SAN	storage area network
SAS	Serial Attached SCSI
SATA	Serial ATA
SDD	Subsystem Device Driver
SFF	Small Form Factor
SIO	Storage and I/O
SMF	Service Management Facility
SMS	System Managed Space
SNMP	Simple Network Management Protocol
SOL	Serial over LAN
SSH	Secure Shell
STMS	StorEdge Traffic Manager Software
SVID	System V Interface Definition
TCP/IP	Transmission Control Protocol/Internet Protocol
TOE	TCP offload engine
URL	Uniform Resource Locator
USB	universal serial bus
VOIP	Voice over Internet Protocol
VSM	Volume Systems Monitoring
WWPN	World-Wide Port Name

Related publications

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

IBM Redbooks

You can search for, view, or download books, papers, Technotes, draft publications and additional materials, as well as order hardcopy Redbooks, at the IBM Redbooks Web site:

ibm.com/redbooks

Related publications from IBM Redbooks include the following:

- *Implementing IBM Director 5.20*, SG24-6188
- *Deployment using Altiris on IBM System x and BladeCenter Servers*, SG24-7303

Other publications

These publications are also relevant as further information sources:

IBM publications

- Installation Instructions for Sun Solaris 10 - IBM BladeCenter HS40 (Type 8839)
  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-5070449
- Installing Sun Solaris 10 Operating System - IBM BladeCenter HS20 (Type 8843)
  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-62921
- Installing Sun Solaris 10 Operating System 1/06 - IBM BladeCenter LS20 (Type 8850)
  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-63018
- Installing Sun Solaris 10 Operating System 1/06 - IBM BladeCenter LS20 (Type 8850)
  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-64047
- *IBM BladeCenter Serial Over LAN Setup Guide*
  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-54666
- *IBM BladeCenter Management Module: Command-Line Interface Reference Guide*
  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-5069717
- *IBM ServerGuide Scripting Toolkit; Hardware Configuration and Installation of the Solaris 10 OS on IBM Blade Servers*
  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-66510
- *Configuring IBM TotalStorage FAStT for use with Veritas Dynamic Multi-Pathing DMP*
  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-57485
- *IBM System storage DS8000 Host Systems Attachment Guide*
  http://www.ibm.com/support/docview.wss?uid=ssg1S7001161

- *IBM System Storage DS8000 Introduction and Planning Guide*
  http://www.ibm.com/support/docview.wss?uid=ssg1S7001073
- IBM System Storage DS4000 Interoperability Matrix
  http://www.ibm.com/servers/storage/disk/ds4000/pdf/interop-matrix.pdf
- IBM System Storage DS8000 Interoperability Matrix
  http://www.ibm.com/systems/storage/disk/ds8000/interop.pdf

Sun Solaris publications

- *Solaris 10 11/06 Installation Guide: Network-Based Installations*
  http://docs.sun.com/app/docs/doc/819-6395
- *Solaris 10 1/06 Installation Guide for IBM BladeCenter Servers*
  http://docs.sun.com/app/docs/doc/819-5455
- *Solaris DHCP Administration Guide*
  http://docs.sun.com/app/docs/doc/806-5529
- *Solaris System Management Agent Administration Guide*
  http://docs.sun.com/app/docs/doc/817-3000
- The Sun Management Center Document Collection:
  http://docs.sun.com/app/docs/coll/810.8
- *Solaris Fibre Channel Storage Configuration and Multipathing Administration Guide*
  http://docs.sun.com/app/docs/doc/819-0139

Online resources

These Web sites are also relevant as further information sources:

IBM Web pages

- Support for System Storage Multipath Subsystem Device Driver
  http://www.ibm.com/servers/storage/support/software/sdd
- IBM DS4000 family overview
  http://www.ibm.com/servers/storage/disk/ds4000
- IBM DS8000 family overview
  http://www.ibm.com/systems/storage/disk/ds8000
- Host bus adapter (HBA) drivers and firmware
  http://www.ibm.com/servers/storage/support/config/hba
- Support for Disk systems
  http://www.ibm.com/servers/storage/support/disk
- Advanced Settings Utility
  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-55021
- IBM ServeRAID Application CD
  http://www.ibm.com/support/docview.wss?uid=psg1MIGR-61707

- IBM Director Documentation and resources

  http://www.ibm.com/systems/management/director/resources/

Sun Web pages

- Adding Device Drivers to the New Boot Install Image in the Solaris 10 OS for x86 Platforms

  http://www.sun.com/bigadmin/features/articles/device_driver_install.html
- Solaris ABI Tools downloads

  http://www.sun.com/software/solaris/programs/abi/get.xml
- Solaris ABI Tools - SolCAT

  http://www.sun.com/software/solaris/programs/abi/solcat.xml
- Sun's Software Portfolio

  http://www.sun.com/software/swportfolio/get.jsp
- DTrace

  http://sun.com/bigadmin/content/dtrace/
- Sun Connection

  http://sun.com/service/sunconnection
- SunSolve Online

  http://sunsolve.sun.com

Other Web pages

- OpenSolaris Community: DTrace

  http://opensolaris.org/os/community/dtrace/
- Symantec Enterprise support

  http://support.veritas.com
- Broadcom Ethernet NIC NetXtreme II Driver Downloads

  http://www.broadcom.com/support/ethernet_nic/netxtremeii.php
- Broadcom Ethernet NIC NetXtreme Server Driver Downloads

  http://www.broadcom.com/support/ethernet_nic/netxtreme_server.php

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services