



QLogic HCA and InfiniPath[®] Software Install Guide

Version 2.2

IB0056101-00 G

Information furnished in this manual is believed to be accurate and reliable. However, QLogic Corporation assumes no responsibility for its use, nor for any infringements of patents or other rights of third parties which may result from its use. QLogic Corporation reserves the right to change product specifications at any time without notice. Applications described in this document for any of these products are for illustrative purposes only. QLogic Corporation makes no representation nor warranty that such applications are suitable for the specified use without further testing or modification. QLogic Corporation assumes no responsibility for any errors that may appear in this document.

No part of this document may be copied nor reproduced by any means, nor translated nor transmitted to any magnetic medium without the express written consent of QLogic Corporation. In accordance with the terms of their valid QLogic agreements, customers are permitted to make electronic and paper copies of this document for their own exclusive use.

The QHT7040, QHT7140, QLE7140, QLE7240, and QLE7280 QLogic Host Channel Adapters are covered by the following patent: 7308535.

Document Revision History	
Rev. 1.0, 8/20/2005	
Rev. 1.1, 11/15/05	
Rev. 1.2, 02/15/06	
Rev. 1.3, Beta 1, 4/15/06	
Rev 1.3, 6/15/06	
Rev. 2.0 Beta, 8/15/06, QLogic Version Number IB0056101-00 A	
Rev. 2.0 Beta 2 10/15/06, QLogic Version Number IB0056101-00 B	
Rev. 2.0 11/30/06, QLogic Version Number IB0056101-00 C	
Rev. 2.0 3/23/07, QLogic Version Number IB0056101-00 D	
Rev. 2.1 8/24/07, QLogic Version Number IB0056101-00 E	
Rev. 2.2 5/27/08, QLogic Version Number IB0056101-00 F	
Rev. 2.2 9/5/08, QLogic Version Number IB0056101-00 G	
Changes	Sections Affected
Re-format to new template (no change to content)	All

Table of Contents

1	Introduction	
	Who Should Read this Guide	1-1
	How this Guide is Organized	1-1
	Overview	1-2
	Interoperability	1-3
	Conventions Used in this Guide	1-3
	Documentation	1-4
	Contact Information	1-5
2	Feature Overview	
	What's New in this Release	2-1
	New Features	2-2
	Supported Distributions and Kernels	2-3
	Compiler Support	2-4
	Software Components	2-4
3	Step-by-Step Installation Checklist	
	Hardware Installation	3-1
	Software Installation	3-2
4	Hardware Installation	
	Hardware Installation Requirements	4-1
	Hardware	4-1
	Form Factors	4-2
	Cabling and Switches	4-3
	Optical Fibre Option	4-4
	Configuring the BIOS.	4-4
	Safety with Electricity	4-5
	Unpacking Information	4-5
	Verify the Package Contents	4-5
	List of the Package Contents.	4-5
	Unpacking the QLogic Adapter	4-8
	Hardware Installation	4-9

Hardware Installation for QLE7240, QLE7280, or QLE7140 with PCI Express Riser	4-9
Dual Adapter Installation.	4-9
Installation Steps.	4-9
Hardware Installation for QHT7140 with HTX Riser	4-12
Hardware Installation for QLE7240, QLE7280, and QLE7140 Without a PCI Express Riser	4-15
Hardware Installation for the QHT7140 Without an HTX Riser	4-16
Switch Configuration and Monitoring	4-17
Cabling the Adapter to the InfiniBand Switch.	4-17
Completing the Installation	4-18

5 Software Installation

Cluster Setup	5-1
Types of Nodes in a Cluster Environment.	5-1
Supported Linux Distributions	5-2
Setting Up Your Environment	5-2
Downloading and Unpacking the InfiniPath and OpenFabrics Software	5-3
Choosing the Appropriate Download Files	5-3
Unpacking the InfiniPath tar File	5-4
Check for Missing Files or RPMs	5-5
Installing the InfiniPath and OpenFabrics RPMs	5-6
Choosing the RPMs to Install	5-7
Using rpm to Install InfiniPath and OpenFabrics.	5-8
Installed Layout	5-9
Starting the InfiniPath Service	5-10
InfiniPath and OpenFabrics Driver Overview	5-11
Configuring the InfiniPath Drivers.	5-11
Configuring the <code>ib_ipath</code> Driver	5-12
Configuring the <code>ipath_ether</code> Network Interface	5-12
<code>ipath_ether</code> Configuration on Red Hat	5-12
<code>ipath_ether</code> Configuration on SLES	5-14
OpenFabrics Drivers and Services Configuration and Startup.	5-16
Configuring the IPoIB Network Interface	5-17
Configuring and Administering the VNIC Interface	5-18
OpenSM	5-23
SRP	5-24
Using OFED SRP	5-24
MPI over uDAPL	5-26
Other Configuration: Changing the MTU Size	5-26

Starting and Stopping the InfiniPath Software	5-26
Rebuilding or Reinstalling Drivers After a Kernel Upgrade	5-28
Rebuilding or Reinstalling Drivers if a Different Kernel is Installed.....	5-29
Further Information on Configuring and Loading Drivers.....	5-29
LED Link and Data Indicators.....	5-29
Adapter Settings.....	5-30
Customer Acceptance Utility	5-31
Removing Software Packages	5-33
Uninstalling InfiniPath and OpenFabrics RPMs.....	5-33
Uninstalling OFED 1.3 Software	5-33
Downgrading RPMs.....	5-33
Additional Installation Instructions	5-33
Installing Lustre	5-34
Installing QLogic MPI in an Alternate Location	5-34
Installing on an Unsupported Distribution	5-35
Managing and Installing Software Using Rocks	5-36
Installing Rocks and InfiniPath RPMs.....	5-37
Further Information on Rocks and kickstart	5-41

A Installation Troubleshooting

Hardware Issues	A-1
Node Spontaneously Reboots.....	A-1
Some HTX Motherboards May Need Two or More CPUs in Use	A-1
BIOS Settings.....	A-1
MTRR Mapping and Write Combining.....	A-2
Edit BIOS Settings to Fix MTRR Issues.....	A-2
Use the <code>ipath_mtrr</code> Script to Fix MTRR Issues	A-3
Issue with Supermicro® H8DCE-HTe and QHT7040	A-3
Software Installation Issues	A-3
Version Number Conflict with <code>opensm-*</code> on RHEL5 Systems.....	A-4
OpenFabrics Dependencies	A-4
OpenFabrics Library Dependencies	A-4
Missing Kernel RPM Errors.....	A-5
Resolving Conflicts	A-6
<code>mpirun</code> Installation Requires 32-bit Support	A-6
<code>ifup</code> on <code>ipath_ether</code> on SLES 10 Reports "unknown device".....	A-7
Lockable Memory Error on Initial Installation of InfiniPath.....	A-7

B **Configuration Files**

C **RPM Descriptions**

InfiniPath and OpenFabrics RPMs	C-1
Different Nodes May Use Different RPMs	C-1
RPM Version Numbers and Identifiers	C-1
RPM Organization	C-2
Documentation and InfiniPath RPMs	C-2
OpenFabrics RPMs	C-4

List of Figures

Figure		Page
4-1	QLogic QLE7280 with IBA7220 ASIC	4-7
4-2	QLogic QLE7140 Card with Riser, Top View	4-7
4-3	QLogic QHT7040/QHT7140 Full and Low Profile Cards with Riser, Top View	4-8
4-4	PCIe Slot in a Typical Motherboard	4-10
4-5	QLogic PCIe HCA Assembly with Riser Card	4-11
4-6	Assembled PCIe HCA with Riser	4-12
4-7	HTX Slot.	4-13
4-8	QLogic QHT7140 Adapter with Riser Card	4-14
4-9	Assembled QHT7140 with Riser	4-15
4-10	QHT7140 Without Riser Installed in a 3U Chassis	4-17
5-1	Relationship Between InfiniPath and OpenFabrics Software.	5-11

List of Tables

Table		Page
1-1	Typographical Conventions	1-3
2-1	QLogic Adapter Model Numbers	2-1
2-2	InfiniPath/OpenFabrics Supported Distributions and Kernels	2-3
4-1	Adapter Models and Related Platforms	4-1
4-2	QLogic InfiniBand Cables.	4-3
5-1	InfiniPath/OpenFabrics Supported Distributions and Kernels	5-2
5-2	Available Packages	5-3
5-3	InfiniPath and OpenFabrics RPMs to Use for Each Node in a Cluster	5-6
5-4	RPMS to Install	5-7
5-5	LED Link and Data Indicators	5-29
5-6	<code>ipath_checkout</code> Options	5-32
B-1	Configuration Files	B-1
C-1	InfiniPath Documentation/RPMs	C-2
C-2	InfiniPath/RPMs	C-3
C-3	InfiniPath-Devel/RPMs	C-3
C-4	InfiniPath-MPI/RPMs	C-4
C-5	OpenFabrics Documentation/RPMs	C-4
C-6	OpenFabrics/RPMs	C-4
C-7	OpenFabrics-Devel/RPMs	C-7
C-8	OpenSM/RPMs	C-8
C-9	Other HCAs/RPMs	C-8
C-10	Other HCAs-Devel/RPMs.	C-9
C-11	OtherMPIs/RPMs	C-9

Notes

1 Introduction

This chapter describes the contents, intended audience, and organization of the QLogic HCA and InfiniPath Software Install Guide.

The *QLogic HCA and InfiniPath Software Install Guide* contains instructions for installing the QLogic Host Channel Adapters (HCAs) and the InfiniPath and OpenFabrics software. The following adapters are covered in this guide:

- QLE7140 PCI Express® (PCIe)
- QLE7240 PCI Express
- QLE7280 PCI Express
- QHT7040/QHT7140 HyperTransport Expansion (HTX™)

Who Should Read this Guide

This installation guide is intended for cluster administrators responsible for installing the QLogic QLE7140, QLE7240, QLE7280 or QHT7040/QHT7140 adapter and InfiniPath software on their Linux® cluster. Additional detailed installation information and instructions for administering the QLogic cluster can be found in the *QLogic HCA and InfiniPath Software User Guide*.

The *QLogic HCA and InfiniPath Software Install Guide* assumes that you are familiar with both cluster networking and the specific hardware that you plan to use. Before installing the HCA, you should have basic knowledge of your host and target operating systems, and working knowledge of message passing concepts.

This document does not contain all the information you need to use basic Linux commands or to perform all system administration tasks. For this information, see the software documentation you received with your system.

How this Guide is Organized

The *QLogic HCA and InfiniPath Software Install Guide* is organized into these sections:

- [Section 1](#), Introduction, contains an overview of the HCAs and software, describes interoperability with other products, lists all related documentation, and provides QLogic contact information.
- [Section 2](#), Feature Overview, contains features for this release, the supported QLogic adapter models, supported distributions and kernels, and a list of the software components.

- [Section 3](#), Step-by-Step Installation Checklist, provides a high-level overview of the hardware and software installation procedures.
- [Section 4](#), Hardware Installation, includes instructions for installing the QLogic QLE7140, QLE7240, QLE7280, QHT7040, and QHT7140 HCAs.
- [Section 5](#), Software Installation, includes instructions for installing the QLogic InfiniPath and OpenFabrics software.
- [Appendix A](#), Installation Troubleshooting, contains information about issues that may occur during installation.
- [Appendix B](#), Configuration Files, contains descriptions of the configuration and configuration template files used by the InfiniPath and OpenFabrics software.
- [Appendix C](#), RPM Descriptions
- [Index](#), lists major subjects and concepts with page numbers for easy reference.

Overview

The material in this documentation pertains to an InfiniPath *cluster*. A cluster is defined as a collection of nodes, each attached to an InfiniBand™-based fabric through the QLogic interconnect. The nodes are Linux-based computers, each having up to 16 processors.

The QLogic HCAs are InfiniBand 4X. The Double Data Rate (DDR) QLE7240 and QLE7280 adapters have a raw data rate of 20Gbps (data rate of 16Gbps). For the Single Data Rate (SDR) adapters, the QLE7140 and QHT7140, the raw data rate is 10Gbps (data rate of 8Gbps). The QLE7240 and QLE7280 can also run in SDR mode.

The QLogic adapters utilize standard, off-the-shelf InfiniBand 4X switches and cabling. The QLogic interconnect is designed to work with all InfiniBand-compliant switches.

NOTE:

If you are using the QLE7240 or QLE7280, and want to use DDR mode, then DDR-capable switches must be used.

InfiniPath OpenFabrics software is interoperable with other vendors' InfiniBand Host Channel Adapters (HCAs) running compatible OpenFabrics releases. There are several options for subnet management in your cluster:

- Use the embedded Subnet Manager (SM) in one or more managed switches supplied by your InfiniBand switch vendor.
- Use the Open source Subnet Manager (OpenSM) component of OpenFabrics.
- Use a host-based Subnet Manager.

Interoperability

QLogic InfiniPath participates in the standard InfiniBand subnet management protocols for configuration and monitoring. Note that:

- InfiniPath OpenFabrics (including Internet Protocol over InfiniBand (IPoIB)) is interoperable with other vendors' InfiniBand HCAs running compatible OpenFabrics releases.
- The QLogic MPI and Ethernet emulation stacks (`ipath_ether`) are not interoperable with other InfiniBand HCAs and Target Channel Adapters (TCAs). Instead, InfiniPath uses an InfiniBand-compliant, vendor-specific protocol that is highly optimized for MPI and Transmission Control Protocol (TCP) between InfiniPath-equipped hosts.

NOTE:

See the OpenFabrics web site at www.openfabrics.org for more information on the OpenFabrics Alliance.

Conventions Used in this Guide

This guide uses the typographical conventions listed in [Table 1-1](#).

Table 1-1. Typographical Conventions

Convention	Meaning
<code>command</code>	Fixed-space font is used for literal items such as commands, functions, programs, files and pathnames, and program output.
<code>variable</code>	Italic fixed-space font is used for variable names in programs and command lines.
<i>concept</i>	Italic font is used for emphasis and concepts, as well as for documentation names/titles.

Table 1-1. Typographical Conventions (Continued)

Convention	Meaning
user input	Bold fixed-space font is used for literal items in commands or constructs that you type.
\$	Indicates a command line prompt.
#	Indicates a command line prompt as root when using <code>bash</code> or <code>sh</code> .
[]	Brackets enclose optional elements of a command or program construct.
...	Ellipses indicate that a preceding element can be repeated.
>	A right caret identifies the cascading path of menu commands used in a procedure.
2.2	The current version number of the software is included in the RPM names and within this documentation.
NOTE:	Indicates important information.

Documentation

The product documentation includes:

- *The QLogic HCA and InfiniPath Software Install Guide*
- *The QLogic HCA and InfiniPath Software User Guide*
- *The QLogic FastFabric Users Guide* (for information on QLogic InfiniServ Tools)
- *The OFED+ Users Guide* (for information on QLogic VNIC and QLogic SRP)
- Release Notes
- Quick Start Guide
- Readme file

For more information on system administration, using the QLogic Message-Passing Interface (MPI), and troubleshooting adapter hardware and software, see the *QLogic HCA and InfiniPath Software User Guide*.

Contact Information

Support Headquarters	QLogic Corporation 4601 Dean Lakes Blvd Shakopee, MN 55379 USA
QLogic Web Site	www.qlogic.com
Technical Support Web Site	support.qlogic.com
Technical Support Email	support@qlogic.com
Technical Training Email	tech.training@qlogic.com
North American Region	
Email	support@qlogic.com
Phone	+1-952-932-4040
Fax	+1 952-974-4910
All other regions of the world	
QLogic Web Site	www.qlogic.com

Notes

2 Feature Overview

This section contains the features for this release, the supported QLogic adapter models, supported distributions and kernels, and a list of the software components.

What's New in this Release

This release adds support for the QLE7240 and QLE7280 InfiniBand DDR Host Channel Adapters (HCAs), which offer twice the link bandwidth of SDR HCAs. The extra bandwidth improves performance for both latency-sensitive and bandwidth-intensive applications.

This version of the InfiniPath software provides support for all of the QLogic HCAs in [Table 2-1](#).

Table 2-1. QLogic Adapter Model Numbers

QLogic Model Number	Description
QHT7040	Single port 10Gbps SDR 4X InfiniBand to HTX adapter. For systems with HTX expansion slots.
QHT7140 ^a	Single port 10Gbps SDR 4X InfiniBand to HTX adapter. For systems with HTX expansion slots.
QLE7140	Single port 10Gbps SDR 4X InfiniBand to PCI Express x8 adapter. Supported on systems with PCI Express (PCIe) x8 or x16 slots.
QLE7240	Single port 20Gbps DDR 4X InfiniBand to PCI Express x8 adapter. Supported on systems with PCI Express x8 or x16 slots.

Table 2-1. QLogic Adapter Model Numbers (Continued)

QLogic Model Number	Description
QLE7280	Single port 20Gbps DDR 4X InfiniBand to PCI Express x16 adapter. Supported on systems with PCI Express x16 slots. The QLE7280 is backward compatible; it can also be used with PCIe adapters that connect to x8 slots.

Table Notes

PCIe is Gen 1

^a The QHT7140 has a smaller form factor than the QHT7040, but is otherwise the same. Throughout this document, the QHT7040 and QHT7140 will be collectively referred to as the *QHT7140* unless otherwise noted.

New Features

The following features are new to the 2.2 release:

- Expanded MPI scalability enhancements for PCI Express have been added. On the QLE7240 and QLE7280, up to 16 dedicated hardware contexts per node are available. The QHT7140 has eight per node. The QLE7140 has four per node.
- A total of 64 processes on the QLE7240 and QLE7280 are supported when context sharing is enabled. The QHT7040 and QHT7140 support a total of 32 processes per adapter. The QLE7140 supports a total of 16 processes per adapter.
- This release continues support for multiple high-performance native PSM Message Passing Interface (MPI) implementations added in the 2.1 release. (PSM is QLogic's accelerated library for high performance MPIs). In addition to QLogic MPI, the currently supported MPI implementations are HP-MPI, Open MPI, MVAPICH, and Scali. Open MPI provides MPI-2 functionality, including one-sided operations and dynamic processes. These all offer the same high performance as QLogic MPI.
- Dual PCIe QLogic adapters per node are supported.
- Driver support for the QLogic Virtual Network Interface Controller (VNIC) is provided in this release. The VNIC Upper Layer Protocol (ULP) works in concert with firmware running on Virtual Input/Output (VIO) hardware such as the SilverStorm™ Ethernet Virtual I/O Controller (EVIC), providing virtual Ethernet connectivity for Linux operating systems.
- The QLogic InfiniBand Fabric Suite CD is available separately for purchase. The CD includes FastFabric, the QLogic Subnet Manager (SM), and the Fabric Viewer GUI for the QLogic SM.

- A subset of the QLogic InfiniBand Fabric Suite, the enablement tools, are offered with this release.
- Two separate SCSI RDMA Protocol (SRP) modules are provided: the standard OpenFabrics (OFED) SRP, and QLogic SRP.
- QLogic MPI supports running exclusively on a single node without the installation of the HCA hardware.
- OpenMPI and MVAPICH libraries built with the GNU, PGI, PathScale™, and Intel® compilers are available, with corresponding `mpitests` RPMs. You can use `mpi-selector` to choose which MPI you want. These all run over PSM.
- 4K Maximum Transfer Unit (MTU) is supported and is on by default. To take advantage of 4KB MTU, use a switch that supports 4KB MTU. QLogic also supports 2KB switches, and 4KB MTU switches configured for 2KB MTU. QLogic switches with firmware version 4.1 or later are recommended.
- The Lustre® cluster filesystem is supported.

Additional up-to-date information can be found on the QLogic web site, specifically:

- The high performance computing page at www.qlogic.com/Products/HPC_products_landingpage.aspx
- The InfiniBand HCA page at www.qlogic.com/Products/HPC_products_infipathhcas.aspx

Supported Distributions and Kernels

The QLogic interconnect runs on AMD™ Opteron™ and Intel EM64T systems running Linux®. The currently supported distributions and associated Linux kernel versions for InfiniPath and OpenFabrics are listed in [Table 2-2](#). The kernels are the ones that shipped with the distributions.

Table 2-2. InfiniPath/OpenFabrics Supported Distributions and Kernels

Distribution	InfiniPath/OpenFabrics Supported Kernels
Fedora Core 6 (FC6)	2.6.22 (x86_64)
Red Hat® Enterprise Linux® 4.4, 4.5, 4.6 (RHEL4.4, 4.5, 4.6)	2.6.9-42 (U4), 2.6.9-55 (U5) 2.6.9-67 (U6) (x86_64)
CentOS 4.4, 4.5 (Rocks 4.4, 4.5, 4.6)	2.6.9-42, 2.6.9.55, 2.6.9-67 (x86_64)
Scientific Linux 4.4, 4.5, 4.6	2.6.9-42, 2.6.9.55, 2.6.9-67 (x86_64)

Table 2-2. InfiniPath/OpenFabrics Supported Distributions and Kernels (Continued)

Distribution	InfiniPath/OpenFabrics Supported Kernels
Red Hat Enterprise Linux 5.0 (RHEL 5.0), RHEL 5.1	2.6.18-8, 2.6.18-53 (x86_64)
CentOS 5.0, 5.1 (Rocks 5.0, 5.1)	2.6.18, 2.6.18-53 (x86_64)
Scientific Linux 5.0, 5.1	2.6.18, 2.6.18-53 (x86_64)
SUSE® Linux Enterprise Server (SLES 10 GM, SP 1)	2.6.16.21, 2.6.16.46 (x86_64)

NOTE:

Fedora Core 4 and Fedora Core 5 are not supported in the InfiniPath 2.2 release.

Compiler Support

QLogic MPI supports a number of compilers. These include:

- PathScale Compiler Suite 3.0 and 3.1
- PGI 5.2 , 6.0, and 7.1
- Intel 9.x and 10.1
- GNU gcc 3.3.x, 3.4.x, 4.0 and 4.1 compiler suites
- gfortran

The PathScale Compiler Suite Version 3.x is now supported on systems that have the GNU 4.0 and 4.1 compilers and compiler environment (header files and libraries).

Please check the QLogic web site for updated information on supported compilers.

Software Components

The software includes the InfiniPath HCA driver, QLogic MPI, standard networking over Ethernet emulation, InfiniPath Subnet Management Agent and associated utilities, and OFED for InfiniPath.

This release includes a full set of OFED 1.3 usermode RPMs, with some enhancements, including a new version of the VNIC tools and driver, and support for the QLE7240 and QLE7280 adapters.

Included components are:

- InfiniPath driver
- InfiniPath Ethernet emulation (`ipath_ether`)
- InfiniPath libraries
- InfiniPath utilities, configuration, and support tools, including `ipath_checkout`, `ipath_control`, `ipath_pkt_test`, and `ipathstats`
- QLogic MPI
- QLogic MPI benchmarks and utilities
- OpenMPI and MVAPICH libraries built with the GNU, PGI, PathScale, and Intel compilers, with corresponding `mpitests` RPMs and `mpi-selector`
- OpenFabrics protocols, including Subnet Management Agent
- OpenFabrics libraries and utilities
- QLogic VNIC module
- Enablement tools

This release provides support for the following protocols and transport services:

- IPoIB (TCP/IP networking in either Connected or Datagram mode)
- Sockets Direct Protocol (SDP)
- Open source Subnet Manager (OpenSM)
- Unreliable Datagram (UD)
- Reliable Connection (RC)
- Unreliable Connection (UC)
- Shared Receive Queue (SRQ)
- Reliable Datagram Sockets (RDS)
- iSCSI Extensions for RDMA (iSER)

This release supports two versions of SCSI RDMA Protocol (SRP):

- OFED SRP
- QLogic SRP

No support is provided for Reliable Datagram (RD).

NOTE:

OpenFabrics programs (32-bit) using the verbs interfaces are not supported in this InfiniPath release, but may be supported in a future release.

More details about the hardware and software can be found in [Section 4](#) and [Section 5](#).

3 Step-by-Step Installation Checklist

This section provides an overview of the hardware and software installation procedures. Detailed steps are found in [Section 4 “Hardware Installation”](#) and [Section 5 “Software Installation”](#).

Hardware Installation

The following steps summarize the basic hardware installation procedure:

1. Check that the adapter hardware is appropriate for your platform. See [Table 4-1](#).
2. Check to see that you have the appropriate cables and switches, as described in [“Cabling and Switches” on page 4-3](#).
3. Check to see that you are running a supported Linux distribution/kernel. See [Table 2-2](#) or [Table 5-1](#).
4. Verify that the BIOS for your system is configured for use with the QLogic adapter. See [“Configuring the BIOS” on page 4-4](#).
5. Following the safety instructions in [“Safety with Electricity” on page 4-5](#). Unpack the adapter ([“Unpacking Information” on page 4-5](#)) and verify the package contents.
6. Install the adapter by following the instructions in [“Hardware Installation” on page 4-9](#).
7. Cable the adapter to the switch, as described in [“Cabling the Adapter to the InfiniBand Switch” on page 4-17](#). Check that all InfiniBand switches are configured.
8. Follow the steps in [“Completing the Installation” on page 4-18](#) to finish the installation.

Software Installation

The following steps summarize the basic InfiniPath and OpenFabrics software installation and startup. These steps must be performed on each node in the cluster:

1. Make sure that the HCA hardware installation has been completed according to the instructions in [“Hardware Installation” on page 4-1](#).
2. Verify that the Linux kernel software is installed on each node in the cluster. The required kernels and supported Linux distributions for both InfiniPath and OpenFabrics are defined in [Table 5-1](#).
3. Make sure that your environment has been set up as described in [“Setting Up Your Environment” on page 5-2](#).
4. Download your version of the InfiniPath/OpenFabrics software from the QLogic web site to a local server directory. See [“Choosing the Appropriate Download Files” on page 5-3](#).
5. Unpack the `tar` file and check for any missing files or RPMs. See [“Unpacking the InfiniPath tar File” on page 5-4](#) and [“Check for Missing Files or RPMs” on page 5-5](#).
6. Install the appropriate packages on each cluster node as described in [“Installing the InfiniPath and OpenFabrics RPMs” on page 5-6](#). Also see [Table 5-4](#) for a list of RPMs to install.

NOTE:

Rocks may be used as a cluster install method. See [“Managing and Installing Software Using Rocks” on page 5-36](#).

7. Under the following circumstances, the system needs to be rebooted:
 - If this is the first InfiniPath installation OR
 - If you have installed VNIC with the OpenFabrics RPM set

The system can be rebooted after all the software has been installed. If you are only upgrading from a prior InfiniPath software installation, drivers can be restarted manually. See [“Starting the InfiniPath Service” on page 5-10](#).

8. If you want to use the optional InfiniPath (`ipath_ether`) and OpenFabrics drivers (`ipoib`) and services (`opensm`, `srp`), configure them as described in [“Configuring the InfiniPath Drivers” on page 5-11](#) and [“OpenFabrics Drivers and Services Configuration and Startup” on page 5-16](#).
9. Check the system state by observing the LEDs. See [“LED Link and Data Indicators” on page 5-29](#).

10. Optimize your adapter for the best performance. See [“Adapter Settings” on page 5-30](#). Also see the Performance and Management Tips section in the *QLogic HCA and InfiniPath Software User Guide*.
11. Perform the recommended health checks. See [“Customer Acceptance Utility” on page 5-31](#).
12. After installing the InfiniPath and OpenFabrics software, refer to the *QLogic HCA and InfiniPath Software User Guide* for more information about using InfiniPath, QLogic MPI, and OpenFabrics products.

Notes

4 Hardware Installation

This section lists the requirements and provides instructions for installing the QLogic InfiniPath Interconnect adapters. Instructions are included for the QLogic DDR PCI Express adapters, the QLE7240 and QLE7280; the QLogic InfiniPath PCIe adapter and PCIe riser card, QLE7140; and the QHT7040 or QHT7140 adapter hardware and HTX riser card. These components are collectively referred to as the *adapter* and the *riser card* in the remainder of this document.

The adapter is a low-latency, high-bandwidth, high message rate cluster interconnect for InfiniBand. The QLogic interconnect is InfiniBand 4X, with a raw data rate of 20Gbps (data rate of 16Gbps) for the QLE7240 and QLE7280; and 10Gbps (data rate of 8Gbps) for the QLE7140, QHT7040, and QHT7140.

OpenFabrics is interoperable with other vendors' InfiniBand Host Channel Adapters (HCAs) running compatible OpenFabrics releases.

Hardware Installation Requirements

This section lists hardware and software environment requirements for installing the QLogic QLE7240, QLE7280, QLE7140, QHT7040, or QHT7140.

Hardware

QLogic interconnect adapters are for use with UL listed computers. The following statement is true for all the adapters:

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operations.

Different adapter cards work on different platforms. [Table 4-1](#) shows the relationship between the adapter model and different types of motherboards.

Table 4-1. Adapter Models and Related Platforms

QLogic Model Number	Platform	Plugs Into
QLE7240	PCI Express systems	Standard PCI Express x8 or x16 slot
QLE7280	PCI Express systems	Standard PCI Express x16 slot

Table 4-1. Adapter Models and Related Platforms (Continued)

QLogic Model Number	Platform	Plugs Into
QLE7140	PCI Express systems	Standard PCI Express x8 or x16 slot
QHT7040	Motherboards with HTX connectors	HyperTransport HTX slot
QHT7140	Motherboards with HTX connectors	HyperTransport HTX slot

Installation of the QLE7240, QLE7280, QLE7140, QHT7040, or QHT7140 in a 1U or 2U chassis requires the use of a riser card. See [Figure 4-4](#) for an illustration of a PCI Express (PCIe) slot in a typical motherboard. See [Figure 4-7](#) for an illustration of an HTX slot for a typical Opteron motherboard.

The motherboard vendor is the optimal source for information on the layout and use of HyperTransport and PCI Express-enabled expansion slots on supported motherboards.

Form Factors

The **QLE7240**, **QLE7280**, and **QLE7140** are the model numbers for the adapters that ship in the standard PCI Express half-height, short-form factor. These adapters can be used with either full-height or low-profile face plates.

The **QHT7040** is the model number for the adapter that shipped in the HTX full-height factor. The HTX low-profile form factor is referred to as the **QHT7140**. It is the same as the QHT7040, except for its more compact size. In either case, the adapter is backward and forward compatible for the motherboards in which it is supported. The QHT7040 and QHT7140 HTX adapters are collectively referred to as the *QHT7140* unless otherwise stated.

When the QHT7040 or QHT7140 adapter is installed with the riser card, it may prevent some or all of the other PCI expansion slots from being used, depending on the form factor of the adapter and motherboard.

Run `ipath_control -i` to see information on which form adapter is installed. The file `/sys/bus/pci/drivers/ib_ipath/00/boardversion` contains the same information. For more information, see the Troubleshooting appendix in the *QLogic HCA and InfiniPath Software User Guide*.

Cabling and Switches

The cable installation uses a standard InfiniBand (IB) 4X cable. Any InfiniBand cable that has been qualified by the vendor should work. For SDR, the longest passive copper IB cable that QLogic has currently qualified is 20 meters. For DDR-capable adapters and switches, the DDR-capable passive copper cables cannot be longer than 10 meters. Active cables can eliminate some of the cable length restrictions.

InfiniBand switches are available through QLogic.

NOTE:

If you are using the QLE7240 or QLE7280 and want to use DDR mode, then DDR-capable switches must be used.

The copper cables listed in [Table 4-2](#) are available from QLogic:

Table 4-2. QLogic InfiniBand Cables

Product Number	Description
7104-1M-Cable	4x-4x cable—1 meter
7104-2M-Cable	4x-4x cable—2 meters
7104-3M-Cable	4x-4x cable—3 meters
7104-4M-Cable	4x-4x cable—4 meters
7104-5M-Cable	4x-4x cable—5 meters
7104-6M-Cable	4x-4x cable—6 meters
7104-7M-Cable	4x-4x cable—7 meters
7104-8M-Cable	4x-4x cable—8 meters
7104-9M-Cable	4x-4x Cable—9 meters
7104-10M-Cable	4x-4x cable—10 meters
7104-12M-Cable	4x-4x cable—12 meters (SDR only)
7104-14M-Cable	4x-4x cable—14 meters (SDR only)
7104-16M-Cable	4x-4x cable—16 meters (SDR only)
7104-18M-Cable	4x-4x cable—18 meters (SDR only)

For cabling instructions, see [“Cabling the Adapter to the InfiniBand Switch” on page 4-17.](#)

Optical Fibre Option

The QLogic adapter also supports connection to the switch by means of optical fibres through optical media converters such as the EMCORE™ QT2400. Not all switches support these types of convertors. For more information on the EMCORE convertor, see www.emcore.com.

Intel® and Zarlink™ also offer optical cable solutions. See www.intel.com and www.zarlink.com for more information.

Configuring the BIOS

To achieve the best performance with QLogic adapters, you need to configure your BIOS with specific settings. The BIOS settings, which are stored in non-volatile memory, contain certain parameters characterizing the system. These parameters may include date and time, configuration settings, and information about the installed hardware.

There are two issues concerning BIOS settings of which you need to be aware:

- Advanced Configuration and Power Interface (ACPI) needs to be enabled.
- Memory Type Range Registers (MTRR) mapping needs to be set to “Discrete”.

MTRR is used by the InfiniPath driver to enable write combining to the on-chip transmit buffers. This option improves write bandwidth to the QLogic chip by writing multiple words in a single bus transaction (typically 64 bytes). This option applies only to x86_64 systems.

However, some BIOS' do not have the MTRR mapping option. It may have a different name, depending on the chipset, vendor, BIOS, or other factors. For example, it is sometimes referred to as *32 bit memory hole*, which must be enabled.

You can check and adjust these BIOS settings using the BIOS Setup utility. For specific instructions, follow the hardware documentation that came with your system.

QLogic also provides a script, `ipath_mtrr`, which sets the MTRR registers. These registers enable maximum performance from the InfiniPath driver. This script can be run after the InfiniPath software has been installed. It needs to be run after each system reboot.

For more details, see “[BIOS Settings](#)” on page A-1.

Safety with Electricity

Observe these guidelines and safety precautions when working around computer hardware and electrical equipment:

- Locate the power source shutoff for the computer room or lab where you are working. This is where you will turn OFF the power in the event of an emergency or accident. Never assume that power has been disconnected for a circuit; always check first.
- Do not wear loose clothing. Fasten your tie or scarf, remove jewelry, and roll up your sleeves. Wear safety glasses when working under any conditions that might be hazardous to your eyes.
- Shut down and disconnect the system's power supply from AC service before you begin work, to insure that standby power is not active. Power off all attached devices such as monitors, printers, and external components. Note that many motherboards and power supplies maintain standby power at all times. Inserting or removing components while standby is active can damage them.
- Use normal precautions to prevent electrostatic discharge, which can damage integrated circuits.

Unpacking Information

This section provides instructions for safely unpacking and handling the QLogic adapter. To avoid damaging the adapter, always take normal precautions to avoid electrostatic discharge.

Verify the Package Contents

The QLogic adapter system should arrive in good condition. Before unpacking, check for any obvious damage to the packaging. If you find any obvious damage to the packaging or to the contents, please notify your reseller immediately.

List of the Package Contents

The package contents for the QLE7240 adapter are:

- QLogic QLE7240
- Additional short bracket
- Quick Start Guide

Standard PCIe risers can be used, typically supplied by your system or motherboard vendor.

The package contents for the QLE7280 adapter are:

- QLogic QLE7280
- Additional short bracket
- Quick Start Guide

Standard PCIe risers can be used, typically supplied by your system or motherboard vendor.

The package contents for the QLE7140 adapter are:

- QLogic QLE7140
- Quick Start Guide

Standard PCIe risers can be used, typically supplied by your system or motherboard vendor. The contents are illustrated in [Figure 4-2](#).

The package contents for the QHT7140 adapter are:

- QLogic QHT7140
- HTX riser card for use in 1U or 2U chassis
- Quick Start Guide

The contents are illustrated in [Figure 4-3](#).

The IBA6120, IBA6110, and IBA7220 are the QLogic ASICs, which are the central components of the interconnect. The location of the IBA7220 ASIC on the adapter is shown in [Figure 4-1](#). The location of the IBA6120 ASIC on the adapter is shown in [Figure 4-2](#). The location of the IBA6110 ASIC on the adapter is shown in [Figure 4-3](#).

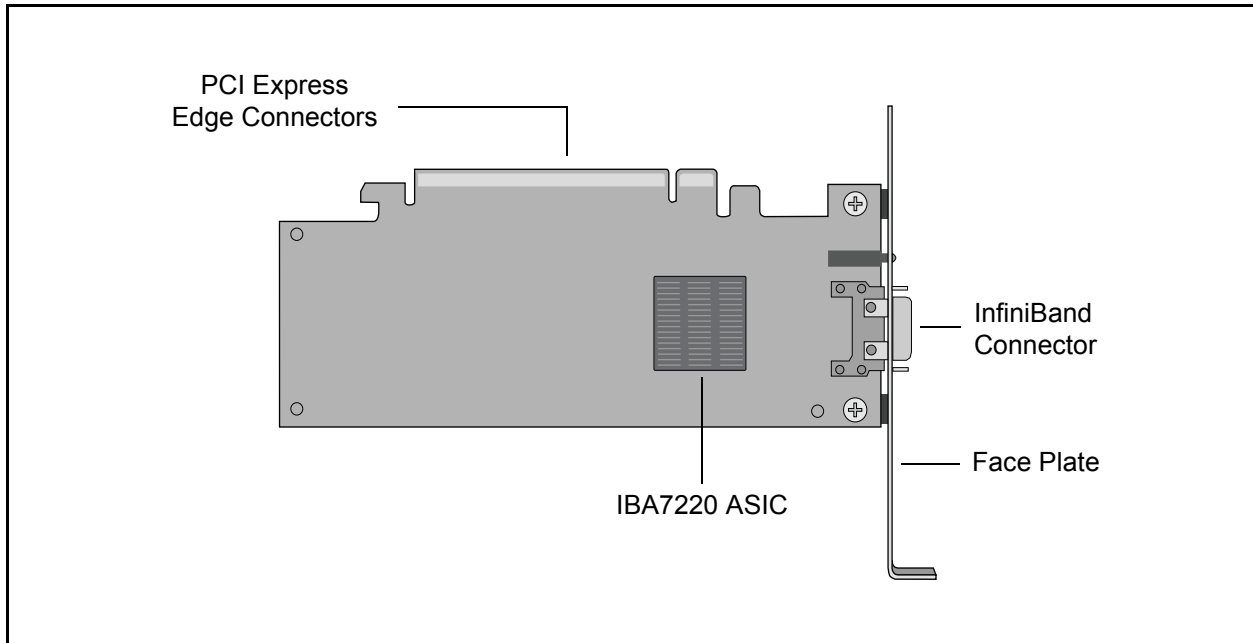


Figure 4-1. QLogic QLE7280 with IBA7220 ASIC

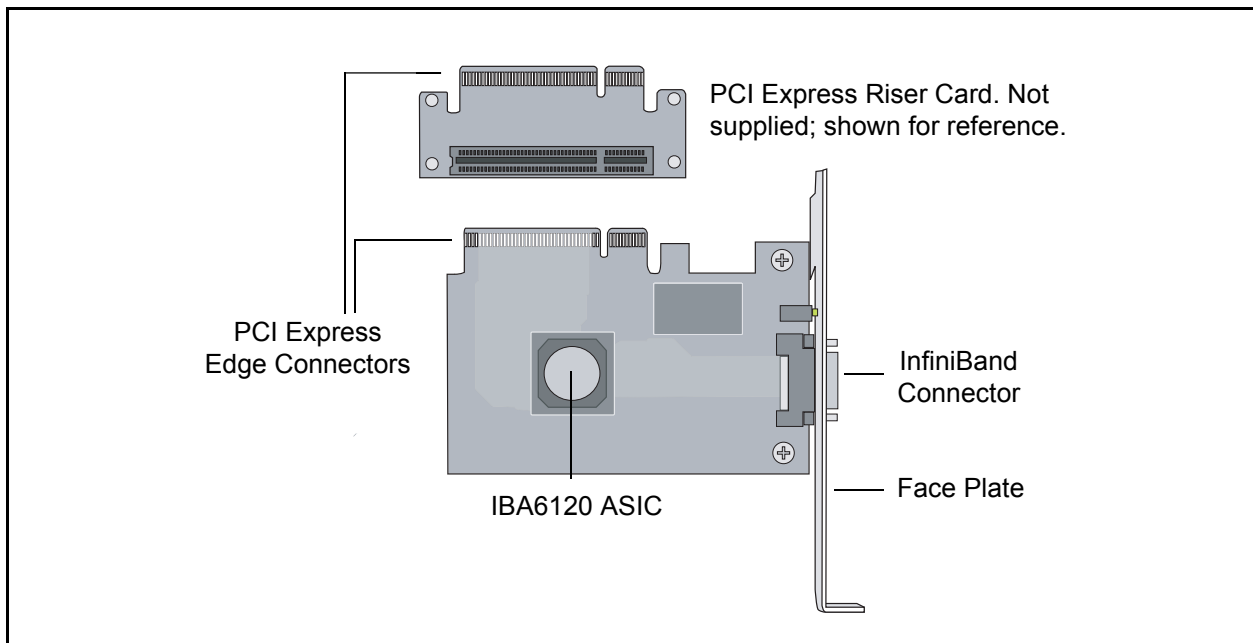


Figure 4-2. QLogic QLE7140 Card with Riser, Top View

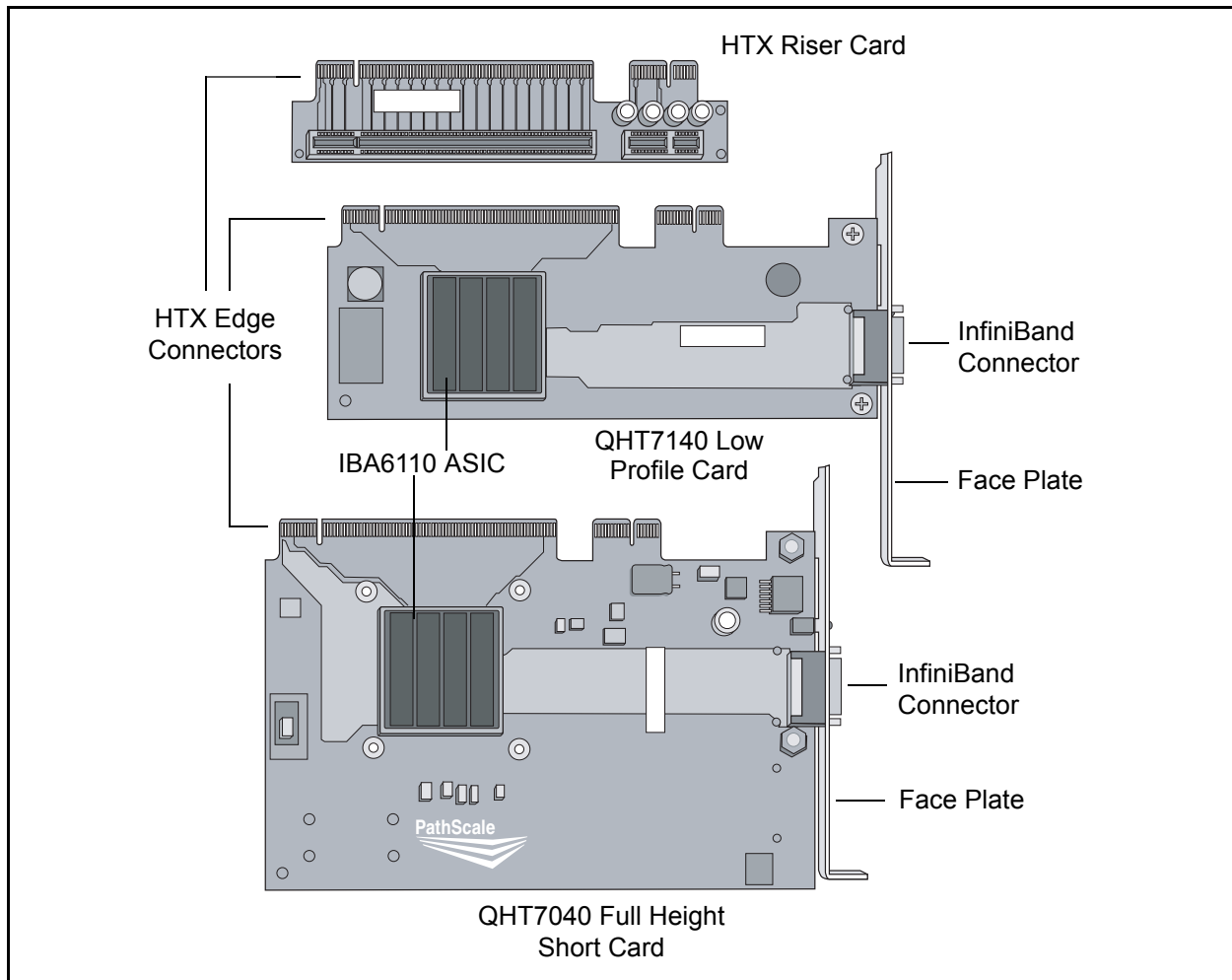


Figure 4-3. QLogic QHT7040/QHT7140 Full and Low Profile Cards with Riser, Top View

Unpacking the QLogic Adapter

Follow these steps when unpacking the QLogic adapter:

1. When unpacking, ground yourself before removing the QLogic adapter from the anti-static bag.
2. Grasping the QLogic adapter by its face plate, pull the adapter out of the anti-static bag. Handle the adapter only by its edges or the face plate. Do not allow the adapter or any of its components to touch any metal parts.
3. After checking for visual damage, store the adapter and the riser card in their anti-static bags until you are ready to install them.

Hardware Installation

This section contains hardware installation instructions for the QLE7240, QLE7280, QLE7140, QHT7040, and QHT7140.

Hardware Installation for QLE7240, QLE7280, or QLE7140 with PCI Express Riser

Installation for the QLE7240, QLE7280, and QLE7140 is similar. The following instructions are for the QLE7140, but can be used for any of these three adapters.

Most installations will be in 1U and 2U chassis, using a PCIe right angle riser card. This results in an installation of the adapter that is parallel to the motherboard. This type of installation is described first. Installation in a 3U chassis is described in [“Hardware Installation for the QHT7140 Without an HTX Riser” on page 4-16](#).

Installing the QLogic QLE7140 in a 1U or 2U chassis requires a PCIe right angle riser card.

A taller riser card can be used if necessary. The QLE7140 can connect to any of the standard compatible PCI Express riser cards.

Dual Adapter Installation

If you have a motherboard with dual PCIe slots, dual adapters can be installed. The adapters must match. For example, on a motherboard with two x16 slots, dual QLE7280 adapters can be installed, but not a QLE7240 adapter and a QLE7280 adapter. Check the design of your motherboard to see how riser cards can be used.

Follow the instructions in [“Installation Steps” on page 4-9](#).

See the Using MPI section in the *QLogic HCA and InfiniPath Software User Guide* for information on using the `IPATH_UNIT` environment variable to control which HCA to use.

Installation Steps

To install the QLogic adapter with a PCIe riser card:

1. The BIOS should already be configured properly by the motherboard manufacturer. However, if any additional BIOS configuration is required, it will usually need to be done before installing the QLogic adapter. See [“Configuring the BIOS” on page 4-4](#).
2. Shut down the power supply to the system into which you will install the QLogic adapter.
3. Take precautions to avoid electrostatic damage (ESD) to the cards by properly grounding yourself or touching the metal chassis to discharge static electricity before handling the cards.

4. Remove the cover screws and cover plate to expose the system's motherboard. For specific instructions on how to do this, follow the hardware documentation that came with your system.
5. Locate the PCIe slot on your motherboard. Note that the PCIe slot has two separate sections, with the smaller slot opening located towards the front (see [Figure 4-4](#)). These two sections correspond to the shorter and longer connector edges of the adapter and riser.

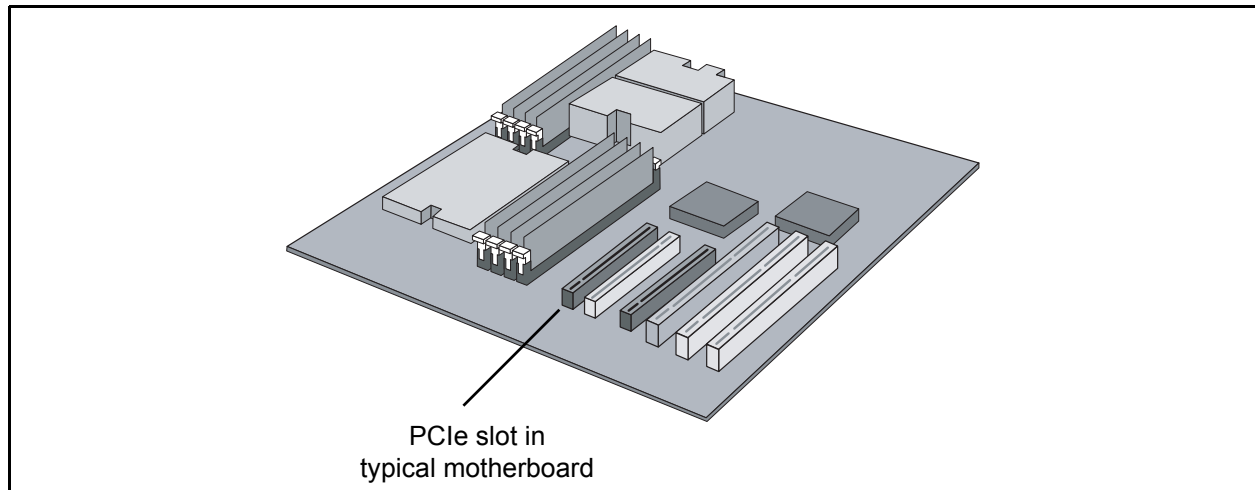


Figure 4-4. PCIe Slot in a Typical Motherboard

6. Determine if a blanking panel is installed in your chassis. If it is, remove it so that the InfiniBand connector will be accessible. Refer to your system vendor instructions for how to remove the blanking panel.
7. Remove the QLogic adapter from the anti-static bag.
8. Locate the face plate on the connector edge of the card.

9. Connect the QLogic adapter and PCIe riser card together, forming the assembly that you will insert into your motherboard. First, visually line up the adapter slot connector edge with the edge connector of the PCIe riser card (see [Figure 4-5](#)).

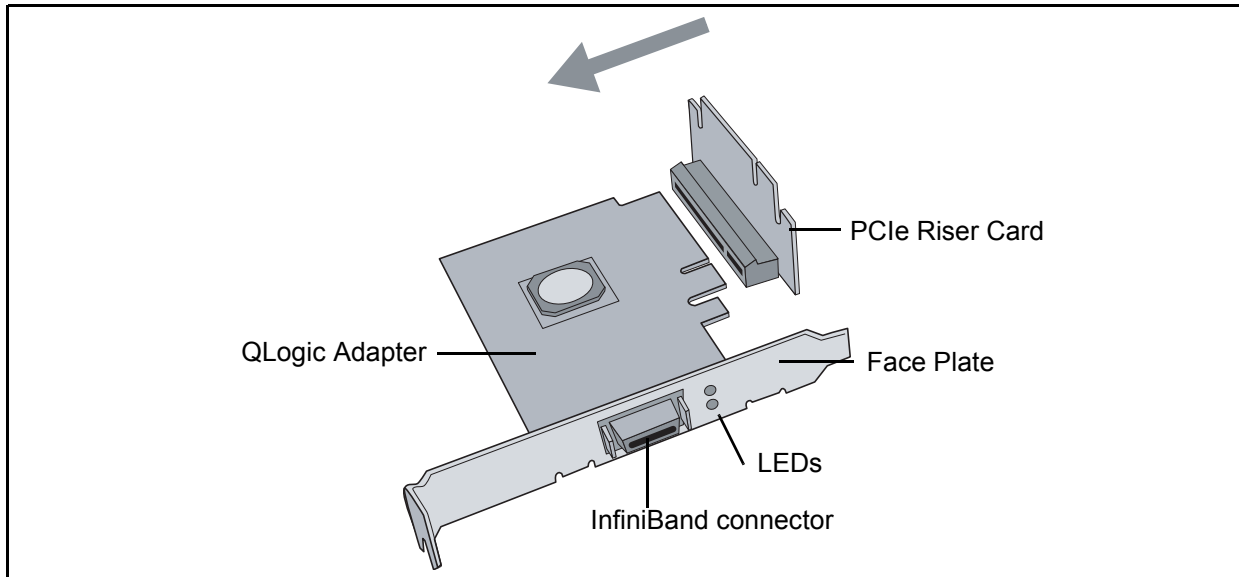


Figure 4-5. QLogic PCIe HCA Assembly with Riser Card

10. Holding the QLogic adapter by its edges, carefully insert the card slot connector into the PCIe riser card edge connector, as show in [Figure 4-5](#). The result is a combined L-shaped assembly of the PCIe riser card and QLogic adapter. This assembly is what you will insert into the PCIe slot on the motherboard in the next step.
11. Turn the assembly so that the riser card connector edge is facing the PCIe slot on the motherboard, and the face plate is toward the front of the chassis.
12. Holding this assembly above the motherboard at about a 45 degree angle, slowly lower it so that the connector on the face plate clears the blanking panel opening of the chassis from the inside. Slowly align the connector edge of the riser card with the motherboard's PCIe slot. The short section of the connector must align with the short section of the slot.

13. Insert the riser assembly into the motherboard's PCIe slot, ensuring good contact. The QLogic adapter should now be parallel to the motherboard and about one inch above it (see [Figure 4-6](#)).

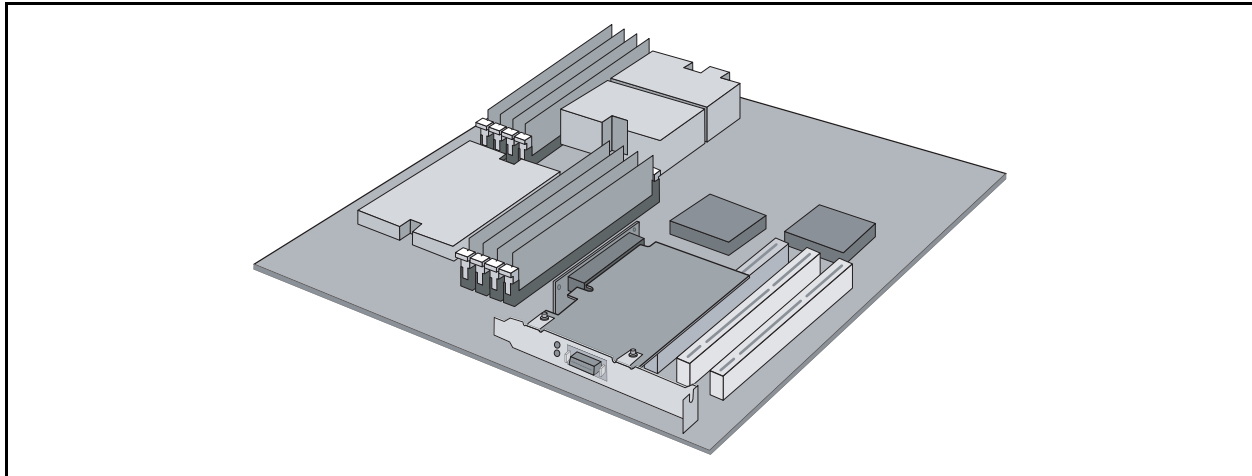


Figure 4-6. Assembled PCIe HCA with Riser

14. Secure the face plate to the chassis. The QLogic adapter has a screw hole on the side of the face plate that can be attached to the chassis with a retention screw. The securing method may vary depending on the chassis manufacturer. Refer to the system documentation for information about mounting details such as mounting holes, screws to secure the card, or other brackets.

The QLogic PCIe HCA with PCIe riser card is now installed. Next, install the cables as described in [“Cabling the Adapter to the InfiniBand Switch” on page 4-17](#). Then test your installation by powering up and verifying link status (see [“Completing the Installation” on page 4-18](#)).

Hardware Installation for QHT7140 with HTX Riser

Most installations will be in 1U and 2U chassis, using the HTX riser card. This results in a horizontal installation of the QHT7140. This type of installation is described in this section. Installation in a 3U chassis is described in [“Hardware Installation for the QHT7140 Without an HTX Riser” on page 4-16](#).

Installation of QLogic QHT7140 in a 1U or 2U chassis requires an HTX riser card.

NOTE:

The illustrations in this section are shown for the full height short form factor. Installation of the HTX low profile form factor follows the same steps.

To install the QLogic adapter with an HTX riser card:

1. The BIOS should be already be configured properly by the motherboard manufacturer. However, if any additional BIOS configuration is required, it will usually need to be done before installing the QLogic adapter. See [“Configuring the BIOS” on page 4-4.](#)
2. Shut down the power supply to the system into which you will install the QLogic adapter.
3. Take precautions to avoid electrostatic discharge (ESD) damage to the cards by properly grounding yourself or touching the metal chassis to discharge static electricity before handling the cards.
4. Remove the cover screws and cover plate to expose the system’s motherboard. For specific instructions on how to do this, follow the hardware documentation that came with your system.
5. Locate the HTX slot on your motherboard. Note that the HTX slot has two separate connectors, corresponding to the connector edges of the adapter. See [Figure 4-7.](#)

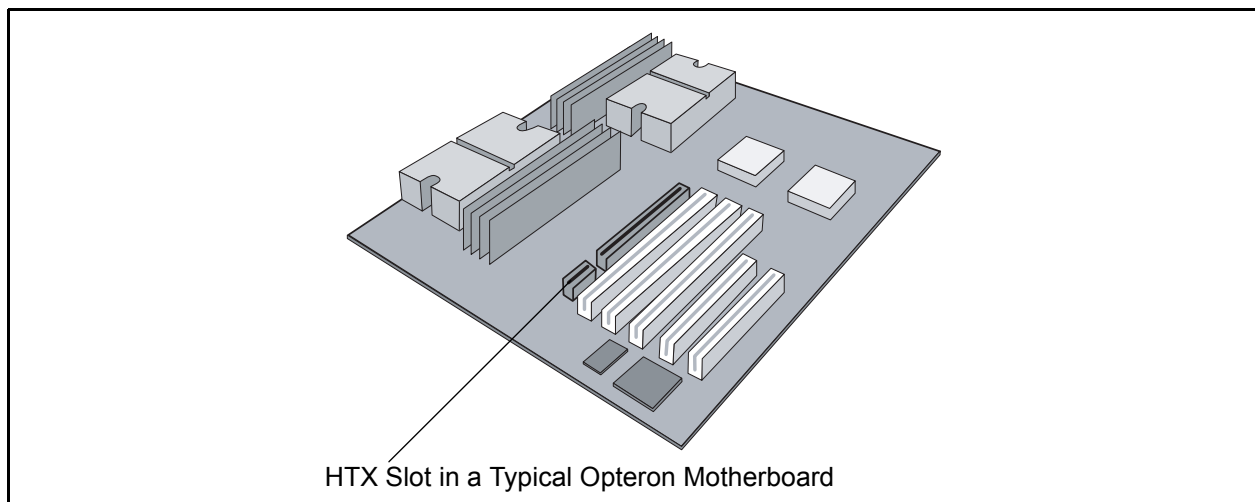


Figure 4-7. HTX Slot

6. Determine if a blanking panel is installed in your chassis. If it is, remove it so that the InfiniBand connector will be accessible. Refer to your system vendor instructions for how to remove the blanking panel.

7. Remove the QLogic QHT7140 from the anti-static bag.

NOTE:

Be careful not to touch any of the components on the printed circuit board during these steps. You can hold the adapter by its face plate or edges.

8. Locate the face plate on the connector edge of the card.
9. Connect the QLogic adapter and HTX riser card together, forming the assembly that you will insert into your motherboard. First, visually line up the adapter slot connector edge with the edge connector of the HTX riser card (see [Figure 4-8](#)).

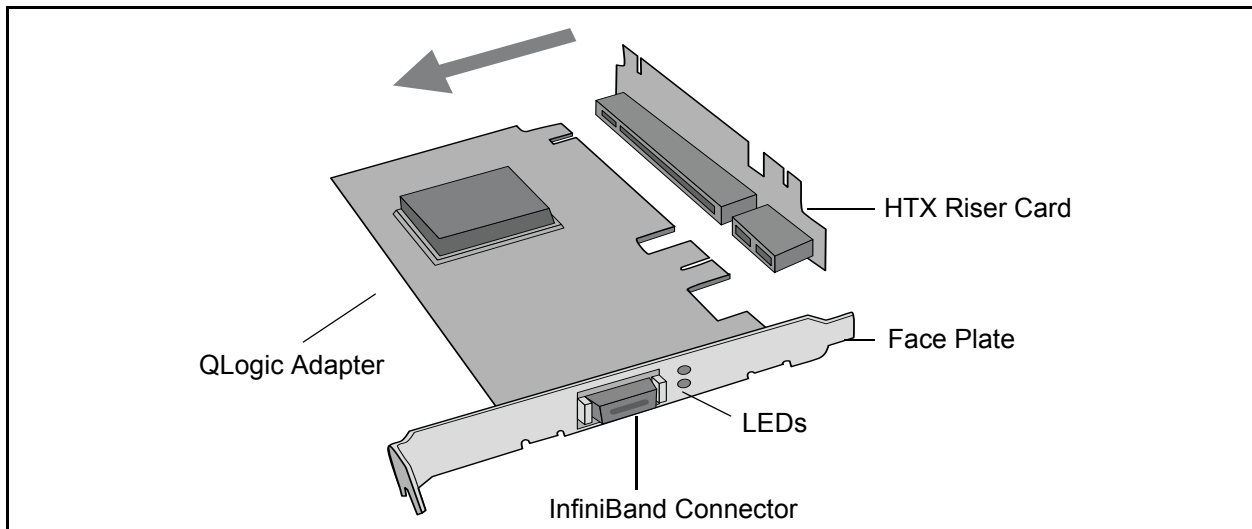


Figure 4-8. QLogic QHT7140 Adapter with Riser Card

10. Holding the QLogic adapter by its edges, carefully insert the card slot connector into the HTX riser card edge connector, as show in [Figure 4-8](#). The result is a combined L-shaped assembly of the HTX riser card and QLogic adapter. This assembly is what you will insert into the HTX slot on the motherboard in the next step.
11. Turn the assembly so that the riser card connector edge is facing the HTX slot on the motherboard, and the face plate is toward the front of the chassis.
12. Holding this assembly above the motherboard at about a 45 degree angle, slowly lower it so that the connector on the face plate clears the blanking panel opening of the chassis from the inside. Slowly align the connector edge of the HTX riser card with the motherboard's HTX slot. The HTX riser and HTX slot must line up perfectly.

13. Insert the HT riser assembly into the motherboard's HTX slot, ensuring good contact. The QLogic adapter should now be parallel to the motherboard and about one inch above it, as shown in [Figure 4-9](#).

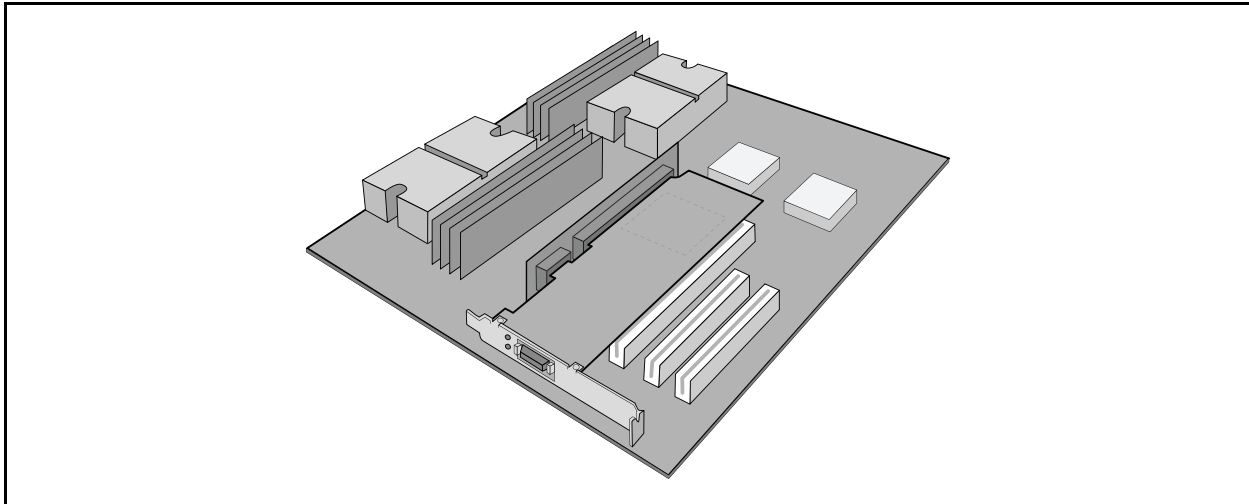


Figure 4-9. Assembled QHT7140 with Riser

14. Secure the face plate to the chassis. The QLogic adapter has a screw hole on the side of the face plate that can be attached to the chassis with a retention screw. The securing method may vary depending on the chassis manufacturer. Refer to the system documentation for information about mounting details such as mounting holes, screws to secure the card, or other brackets.

The QLogic QHT7140 with HTX riser card is now installed. Next, install the cables as described in [“Cabling the Adapter to the InfiniBand Switch”](#) on page 4-17. Then test your installation by powering up and verifying link status (see [“Completing the Installation”](#) on page 4-18).

Hardware Installation for QLE7240, QLE7280, and QLE7140 Without a PCI Express Riser

Installing the QLogic QLE7240, QLE7280, or QLE7140 without a PCI Express riser card requires a 3U or larger chassis.

Installation is similar to the QHT7140 HTX adapter, except that the card slot connectors on these adapters fit into the PCIe slot rather than the HTX slot. Follow the instructions in [“Hardware Installation for the QHT7140 Without an HTX Riser”](#) on page 4-16, substituting the PCIe slot for the HTX slot.

Hardware Installation for the QHT7140 Without an HTX Riser

Installing the QLogic QHT7140 without an HTX riser card requires a 3U or larger chassis. The card slot connectors on the QHT7140 fit into the HTX slot in a vertical installation.

To install the QLogic adapter without the HTX riser card:

1. The BIOS should already be configured properly by the motherboard manufacturer. However, if any additional BIOS configuration is required, it will usually need to be done before installing the QLogic adapter. See [“Configuring the BIOS” on page 4-4](#).
2. Shut down the power supply to the system into which you will install the QLogic adapter.
3. Take precautions to avoid electrostatic discharge (ESD) damage to the cards by properly grounding yourself or touching the metal chassis to discharge static electricity before handling the cards.
4. If you are installing the QLogic adapter into a covered system, remove the cover screws and cover plate to expose the system’s motherboard. For specific instructions on how to do this, follow the hardware documentation that came with your system.
5. Locate the HTX slot on your motherboard (see [Figure 4-7](#)).
6. Remove the QLogic adapter from the anti-static bag. Hold the card by the top horizontal section of the bracket, and the top rear corner of the card. Be careful not to touch any of the components on the printed circuit card.
7. Without fully inserting, gently align and rest the HTX card’s gold fingers on top of the motherboard’s HTX slot.

8. Insert the card by pressing firmly and evenly on the top of the horizontal bracket and the top rear corner of the card simultaneously. The card should insert evenly into the slot. Be careful not to push, grab, or put pressure on any other part of the card, and avoid touching any of the components. See [Figure 4-10](#).

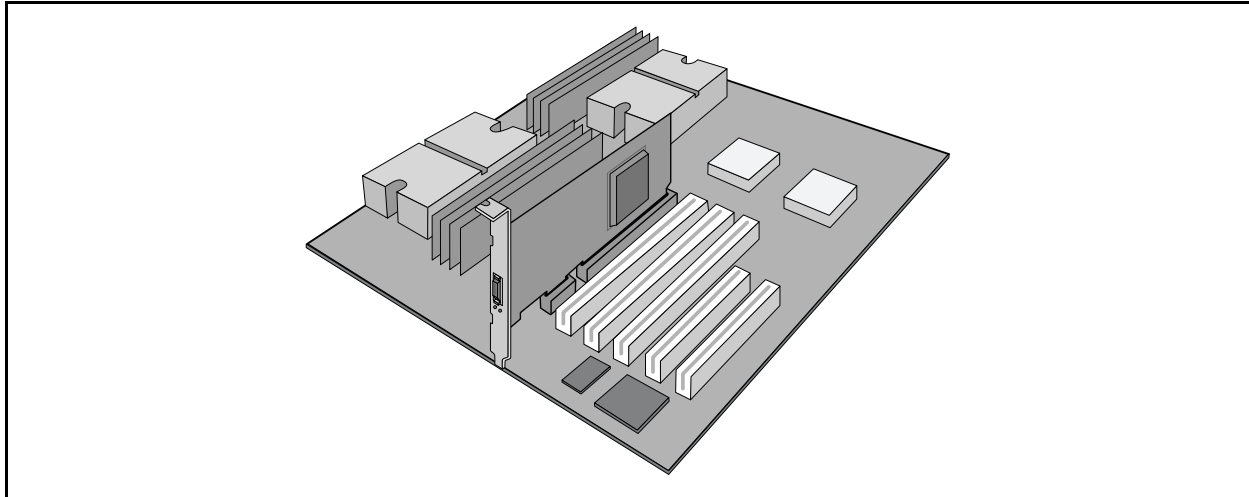


Figure 4-10. QHT7140 Without Riser Installed in a 3U Chassis

9. Secure the face plate to the chassis. The QLogic adapter has a screw hole on the side of the face plate that can be attached to the chassis with a retention screw. The securing method may vary depending on the chassis manufacturer. Refer to the system documentation for information about mounting details such as mounting holes, and screws to secure the card, or other brackets.

Next, install the cables, as described in [“Cabling the Adapter to the InfiniBand Switch”](#) on page 4-17. Then test your installation by powering up the system (see [“Completing the Installation”](#) on page 4-18).

Switch Configuration and Monitoring

The QLogic interconnect is designed to work with all InfiniBand-compliant switches. Follow the vendor documentation for installing and configuring your switches.

Cabling the Adapter to the InfiniBand Switch

Follow the recommendations of your cable vendor for cable management and proper bend radius.

The QLE7240, QLE7280, QLE7140, QHT7040, and QHT7140 adapters are all cabled the same way.

To install the InfiniBand cables:

1. Check that you have removed the protector plugs from the cable connector ends.
2. Different vendor cables might have different latch mechanisms. Determine if your cable has a spring-loaded latch mechanism.
 - If your cable is spring-loaded, grasp the metal shell and pull on the plastic latch to release the cable. To insert, push and the cable snaps into place. You will hear a short “click” sound from the cable connector when it snaps in.
 - If your cable latch mechanism is not spring-loaded, push on the metal case, then push the plastic latch to lock the cable in place.
3. The InfiniBand cables are symmetric; either end can be plugged into the switch. Connect the InfiniBand cable to the connector on the QLogic QLE7240, QLE7280, QLE7140 or QHT7140. Depress the side latches of the cable when connecting. (On some cables, this latch is located at the top of the cable connector.) Make sure the lanyard handle on the cable connector is slid forward toward the card connector until fully engaged.
4. Connect the other end of the cable to the InfiniBand switch.

Completing the Installation

To complete the hardware installation:

1. Complete any other installation steps for other components.
2. Replace the cover plate and back panel.
3. Verify that the power cable is properly connected.
4. Turn on the power supply and boot the system normally.
5. Watch the LED indicators. The LEDs will flash only once, briefly, at power-up. The LEDs are functional only after the InfiniPath software has been installed, the driver has been loaded, and the system is connected to an InfiniBand switch. To use the LEDs to check the state of the adapter, see [“LED Link and Data Indicators” on page 5-29](#).

5 Software Installation

This section provides instructions for installing the InfiniPath and OpenFabrics software. The InfiniPath software includes drivers, protocol libraries, QLogic's implementation of the MPI message passing standard, and example programs, including benchmarks. A complete list of the provided software is in [“Software Components” on page 2-4](#).

Cluster Setup

Information on clusters, supported distributions and kernels, and environment setup is provided in [“Types of Nodes in a Cluster Environment” on page 5-1](#), [“Supported Linux Distributions” on page 5-2](#), and [“Setting Up Your Environment” on page 5-2](#).

Types of Nodes in a Cluster Environment

In a cluster environment, different nodes can be used for different functions, such as launching jobs, developing software, or running jobs. The nodes are defined as follows:

- **Front end node.** This node launches jobs.
- **Compute node.** This node runs jobs.
- **Development or build node.** These are the machines on which examples or benchmarks can be compiled.

Any machine can serve any combination of these three purposes, but a typical cluster has many compute nodes and just a few (or only one) front end nodes. The number of nodes used for development will vary. These node names are used throughout this guide.

Supported Linux Distributions

The currently supported distributions and associated Linux kernel versions for InfiniPath and OpenFabrics are listed in [Table 5-1](#). The kernels are the ones that shipped with the distributions.

Table 5-1. InfiniPath/OpenFabrics Supported Distributions and Kernels

Distribution	InfiniPath/OpenFabrics Supported Kernels
Fedora 6	2.6.22 (x86_64)
Red Hat® Enterprise Linux® 4.4, 4.5, 4.6 (RHEL4.4, 4.5, 4.6)	2.6.9-42 (U4), 2.6.9-55 (U5) 2.6.9-67 (U6) (x86_64)
CentOS 4.4, 4.5 (Rocks 4.4, 4.5, 4.6)	2.6.9-42, 2.6.9.55, 2.6.9-67 (x86_64)
Scientific Linux 4.4, 4.5, 4.6	2.6.9-42, 2.6.9.55, 2.6.9-67 (x86_64)
Red Hat Enterprise Linux 5.0 (RHEL5.0), RHEL 5.1	2.6.18-8, 2.6.18-53 (x86_64)
CentOS 5.0, 5.1 (Rocks 5.0, 5.1)	2.6.18, 2.6.18-53 (x86_64)
Scientific Linux 5.0, 5.1	2.6.18, 2.6.18-53 (x86_64)
SUSE® Linux Enterprise Server (SLES 10 GM, SP 1)	2.6.16.21, 2.6.16.46 (x86_64)

NOTE:

Fedora Core 4 and Fedora Core 5 are not supported in the InfiniPath 2.2 release.

Setting Up Your Environment

Keep the following in mind when setting up the environment:

- The `kernel-devel` (for Red Hat and Red Hat-derived kernels) RPM or `kernel-source` (for SLES) RPM for your distribution/kernel must be installed.
- The runtime and build environments must be the same. Compatibility between executables built on different Linux distributions cannot be guaranteed.
- Install `sysfsutils` for your distribution before installing the OpenFabrics RPMs, as there are dependencies. Check your distribution's documentation for information about `sysfsutils`. This package is called `udev` on SLES 10.

- Make sure that all previously existing (stock) OpenFabrics RPMs are uninstalled. See “[Removing Software Packages](#)” on page 5-33 for more information on uninstalling. If you are using RHEL5, make sure that `opensm-*` is manually uninstalled. See “[Version Number Conflict with opensm-* on RHEL5 Systems](#)” on page A-4 for more information.
- Among the many optional packages that each distribution offers, the InfiniPath software requires `openssh` and `openssh-server` and, if the Multi-Purpose Daemon (MPD) job launcher or the `ipath_mtrr` script is to be used, `python`. These packages must be on every node. Note that in the SLES 10 distribution, `openssh-server` is a part of the `openssh` package.
- It is possible to have a cluster running with different kernel versions. However, QLogic recommends and supports clusters where all nodes run equivalent software.
- Different distributions require different versions of the InfiniPath software distribution, as described in “[Installing the InfiniPath and OpenFabrics RPMs](#)” on page 5-6.

Downloading and Unpacking the InfiniPath and OpenFabrics Software

This section assumes that the correct Linux kernel, a supported distribution, and suggested packages (see “[Setting Up Your Environment](#)” on page 5-2) have been installed on every node.

Choosing the Appropriate Download Files

Several components are available as separate downloads, as noted in [Table 5-2](#). All files are available from the QLogic web site: <http://www.qlogic.com>.

Table 5-2. Available Packages

Package	Description	Comments
InfiniPath 2.2 software	Multiple interdependent RPM packages make up the InfiniPath and OpenFabrics software. Includes QLogic VNIC.	Associated documentation includes Readme and Release Notes for InfiniPath, and the QLogic HCA and InfiniPath software installation and user’s guides.
QLogic SRP	QLogic’s version of SRP. Available as a separate download.	Has associated Readme, Release Notes, and documentation.

Table 5-2. Available Packages (Continued)

Package	Description	Comments
Enablement tools	Subset of QLogic InfiniBand Fabric Suite. Available as a separate download.	Has associated Readme, Release Notes, and documentation.
QLogic Fabric Suite	The QLogic Fabric Suite CD is available separately for purchase. This CD provides management tools and the QLogic host-based SM.	CD may be purchased separately. Follow the links on the QLogic download page. Documentation is included.
OFED 1.3 source	Optional source for the Open Fabrics Enterprise Distribution (OFED 1.3) libraries and utilities as built and shipped with the InfiniPath 2.2 release. Packaged as a single <code>tar</code> file.	Documentation is included in the <code>tar</code> file.

Follow the links for your distribution to the desired download, then follow the instructions on the web page for downloading the files to a convenient directory.

Instructions for installing the InfiniPath and OpenFabrics software are provided in the following sections.

QLogic SRP and the enablement tools have separate instructions that are available from the download page on the QLogic web site.

NOTE:

Installation information for the QLogic Fabric Suite and the OFED 1.3 source packages is included with the packages.

Unpacking the InfiniPath `tar` File

After downloading the InfiniPath 2.2 `tar` file, type:

```
$ tar zxvf InfiniPath2.2-xxx-yyy.tgz
```

In this command, `xxx` is the distribution identifier (RHEL4 or SLES10), and `yyy` is the platform architecture, `x86_64`. The `tar` command creates a directory based on the `tar` file name and places the RPMs and other files in this directory.

NOTE:

The files can be downloaded to any directory. The install process will create and install the files in the correct directories. The locations of the directories after installation are listed in [“Installed Layout” on page 5-9](#).

The RPMs are organized as follows:

```
InfiniPath_license.txt,LEGAL.txt (top level)
Documentation/
InfiniPath/
InfiniPath-Devel/
InfiniPath-MPI/
OpenFabrics/
OpenFabrics-Devel/
OpenSM/
OtherHCAs/
OtherHCAs-Devel/
OtherMPIs/
```

A complete list of RPMs is in [“RPM Descriptions” on page C-1](#).

Check for Missing Files or RPMs

Run the `rpm` command with the `--verify` option to check if there are files missing from the RPMs. For example:

```
$ rpm -a --verify 'InfiniPath-MPI/mpi*' 'InfiniPath/infinipath*'
```

In the case of OpenFabrics RPMs, the verification command is slightly different, since these RPMs have many different prefixes:

```
$ rpm --verify rpm_name_pre
```

`rpm_name_pre` is the descriptive name that precedes the version and repository identifiers in an RPM. For example:

```
$ rpm --verify libibverbs
```

This command verifies the contents of:

```
libibverbs-2.2-xxx.1_1.yyy.x86_64.rpm
```

The `rpm` command cannot check for missing RPMs. Use `ipath_checkout` after installation to flag missing RPMs. See [“Adapter Settings” on page 5-30](#) for more information.

Installing the InfiniPath and OpenFabrics RPMs

Linux distributions of InfiniPath and OpenFabrics software are installed from binary RPMs. RPM is a Linux packaging and installation tool used by Red Hat, SUSE, and CentOS. There are multiple interdependent RPM packages that make up the InfiniPath and OpenFabrics software.

The OpenFabrics kernel module support is now part of the InfiniPath RPMs. This kernel module includes support for kernel-based protocols such as IPoIB, SRP, the multicast communications manager, and the subnet administration agent.

Programs that incorporate the user IB verbs interfaces, such as diagnostics, benchmarks, verbs-based MPIs (for example, Intel MPI Version 3.0), and SDP sockets must have RPMs installed in the OpenFabrics directories. See [Appendix C “RPM Descriptions”](#) for a complete description of the RPMs’ contents.

Each set of RPMs uses a build identifier *xxx* and a distribution identifier *yyy*. The distribution identifiers for this release are `rhel14` and `sles10`. The RPM distribution identifiers are listed in [Table 5-3](#) with the associated operating systems.

Table 5-3. InfiniPath and OpenFabrics RPMs to Use for Each Node in a Cluster

Distribution Identifier	Used On
<code>rhel14</code>	Fedora 6, Red Hat Enterprise Linux 4.4 (RHEL4.4), RHEL4.5, RHEL 4.6, CentOS 4.4-4.6 (Rocks 4.4-4.6), Scientific Linux 4.4-4.6, RHEL5.0-5.1, CentOS 5.0-5.1, Scientific Linux 5.0-5.1, for x86_64 systems
<code>sles10</code>	SLES 10 and SLES 10 SP1 for x86_64 systems

NOTE:

RPMs contain `config` files. Your current `config` files will not be overwritten when the new RPMs are installed. New `config` files contain the suffix `.rpmnew` and can be found in the `/etc/sysconfig` directory. Check the new files to see if there is anything you want to add to your standard `config` files.

Choosing the RPMs to Install

Although QLogic recommends that all RPMs are installed on all nodes, some are optional depending on which type of node is being used. To see which RPMs are required or optional for each type of node, according to its function as a compute node, front end node, development machine, or Subnet Manager (SM), see [Appendix C “RPM Descriptions”](#).

- Installing all the RPMs (except for OpenSM) on all nodes is easiest.
- If you do not plan to run all of the available programs, you can install a subset of the RPMs. See [Table 5-4](#).
- Install the `OpenSM` RPM only if you do not plan to use a switch-based or host-based SM. The `OpenSM` RPM is normally installed on the node on which it will be used. If installed, it is *on* by default. This behavior can be modified. See [“OpenSM” on page 5-23](#) for more information.
- Install the `infinipath` RPM on the all nodes where you install the `mpi-frontend` RPM.

Table 5-4. RPMs to Install

If you want to:	Then install:
Use InfiniPath, QLogic MPI, and basic OpenFabrics protocols such as IPoIB and SRP	The RPM sets in the <code>InfiniPath*</code> and <code>InfiniPath-MPI</code> directories
Run OpenFabrics programs that use the IB verbs interfaces, such as diagnostics, benchmarks, verbs-based MPI, VNIC, and SDP sockets	The RPM sets in the <code>InfiniPath*</code> and the <code>OpenFabrics*</code> directories
Use all of the above programs	The RPM sets in the <code>InfiniPath*</code> , <code>InfiniPath-MPI</code> , and the <code>OpenFabrics*</code> directories
Add OpenSM as a your subnet manager	The RPM sets in the <code>OpenFabrics</code> and <code>OpenSM</code> directories. (OpenSM has dependencies on OpenFabrics.)
Use other MPIs (MVAPICH and Open MPI) that have been compiled with Path-Scale, GNU, PGI, and Intel compilers. Includes test files.	The RPM sets in the <code>InfiniPath*</code> , <code>InfiniPath-MPI</code> , and <code>OtherMPIs</code> directories
Use other HCAs with OpenFabrics	The RPM sets in the <code>OpenFabrics*</code> and <code>OtherHCAs*</code> directories

Using rpm to Install InfiniPath and OpenFabrics

The RPMs need to be available on each node on which they will be used. One way to do this is to copy the RPMs to a directory on each node that will need them. Another way is to put the RPMs in a directory that is accessible (e.g., via Network File System (NFS)) to every node. After making sure the RPMs are available on each node, login as root and, for the `InfiniPath` and `InfiniPath-MPI` RPMs, run the command:

```
# rpm -Uvh InfiniPath/*.rpm InfiniPath-MPI/*.rpm \  
InfiniPath-Devel/*.rpm
```

The output during the install process will be similar to the following. It will vary depending on which kernel you are using:

```
Preparing...  
##### [100%]  
/usr/src/infinipath/drivers/drivers-2.6.22_FC6  
/usr/src/infinipath/drivers  
/usr/src/infinipath/drivers  
1:infinipath-kernel  
##### [100%]  
Building and installing InfiniPath modules for 2.6.22_FC6  
2.6.22.9-61.fc6 kernel...
```

Check that all older stock OFED RPMs have been uninstalled ([“Uninstalling OFED 1.3 Software” on page 5-33](#)), then, for the OpenFabrics RPMs (including VNIC), run the command:

```
# rpm -Uvh OpenFabrics/*.rpm OpenFabrics-Devel/*.rpm
```

Install the `OpenSM` RPM only if you do not plan to use a switch-based or host-based SM. The `OpenSM` RPM is normally installed on the node where it will be used. If installed, it is *on* by default. This behavior can be modified. See [“OpenSM” on page 5-23](#) for more information.

```
# rpm -Uvh OpenSM/*.rpm
```

The `opensm-devel` RPM is located with the other `OpenFabrics-Devel` RPMs.

If you want to use other MPIs, run the command:

```
# rpm -Uvh InfiniPath/*.rpm InfiniPath-MPI/*.rpm \  
InfiniPath-Devel/*.rpm OpenFabrics/*.rpm OpenFabrics-Devel/*.rpm \  
OtherMPIs/*.rpm
```

If you want to use other HCAs, run the command:

```
# rpm -Uvh OpenFabrics/*.rpm OpenFabrics-Devel/*.rpm \  
OtherHCAs/*.rpm OtherHCAs-Devel/*.rpm
```

NOTE:

Parallel command starters can be used for installation on multiple nodes, but this subject is beyond the scope of this document.

Installed Layout

The default installed layout for the InfiniPath software is described in the following paragraphs.

The InfiniPath shared libraries are installed in:

```
/usr/lib for 32-bit applications  
/usr/lib64 for 64-bit applications
```

MPI include files are in:

```
/usr/include
```

MPI programming examples and the source for several MPI benchmarks are in:

```
/usr/share/mpich/examples
```

InfiniPath utility programs, as well as MPI utilities and benchmarks, are installed in:

```
/usr/bin
```

Documentation is found in:

```
/usr/share/man  
/usr/share/doc/infinipath  
/usr/share/doc/mpich-infinipath
```

Configuration files are found in:

```
/etc/sysconfig
```

Init scripts are found in:

```
/etc/init.d
```

The InfiniPath kernel modules in this release are installed in:

```
/lib/modules/`uname -r`/updates
```

Putting the modules in this directory avoids replacing kernel modules that may be provided by your Linux distribution; you may want to use these modules if the InfiniPath software is removed. Modules are renamed if they can cause conflicts. For example, the module `ipath_core.ko` was previously renamed to `ib_ipath.ko`, but conflicts can arise if `ipath_core.ko` is still present. If it is found during installation of the `infinipath-kernel` RPM, `ipath_core.ko` is renamed `ipath_core.ko.bak`.

Other OFED-installed modules may also be in this directory; these are also renamed if found during the install process.

Starting the InfiniPath Service

If this is the initial installation of InfiniPath (see [“Lockable Memory Error on Initial Installation of InfiniPath” on page A-7](#)), or if you have installed VNIC with the OpenFabrics RPM set, reboot after installing.

If this is an upgrade, you can restart the InfiniPath service without rebooting. To enable the driver, run the command (as root):

```
# chkconfig infinipath on 2345
```

Then start the InfiniPath support (as root) by typing:

```
# /etc/init.d/infinipath start
```

Complete information about starting, stopping, and restarting the services are in [“Starting and Stopping the InfiniPath Software” on page 5-26](#).

When all of the InfiniPath and OpenFabrics software has been installed correctly, the default settings at startup are as follows:

- InfiniPath `ib_ipath` is enabled.
- InfiniPath `ipath_ether` is not running until it is configured. See [“Configuring the ipath_ether Network Interface” on page 5-12](#) for configuration instructions.
- OpenFabrics IPoIB is not running until it is configured. Once enabled, IPoIB is running in connected mode. See [“Configuring the IPoIB Network Interface” on page 5-17](#) for configuration instructions.
- OFED SRP is not running until the module is loaded and the SRP devices on the fabric have been discovered.
- VNIC is not running until it is configured.
- OpenSM is enabled on startup. Either install it on only one node, or disable it on all nodes except where it will be used as an SM.

Other optional drivers can now be configured and enabled, as described in [“InfiniPath and OpenFabrics Driver Overview” on page 5-11](#), [“Configuring the InfiniPath Drivers” on page 5-11](#), and [“OpenFabrics Drivers and Services Configuration and Startup” on page 5-16](#).

InfiniPath and OpenFabrics Driver Overview

The InfiniPath `ib_ipath` module provides low level QLogic hardware support, and is the base driver for both the InfiniPath and OpenFabrics software components. The `ib_ipath` module does hardware initialization, handles InfiniPath-specific memory management, and provides services to other InfiniPath and OpenFabrics modules.

It provides the hardware and hardware management functions for MPI/PSM programs, the `ipath_ether` Ethernet emulation, and general OpenFabrics protocols such as IPoB and SDP. The module also contains a subnet management agent.

Figure 5-1 shows the relationship between the InfiniPath and OpenFabrics software. Not all components are shown.

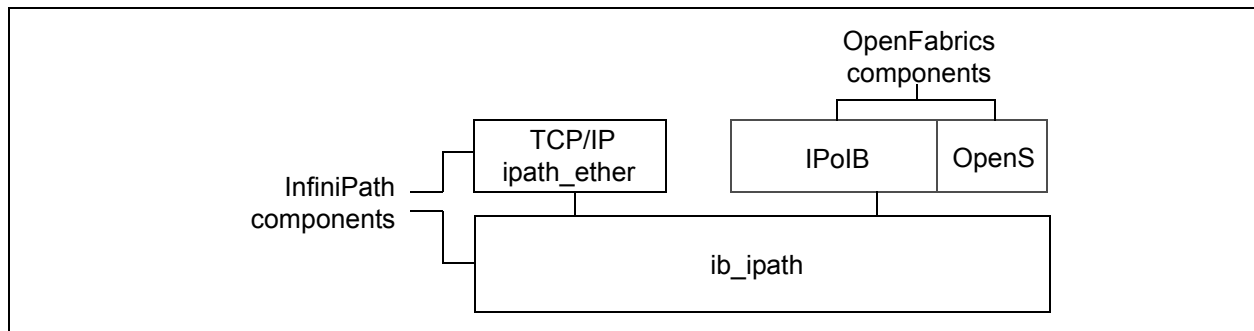


Figure 5-1. Relationship Between InfiniPath and OpenFabrics Software

If you want to enable Transmission Control Protocol-Internet Protocol (TCP/IP) networking for running Ethernet traffic over the InfiniPath link, you can configure the optional `ipath_ether` network interface files.

NOTE:

It is not necessary to configure the `ipath_ether` driver to run MPI jobs.

Optional configurable OpenFabrics components are:

- IPoB network interface
- VNIC
- OpenSM
- SRP (OFED and QLogic modules)
- MPI over uDAPL (can be used by Intel MPI or HP-MPI)

Configuring the InfiniPath Drivers

This section provides information on InfiniPath driver configuration.

Configuring the `ib_ipath` Driver

The `ib_ipath` module provides both low-level InfiniPath support and management functions for OpenFabrics protocols. The `ib_ipath` driver has several configuration variables that set reserved buffers for the software, define events to create trace records, and set the debug level. The startup script for `ib_ipath` is installed automatically as part of the software installation, and normally does not need to be changed.

The primary configuration file for the InfiniPath drivers `ib_ipath` and `ipath_ether`, and other modules and associated daemons, is:

```
/etc/sysconfig/infinipath
```

Normally, this configuration file is set up correctly at installation and the drivers are loaded automatically during system boot once the RPMs have been installed. If you are upgrading, your existing configuration files will not be overwritten.

The device files are:

```
/dev/ipath  
/dev/ipath0, /dev/ipath1, ...
```

The numbered device files allow access to a specific InfiniPath unit.

See the `ib_ipath` man page for more details.

Configuring the `ipath_ether` Network Interface

These instructions are for enabling TCP/IP networking over the InfiniPath link. To enable IPoB networking, see [“Configuring the IPoB Network Interface” on page 5-17](#).

You must create a network device configuration file for the layered Ethernet device on the QLogic adapter. This configuration file will resemble the configuration files for the other Ethernet devices on the nodes.

Two slightly different procedures are given in the following sections for the `ipath` configuration; one for Fedora/RHEL (see [“`ipath_ether` Configuration on Red Hat” on page 5-12](#)) and one for SLES 10 (see [“`ipath_ether` Configuration on SLES” on page 5-14](#)).

Many of the entries that are used in the configuration directions are explained in the file `sysconfig.txt`, located in the following directory:

```
/usr/share/doc/initscripts-*/sysconfig.txt
```

`ipath_ether` Configuration on Red Hat

The following procedure will cause the `ipath_ether` network interfaces to be automatically configured the next time you reboot the system. These instructions are for the Fedora 6, Red Hat Enterprise Linux 4 (RHEL4), and RHEL5 distributions.

Servers typically have two Ethernet devices present, numbered as 0 (`eth0`) and 1 (`eth1`). This example creates a third device, `eth2`.

NOTE:

When multiple QLogic HCAs are present, the configuration for `eth3`, `eth4`, and so on follow the same format as for adding `eth2` in the following example.

1. Check for the number of Ethernet drivers you currently have by typing either one of the following two commands:

```
$ ifconfig -a
$ ls /sys/class/net
```

It is assumed that two Ethernet devices (numbered 0 and 1) are already present.

2. Edit the file `/etc/modprobe.conf` (as root); add the following line:

```
alias eth2 ipath_ether
```

3. Create or edit the following file (as root):

```
/etc/sysconfig/network-scripts/ifcfg-eth2
```

If you are using Dynamic Host Configuration Protocol (DHCP), add the following lines to `ifcfg-eth2`:

```
# QLogic Interconnect Ethernet
DEVICE=eth2
ONBOOT=yes
BOOTPROTO=dhcp
```

If you are using static IP addresses, use the following lines instead, substituting your own IP address for the one in the example. The normal matching netmask is shown.

```
# QLogic Interconnect Ethernet
DEVICE=eth2
BOOTPROTO=static
ONBOOT=YES
IPADDR=192.168.5.101 #Substitute your IP address here
NETMASK="255.255.255.0"#Normal matching netmask
TYPE=Ethernet
```

This change causes the `ipath_ether` Ethernet driver to be loaded and configured during system startup. To check your configuration, and make the `ipath_ether` Ethernet driver available immediately, type the following command (as root):

```
# /sbin/ifup eth2
```

4. Check whether the Ethernet driver has been loaded with:

```
$ lsmod | grep ipath_ether
```

5. Verify that the driver is up with:

```
$ ifconfig -a
```

ipath_ether Configuration on SLES

The following procedure causes the `ipath_ether` network interfaces to be automatically configured the next time you reboot the system. These instructions are for the SLES 10 distribution.

Servers typically have two Ethernet devices present, numbered as 0 (`eth0`) and 1 (`eth1`). This example creates a third device, `eth2`.

NOTE:

- When multiple QLogic HCAs are present, the configuration for `eth3`, `eth4`, and so on follow the same format as for adding `eth2` in the following example. Similarly, in [Step 2](#), add one to the unit number (replace `.../00/guid` with `/01/guid` for the second QLogic interface), and so on.
- The Media Access Control (MAC) address is a unique identifier attached to most forms of networking equipment. [Step 2](#) determines the MAC address to use, and will be referred to as `$MAC` in the subsequent steps. `$MAC` must be replaced in each case with the string printed in [Step 2](#).

As the root user, perform the following steps:

1. Be sure that the `ipath_ether` module is loaded:

```
# lsmod | grep -q ipath_ether || modprobe ipath_ether
```

2. Determine the MAC address that will be used:

```
# sed 's/^\(..:..:..\):...:..\/\1/' \  
/sys/bus/pci/drivers/ib_ipath/00/guid
```

NOTE:

When cutting and pasting commands such as the above from PDF documents, the quotes are special characters and may not be translated correctly.

The output should appear similar to this (six hex digit pairs, separated by colons):

```
00:11:75:04:e0:11
```


The Globally Unique Identifier (GUID) can also be returned by running:

```
# ipath_control -i
$Id: QLogic Release2.2 $ $Date: 2007-09-05-04:16 $
00: Version: ChipABI 2.0, InfiniPath_QHT7140, InfiniPath1 3.2,
PCI 2, SW Compat 2
00: Status: 0xe1 Initted Present IB_link_up IB_configured
00: LID=0x30 MLID=0x0 GUID=00:11:75:00:00:04:e0:11 Serial:
1236070407
```

Removing the middle two 00:00 octets from the GUID in the above output will form the MAC address.

If either [Step 1](#) or [Step 2](#) fails, the problem must be found and corrected before continuing. Verify that the RPMs are installed correctly, and that `infinipath` has started correctly. If problems continue, run `ipathbug-helper` and report the results to your reseller or InfiniPath support organization.

3. Edit the file:

```
/etc/udev/rules.d/30-net_persistent_names.rules
```

If this file does not exist, skip to [Step 4](#).

Check each of the lines, starting with `SUBSYSTEM=`, to find the highest numbered interface. (For standard motherboards, the highest numbered interface will typically be 1.)

Add a new line at the end of the file, incrementing the interface number by one. In this example, it becomes `eth2`. The new line will look like this:

```
SUBSYSTEM=="net", ACTION=="add", SYSFS{address}=="$MAC",
IMPORT="/lib/udev/rename_netiface %k eth2"
```

This will appear as a single line in the file. `$MAC` is replaced by the string from [Step 2](#).

4. Create the network module file:

```
/etc/sysconfig/hardware/hwcfg-eth-id-$MAC
```

Add the following lines to the file:

```
MODULE=ipath_ether
STARTMODE=auto
```

This step will cause the `ipath_ether` Ethernet driver to be loaded and configured during system startup.

5. Create the network configuration file:

```
/etc/sysconfig/network/ifcfg-eth2
```

If you are using Dynamic Host Configuration Protocol (DHCP), add these lines to the file:

```
STARTMODE=onboot  
BOOTPROTO=dhcp  
NAME='InfiniPath Network Card'  
_nm_name=eth-id-$MAC
```

Proceed to [Step 6](#).

If you are using static IP addresses (not DHCP), add these lines to the file:

```
STARTMODE=onboot  
BOOTPROTO=static  
NAME='InfiniPath Network Card'  
NETWORK=192.168.5.0  
NETMASK=255.255.255.0  
BROADCAST=192.168.5.255  
IPADDR=192.168.5.211  
_nm_name=eth-id-$MAC
```

Make sure that you substitute your own IP address for the IPADDR in the example. The BROADCAST, NETMASK, and NETWORK lines need to match for your network.

6. To verify that the configuration files are correct, run the commands:

```
# ifup eth2  
# ifconfig eth2
```

You may have to reboot the system for the configuration changes to take effect.

OpenFabrics Drivers and Services Configuration and Startup

The IPoIB network interface, VNIC interface, and OpenSM components of OpenFabrics can be configured to be on or off. IPoIB is *off* by default; VNIC is *off* by default; OpenSM is *on* by default.

IPoIB, VNIC, OpenSM, SRP, and MPI over uDAPL configuration and startup is explained in more detail in the following sections.

NOTE:

The following instructions work for all supported distributions.

Configuring the IPoIB Network Interface

The following instructions show you how to manually configure your OpenFabrics IPoIB network interface. This example assumes that you are using `sh` or `bash` as your shell, all required InfiniPath and OpenFabrics RPMs are installed, and your startup scripts have been run (either manually or at system boot).

For this example, the IPoIB network is 10.1.17.0 (one of the networks reserved for private use, and thus not routable on the Internet), with a /8 host portion, and therefore requires that the netmask be specified.

This example assumes that no hosts files exist, the host being configured has the IP address 10.1.17.3, and DHCP is not used.

NOTE:

Instructions are only for this static IP address case. Configuration methods for using DHCP will be supplied in a later release.

1. Type the following commands (as root):

```
# ifconfig ib0 10.1.17.3 netmask 0xffffffff00
```

2. To verify the configuration, type:

```
# ifconfig ib0
```

The output from this command will be similar to this:

```
ib0    Link encap:InfiniBand HWaddr
00:00:00:02:FE:80:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
inet  addr:10.1.17.3  Bcast:10.1.17.255  Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST  MTU:4096  Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:128
RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
```

3. Type:

```
# ping -c 2 -b 10.1.17.255
```

The output of the `ping` command will be similar to the following, with a line for each host already configured and connected:

```
WARNING: pinging broadcast address
PING 10.1.17.255 (10.1.17.255) 517(84) bytes of data.
174 bytes from 10.1.17.3: icmp_seq=0 ttl=174 time=0.022 ms
64 bytes from 10.1.17.1: icmp_seq=0 ttl=64 time=0.070 ms (DUP!)
64 bytes from 10.1.17.7: icmp_seq=0 ttl=64 time=0.073 ms (DUP!)
```

The IPoIB network interface is now configured.

NOTE:

- The configuration must be repeated each time the system is rebooted.
- IPoIB-CM (Connected Mode) is enabled by default. If you want to change this to datagram mode, edit the file `/etc/sysconfig/infinipath`. Change this line to:
`IPOIB_MODE="datagram"`

Restart `infinipath` (as root) by typing:
`# /etc/init.d/infinipath restart`

The default `IPOIB_MODE` setting is "CM" for Connected Mode.

Configuring and Administering the VNIC Interface

The VirtualNIC (VNIC) Upper Layer Protocol (ULP) works in conjunction with firmware running on Virtual Input/Output (VIO) hardware such as the SilverStorm Ethernet Virtual I/O Controller (EVIC™) or the InfiniBand/Ethernet Bridge Module for IBM® BladeCenter®, providing virtual Ethernet connectivity.

The VNIC interface must be configured before it can be used. To do so, perform the following steps:

1. Discover EVIC/VEx Input/Output Controllers (IOCs) present on the fabric using `ib_qlgc_vnic_query`. For writing the configuration file, you need information about the EVIC/VEx IOCs present on the fabric, for example, their IOCGUID, IOCSTRING, etc. Use the `ib_qlgc_vnic_query` tool to get this information.

When `ib_qlgc_vnic_query` is executed without any options, it displays detailed information about all the EVIC/VEx IOCs present on the fabric. For example:

```
# ib_qlgc_vnic_query
IO Unit Info:
  port LID:          0003
  port GUID:         fe800000000000000066a0258000001
  change ID:         0009
  max controllers:  0x03

controller[ 1]
  GUID:              00066a0130000001
  vendor ID:         00066a
  device ID:         000030
  IO class :         2000
  ID:                Chassis 0x00066A00010003F2, Slot 1, IOC 1
  service entries:  2
    service[ 0]:     1000066a00000001
                    InfiniNIC.InfiniConSys.Control:01
```

```
service[ 1]: 1000066a00000101
InfiniNIC.InfiniConSys.Data:01
.
.
.
```

NOTE:

A VIO hardware card can contain up to six IOCs (and therefore up to six IOCGUIDs); one for each Ethernet port on the VIO hardware card. Each VIO hardware card contains a unique set of IOCGUIDs: (e.g., IOC 1 maps to Ethernet Port 1, IOC 2 maps to Ethernet Port 2, IOC 3 maps to Ethernet Port 3, etc.).

2. Create the VNIC interfaces using the configuration file `/etc/infiniband/qlgc_vnic.cfg`.

Look at the `qlgcvnictools/qlgc_vnic.cfg` sample to see how VNIC configuration files are written. You can use this configuration file as the basis for creating a configuration file by replacing the Destination Global Identifier (DGID), IOCGUID, and IOCSTRING values with those of the EVIC/VEx IOCs present on your fabric.

QLogic recommends using the DGID of the EVIC/VEx IOC, as it ensures the quickest startup of the VNIC service. When DGID is specified, the IOCGUID must also be specified. For more details, see the `qlgc_vnic.cfg` sample file.

3. Edit the VirtualNIC configuration file, `/etc/infiniband/qlogic_vnic.cfg`. For each IOC connection, add a CREATE block to the file using the following format:

```
{CREATE; NAME="eioc2";
PRIMARY={IOCGUID=0x66A0130000105; INSTANCE=0; PORT=1; }
SECONDARY={IOCGUID=0x66A013000010C; INSTANCE=0; PORT=2;}
# }
```

NOTE:

The `qlgc_vnic.cfg` file is case and format sensitive.

- a. Format 1: Defining an IOC using the IOCGUID. Use the following format to allow the host to connect to a specific VIO hardware card, regardless of which chassis and/or slot the VIO hardware card resides:

```
{CREATE;  
NAME="eiocl";  
IOCGUID=0x66A0137FFFE7;}
```

The following is an example of VIO hardware failover:

```
{CREATE; NAME="eiocl";  
PRIMARY={IOCGUID=0x66a01de000003; INSTANCE=1; PORT=1; }  
SECONDARY={IOCGUID=0x66a02de000003; INSTANCE=1; PORT=1;}
```

NOTE:

Do not create EIOC names with similar character strings (e.g., eioc3 and eioc30). There is a limitation with certain Linux operating systems that cannot recognize the subtle differences. The result is that the user will be unable to ping across the network.

- b. Format 2: Defining an IOC using the IOCSTRING. Defining the IOC using the IOCSTRING allows VIO hardware to be hot-swapped in and out of a specific slot. The host attempts to connect to the specified IOC (1, 2, or 3) on the VIO hardware that currently resides in the specified slot of the specified chassis. Use the following format to allow the host to connect to a VIO hardware that resides in a specific slot of a specific chassis:

```
{CREATE;  
NAME="eiocl";  
IOCSTRING="Chassis 0x00066A0005000001, Slot 1, IOC 1";  
# RX_CSUM=TRUE;  
# HEARTBEAT=100; }
```

NOTE:

The IOCSTRING field is a literal, case-sensitive string. Its syntax must be exactly in the format shown in the previous example, including the placement of commas. To reduce the likelihood to syntax error, use the command `ib_qlgc_vnic_query -es`. Note that the chassis serial number must match the chassis Ox (Hex) value. The slot serial number is specific to the line card as well.

Each CREATE block must specify a unique NAME. The NAME represents the Ethernet interface name that will be registered with the Linux operating system.

- c. Format 3: Starting VNIC using DGID. Following is an example of a DGID and IOCGUID VNIC configuration. This configuration allows for the quickest start up of VNIC service:

```
{CREATE; NAME="eioc1";
DGID=0xfe80000000000000000000000000000066a0258000001;IOCGUID=0x66a0130
000001;
}
```

This example uses DGID, IOCGUID and IOCSTRING:

```
{CREATE; NAME="eioc1";
DGID=0xfe80000000000000000000000000000066a0258000001;
IOCGUID=0x66a01300000001;
IOCSTRING="Chassis 0x00066A00010003F2, Slot 1, IOC 1";
}
```

4. Create VirtualNIC interface configuration files. For each Ethernet interface defined in the `/etc/sysconfig/ics_inic.cfg` file, create an interface configuration file, `/etc/sysconfig/network-scripts/ifcfg-<NAME>` (or `/etc/sysconfig/network/ifcfg-<NAME>` on Linux 2.6 kernels), where **<NAME>** is the value of the NAME field specified in the CREATE block.

Following is an example of an `ifcfg-eiocx` setup for Red Hat systems:

```
DEVICE=eioc1
BOOTPROTO=static
IPADDR=172.26.48.132
BROADCAST=172.26.63.130
NETMASK=255.255.240.0
NETWORK=172.26.48.0
ONBOOT=yes
TYPE=Ethernet
```

Following is an example of an `ifcfg-eiocx` setup for SuSE and SLES systems:

```
BOOTPROTO='static'
IPADDR='172.26.48.130'
BROADCAST='172.26.63.255'
NETMASK='255.255.240.0'
NETWORK='172.26.48.0'
STARTMODE='hotplug'
TYPE='Ethernet'
```

5. Start the QLogic VNIC driver and the QLogic VNIC interfaces. Once you have created a configuration file, you can start the VNIC driver and create the VNIC interfaces specified in the configuration file by running the following command:

```
# /etc/init.d/qlgc_vnic start
```

You can stop the VNIC driver and bring down the VNIC interfaces by running the following command:

```
# /etc/init.d/qlgc_vnic stop
```

To restart the QLogic VNIC driver, run the following command:

```
# /etc/init.d/qlgc_vnic restart
```

If you have not started the InfiniBand network stack (`infinipath` or `OFED`), then running the `/etc/init.d/qlgc_vnic start` command also starts the InfiniBand network stack, since the QLogic VNIC service requires the InfiniBand stack.

If you start the InfiniBand network stack separately, then the correct starting order is:

- Start the InfiniBand stack.
- Start QLogic VNIC service.

For example, if you use `infinipath`, the correct starting order is:

```
# /etc/init.d/infinipath start  
# /etc/init.d/qlgc_vnic start
```

Correct stopping order is:

- Stop QLogic VNIC service.
- Stop the InfiniBand stack.

For example, if you use `infinipath`, the correct stopping order is:

```
# /etc/init.d/qlgc_vnic stop  
# /etc/init.d/infinipath stop
```

If you try to stop the InfiniBand stack when the QLogic VNIC service is running, an error message displays, indicating that some of the modules of the InfiniBand stack are in use by the QLogic VNIC service. Also, any QLogic VNIC interfaces that you created are removed (because stopping the InfiniBand network stack unloads the HCA driver, which is required for the VNIC interfaces to be present).

In this case, do the following:

- Stop the QLogic VNIC service with `/etc/init.d/qlgc_vnic stop`.
- Stop the InfiniBand stack again.

If you want to restart the QLogic VNIC interfaces, run the following command:

```
# /etc/init.d/qlgc_vnic restart
```

You can get information about the QLogic VNIC interfaces by using the following script:

```
# ib_qlgc_vnic_info
```

This information is collected from the `/sys/class/infiniband_qlgc_vnic/interfaces/` directory, under which there is a separate directory corresponding to each VNIC interface.

VNIC interfaces can be deleted by writing the name of the interface to the `/sys/class/infiniband_qlgc_vnic/interfaces/delete_vnic` file. For example, to delete interface `veth0`, run the following command:

```
# echo -n veth0 >
/sys/class/infiniband_qlgc_vnic/interfaces/delete_vnic
```

More information for configuration, starting and stopping the interface, and basic troubleshooting is available in the QLogic *OFED+ User Guide*.

OpenSM

OpenSM is an optional component of the OpenFabrics project that provides a subnet manager for InfiniBand networks. You do not need to use OpenSM if any of your InfiniBand switches provide a subnet manager, or if you are running a host-based SM.

After installing the `opensm` package, OpenSM is configured to be *on* after the next machine reboot. Install OpenSM on the machine that will act as a subnet manager in your cluster, or if it has been installed on more than one machine, use the `chkconfig` command (as root) to disable it on all the nodes *except* for the one on which it will be used. Use this method:

```
# chkconfig opensmd off
```

The command to enable it on reboot is:

```
# chkconfig opensmd on
```

You can start `opensmd` without rebooting your machine by typing:

```
# /etc/init.d/opensmd start
```

You can stop `opensmd` again like this:

```
# /etc/init.d/opensmd stop
```

If you want to pass any arguments to the OpenSM program, modify the following file, and add the arguments to the `OPTIONS` variable:

```
/etc/init.d/opensmd
```

For example:

```
# Use the UPDN algorithm instead of the Min Hop algorithm.  
OPTIONS="-R updn"
```

SRP

SRP stands for SCSI RDMA Protocol. It was originally intended to allow the SCSI protocol to run over InfiniBand for Storage Area Network (SAN) usage. SRP interfaces directly to the Linux file system through the SRP Upper Layer Protocol. SRP storage can be treated as another device.

In this release, two versions of SRP are available: QLogic SRP and OFED SRP. QLogic SRP is available as a separate download, and has its own installation and configuration instructions.

SRP has been tested on targets from Engenio™ (now LSI Logic®) and DataDirect Networks™.

NOTE:

Before using SRP, the SRP targets must already be set up by your system administrator.

Using OFED SRP

To use OFED SRP, follow these steps:

1. Add `ib_srp` to the module list in `/etc/sysconfig/infinipath` to have it automatically loaded.
2. Discover the SRP devices on your fabric by running this command (as root):

```
# ibsrpdm
```

In the output, look for lines similar to these:

```
GUID:          0002c90200402c04  
ID:            LSI Storage Systems SRP Driver 200400a0b8114527  
service entries: 1  
service[ 0]: 200400a0b8114527 / SRP.T10:200400A0B8114527
```

```
GUID:          0002c90200402c0c  
ID:            LSI Storage Systems SRP Driver 200500a0b8114527  
service entries: 1  
service[ 0]: 200500a0b8114527 / SRP.T10:200500A0B8114527
```

```
GUID:          21000001ff040bf6
```

```
ID:          Data Direct Networks SRP Target System
           service entries: 1
service[ 0]: f60b04ff01000021 / SRP.T10:21000001ff040bf6
```

Note that not all the output is shown here; key elements are expected to show the match in [Step 3](#).

3. Choose the device you want to use, and run the command again with the `-c` option (as root):

```
# ibsrpdm -c
id_ext=200400A0B8114527,ioc_guid=0002c90200402c04,dgid=fe8000
00000000000002c90200402c05,pkey=ffff,service_id=200400a0b8114
527
id_ext=200500A0B8114527,ioc_guid=0002c90200402c0c,dgid=fe8000
00000000000002c90200402c0d,pkey=ffff,service_id=200500a0b8114
527
id_ext=21000001ff040bf6,ioc_guid=21000001ff040bf6,dgid=fe8000
000000000021000001ff040bf6,pkey=ffff,service_id=f60b04ff01000
021
```

4. Find the result that corresponds to the target you want, and `echo` it into the `add_target` file:

```
# echo
"id_ext=21000001ff040bf6,ioc_guid=21000001ff040bf6,dgid=fe800
0000000000021000001ff040bf6,pkey=ffff,service_id=f60b04ff0100
0021" > /sys/class/infiniband_srp/srp-ipath0-1/add_target
```

5. You can look for the newly created devices in the `/proc/partitions` file. The file will look similar to this example (the partition names may vary):

```
# cat /proc/partitions
major minor #blocks name
8      64  142325760 sde
8      65  142319834 sde1
8      80   71162880 sdf
8      81   71159917 sdf1
8      96    20480 sdg
8      97    20479 sdg1
```

6. Create a mount point (as root) where you will mount the SRP device. For example:

```
# mkdir /mnt/targetname
# mount /dev/sde1 /mnt/targetname
```

NOTE:

Use `sde1` rather than `sde`. See the `mount(8)` man page for more information on creating mount points.

MPI over uDAPL

Some MPI implementations, such as Intel MPI and HP-MPI, can be run over uDAPL. uDAPL is the user mode version of the Direct Access Provider Library (DAPL). Examples of these types of MPI implementations are Intel MPI and one option on Open MPI.

If you are running this type of MPI implementation, the `rdma_cm` and `rdma_ucm` modules will need to be loaded. To load these modules, use these commands (as root):

```
# modprobe rdma_cm
# modprobe rdma_ucm
```

To ensure that the modules are loaded whenever the driver is loaded, add `rdma_cm` and `rdma_ucm` to the `OPENFABRICS_MODULES` assignment in `/etc/sysconfig/infinipath`.

Other Configuration: Changing the MTU Size

The Maximum Transfer Unit (MTU) is set to 4K and enabled in the driver by default. To change the driver default back to 2K MTU, add this line (as root) in `/etc/modprobe.conf` (or in `/etc/modprobe.conf.local` on SLES):

```
options ib_ipath mtu4096=0
```

Restart the driver as described in [“Starting and Stopping the InfiniPath Software” on page 5-26](#).

NOTE:

The switch must also have the default set to 4K.

Starting and Stopping the InfiniPath Software

The InfiniPath driver software runs as a system service, usually started at system startup. Normally, you will not need to restart the software, but you may want to after installing a new InfiniPath release, after changing driver options, or when you are doing manual testing.

Use the following commands to check or configure the state. These methods will not reboot the system.

To check the configuration state, use the command:

```
$ chkconfig --list infinipath
```

To enable the driver, use the command (as root):

```
# chkconfig infinipath on 2345
```

To disable the driver on the next system boot, use the command (as root):

```
# chkconfig infinipath off
```

NOTE:

This command does not stop and unload the driver if the driver is already loaded.

You can start, stop, or restart (as root) the InfiniPath support with:

```
# /etc/init.d/infinipath [start | stop | restart]
```

This method will not reboot the system. The following set of commands shows how to use this script. Note the following:

- If OpenSM is configured and running, it must be stopped before the `infinipath stop` command, and must be started after the `infinipath start` command. Omit the commands to start/stop `opensmd` if you are not running it on that node.
- Omit the `ifdown` and `ifup` step if you are not using `ipath_ether` on that node.

The sequence of commands to restart `infinipath` are as follows. Note that this example assumes that `ipath_ether` is configured as `eth2`.

```
# /etc/init.d/opensmd stop
# ifdown eth2
# /etc/init.d/infinipath stop
...
# /etc/init.d/infinipath start
# ifup eth2
# /etc/init.d/opensmd start
```

The `...` represents whatever activity you are engaged in after InfiniPath is stopped.

An equivalent way to restart `infinipath` this is to use same sequence as above, except use the `restart` command instead of `start` and `stop`:

```
# /etc/init.d/opensmd stop
# ifdown eth2
# /etc/init.d/infinipath restart
# ifup eth2
# /etc/init.d/opensmd start
```

NOTE:

Stopping or restarting InfiniPath terminates any QLogic MPI processes, as well as any OpenFabrics processes that are running at the time. Processes using networking over `ipath_ether` will return errors.

You can check to see if `opensmd` is running by using the following command; if there is no output, `opensmd` is not configured to run:

```
# /sbin/chkconfig --list opensmd | grep -w on
```

You can check to see if `ipath_ether` is running by using the following command. If it prints no output, it is not running.

```
$ /sbin/lsmmod | grep ipath_ether
```

If there is output, look at the output from this command to determine if it is configured:

```
$ /sbin/ifconfig -a
```

When you need to determine which InfiniPath and OpenFabrics modules are running, use the following command:

```
$ lsmod | egrep 'ipath_|ib_|rdma_|findex'
```

Rebuilding or Reinstalling Drivers After a Kernel Upgrade

If you upgrade the kernel, then you must reboot and then rebuild or reinstall the InfiniPath kernel modules (drivers).

To rebuild the drivers, do the following (as root):

```
# cd /usr/src/infinipath/drivers
# ./make-install.sh
# /etc/init.d/infinipath restart
```

To reinstall the InfiniPath kernel modules, do the following (as root):

```
# rpm -U --replacepkgs infinipath-kernel-*
# /etc/init.d/infinipath restart
```

Rebuilding or Reinstalling Drivers if a Different Kernel is Installed

Installation of the InfiniPath driver RPM (`infinipath-kernel-2.2-xxx-yyy`) builds kernel modules for the currently running kernel version. These InfiniPath modules will work only with that kernel. If a different kernel is booted, then you must reboot and then re-install or rebuild the InfiniPath driver RPM.

Here is an example. These commands must be done as root:

```
# export IPATH_DISTRO=2.6.18_EL5.1 KVER=2.6.18-53.1.14.el5
# cd /usr/src/infinipath/drivers
# ./make-install.sh
# /etc/init.d/infinipath restart
```

Reinstallation instructions for the InfiniPath kernel modules are provided in [“Rebuilding or Reinstalling Drivers After a Kernel Upgrade”](#) on page 5-28.

Further Information on Configuring and Loading Drivers

See the `modprobe(8)`, `modprobe.conf(5)`, and `lsmod(8)` man pages for more information. Also see the file:

```
/usr/share/doc/initscripts-*/sysconfig.txt
```

LED Link and Data Indicators

The LEDs function as link and data indicators once the InfiniPath software has been installed, the driver has been loaded, and the fabric is being actively managed by a subnet manager.

[Table 5-5](#) describes the LED states. The green LED indicates the physical link signal; the amber LED indicates the link. The green LED will normally illuminate first. The normal state is *Green On, Amber On* for adapters other than the QLE7240 and QLE7280, which have an additional state, as shown in [Table 5-5](#).

Table 5-5. LED Link and Data Indicators

LED States	Indication
Green OFF Amber OFF	The switch is not powered up. The software is neither installed nor started. Loss of signal. Verify that the software is installed and configured with <code>ipath_control -i</code> . If correct, check both cable connectors.

Table 5-5. LED Link and Data Indicators (Continued)

LED States	Indication
Green ON Amber OFF	Signal detected and the physical link is up. Ready to talk to SM to bring the link fully up. If this state persists, the SM may be missing or the link may not be configured. Use <code>ipath_control -i</code> to verify the software state. If all HCAs are in this state, then the SM is not running. Check the SM configuration, or install and run <code>opensmd</code> .
Green ON Amber ON	The link is configured, properly connected, and ready. Signal detected. Ready to talk to an SM to bring the link fully up. The link is configured. Properly connected and ready to receive data and link packets.
Green BLINKING (quickly) Amber ON	Indicates traffic
Green BLINKING ^a Amber BLINKING	Locates the adapter This feature is controlled by <code>ipath_control -b [On Off]</code>

Table Notes

^a This feature is available only on the QLE7240 and QLE7280 adapters.

Adapter Settings

The following adapter settings can be adjusted for better performance.

- **Use `taskset` to tune CPU affinity on Opteron systems with the QLE7240, QLE7280, and QLE7140.** Latency will be slightly lower for the Opteron socket that is closest to the PCI Express bridge. On some chipsets, bandwidth may be higher on this socket. See the *QLogic HCA and InfiniPath Software User Guide* for more information on using `taskset`. Also see the `taskset(1)` man page.
- **Use an IB MTU of 4096 bytes instead of 2048 bytes, if available, with the QLE7240, QLE7280, and QLE7140.** 4K MTU is enabled in the InfiniPath driver by default. A switch that supports and is configured for 4KB MTU is also required to use this setting. To change this setting, see [“Other Configuration: Changing the MTU Size” on page 5-26](#).

- **Use a PCIe MaxReadRequest size of at least 512 bytes with the QLE7240 and QLE7280.** QLE7240 and QLE7280 adapters can support sizes from 128 bytes to 4096 byte in powers of two. This value is typically set by the BIOS.
- **Use the largest available PCIe MaxPayload size with the QLE7240 and QLE7280.** The QLE7240 and QLE7280 adapters can support 128, 256, or 512 bytes. This value is typically set by the BIOS as the minimum value supported both by the PCIe card and the PCIe root complex.
- **Adjust the MTRR setting.** Check that the BIOS setting for MTRR is *Discrete*. For more information, see [“MTRR Mapping and Write Combining” on page A-2](#).

Customer Acceptance Utility

`ipath_checkout` is a `bash` script that verifies that the installation is correct and that all the nodes of the network are functioning and mutually connected by the InfiniPath fabric. It must be run on a front end node, and requires specification of a `nodefile`. For example:

```
$ ipath_checkout [options] nodefile
```

The `nodefile` lists the hostnames of the nodes of the cluster, one hostname per line. The format of `nodefile` is as follows:

```
hostname1
hostname2
...
```

NOTE:

The hostnames in the `nodefile` are Ethernet hostnames, not IPv4 addresses.

`ipath_checkout` performs the following seven tests on the cluster:

1. Executes the `ping` command to all nodes to verify that they all are reachable from the front end.
2. Executes the `ssh` command to each node to verify correct configuration of `ssh`.
3. Gathers and analyzes system configuration from the nodes.
4. Gathers and analyzes RPMs installed on the nodes.
5. Verifies QLogic hardware and software status and configuration. Includes tests for link speed, PIO bandwidth (incorrect MTRR settings), and MTU size.

6. Verifies the ability to `mpirun` jobs on the nodes.
7. Runs a bandwidth and latency test on every pair of nodes and analyzes the results.

The options available with `ipath_checkout` are shown in [Table 5-6](#).

Table 5-6. `ipath_checkout` Options

Command	Meaning
<code>-h, --help</code>	These options display help messages describing how a command is used.
<code>-v, --verbose</code> <code>--vv, --vverbose</code> <code>--vvv, --vvvverbose</code>	These options specify three successively higher levels of detail in reporting test results. There are four levels of detail in all, including the case where none of these options are given.
<code>-c, --continue</code>	When this option is not specified, the test terminates when any test fails. When specified, the tests continue after a failure, with failing nodes excluded from subsequent tests.
<code>-k, --keep</code>	This option keeps intermediate files that were created while performing tests and compiling reports. Results will be saved in a directory created by <code>mktemp</code> and named <code>infinipath_XXXXXX</code> or in the directory name given to <code>--workdir</code> .
<code>--workdir=DIR</code>	Use <code>DIR</code> to hold intermediate files created while running tests. <code>DIR</code> must not already exist.
<code>--run=LIST</code>	This option runs only the tests in <code>LIST</code> . See the seven tests listed previously. For example, <code>--run=123</code> will run only tests 1, 2, and 3.
<code>--skip=LIST</code>	This option skips the tests in <code>LIST</code> . See the seven tests listed previously. For example, <code>--skip=2457</code> will skip tests 2, 4, 5, and 7.
<code>-d, --debug</code>	This option turns on the <code>-x</code> and <code>-v</code> flags in <code>bash(1)</code> .

In most cases of failure, the script suggests recommended actions. Please see the `ipath_checkout` man page for more information and updates.

Also refer to the Troubleshooting appendix in the *QLogic HCA and InfiniPath Software User Guide*.

Removing Software Packages

This section provides instructions for uninstalling or downgrading the InfiniPath and OpenFabrics software.

Uninstalling InfiniPath and OpenFabrics RPMs

To uninstall the InfiniPath software packages on any node, type the following command (as root) using a `bash` shell:

```
# rpm -e $(rpm -qa 'InfiniPath-MPI/mpi*' 'InfiniPath/infinipath*')
```

This command uninstalls the InfiniPath and MPI software RPMs on that node.

To uninstall the OpenFabrics software packages on any node, type the following command (as root) using a `bash` shell:

```
# rpm -e rpm_name_pre
```

`rpm_name_pre` is the descriptive name that precedes the version and repository identifiers in an RPM. For example:

```
# rpm -e libibverbs
```

This command uninstalls `libibverbs-2.2-*.rpm` on that node.

For both InfiniPath and OpenFabrics, QLogic recommends that you remove all the packages at the same time.

For a list of OpenFabrics RPMs, see [“OpenFabrics RPMs” on page C-4](#).

Uninstalling OFED 1.3 Software

Use the script `ofed_uninstall.sh` to uninstall the OFED software that was installed from the OFED 1.3 tarball.

Downgrading RPMs

If you want to downgrade, remove both the InfiniPath and OpenFabrics RPMs, then install the older bits. QLogic has determined that `rpm` flags like `--oldpackage` will not generate a correct downgrade.

Additional Installation Instructions

This section contains instructions for additional and alternative software installation.

Installing Lustre

This InfiniPath release supports Lustre. Lustre is a fast, scalable Linux cluster file system that interoperates with InfiniBand. To use Lustre, you need:

- A Linux kernel that is one of the supported kernels for this release, patched with Lustre-specific patches
- Lustre modules compiled for the above kernel
- Lustre utilities required for configuration

The InfiniPath Release Notes provide information about the Lustre patches. For general instructions on downloading, installing, and using Lustre, go to: <http://www.lustre.org>.

Installing QLogic MPI in an Alternate Location

QLogic MPI can be installed in an alternate installation location by using the `--prefix` option with `rpm`. This option is useful for installations that support multiple concurrent MPIs, for other site-specific requirements, or for smooth integration of QLogic MPI with `mpi-selector`.

The `mpi-selector` utility and QLogic MPI `mpi-selector` registration are provided by RPMs in the `OtherMPIs` directory.

When this option is used, the argument passed to `--prefix` replaces the default `/usr` prefix: QLogic MPI binaries, documentation, and libraries are installed under that prefix. However, a few configuration files are installed in `/etc` regardless of the desired `--prefix`.

Additionally, installations that are maintained in alternate locations must ensure that the environment variable `$MPICH_ROOT` is always set to the same prefix that was used to install the RPMs with the `--prefix` option. When set, the `$MPICH_ROOT` variable allows QLogic MPI to correctly locate header and library files for MPI compilation and running parallel jobs.

NOTE:

In InfiniPath 2.2, `$MPICH_ROOT` replaces the environment variable `$INFINIPATH_ROOT`, which is now deprecated. In InfiniPath 2.1, `$INFINIPATH_ROOT` assumed an RPM install `--prefix` of `$INFINIPATH_ROOT/usr`.

For example, install all RPMs that relate to QLogic MPI in `/usr/mpi/qlogic`. Leave all remaining InfiniPath libraries and tools in their default installation location (`/usr`). This approach leaves InfiniPath libraries (such as `libpsm_infinipath.so` and `libinfinipath.so`) in standard system directories so that other MPIs can easily find them in their expected location. Also, this scenario leaves InfiniPath-specific utilities such as `ipath_checkout` and `ipath_control` in standard system search paths.

For this example, unpack the InfiniPath tarball as shown in [“Unpacking the InfiniPath tar File” on page 5-4](#). Then move all RPMs that will be prefixed to a new directory called *InfiniPath-MPI-prefixed*. This includes all QLogic MPI development headers, libraries, runtime and documentation RPMs, as well as `mpi-selector` registration scripts. For example:

```
% mkdir InfiniPath-MPI-prefixed
% mv InfiniPath-MPI/mpi-{frontend,benchmark,libs}*
  InfiniPath-Devel/mpi-devel* \
  OtherMPIs/qlogic-mpi-register* \
  Documentation/mpi-doc* InfiniPath-MPI-prefixed/
```

Next, install all non-prefixed RPMs as explained in [“Using rpm to Install InfiniPath and OpenFabrics” on page 5-8](#):

```
% rpm -Uvh InfiniPath/*.rpm InfiniPath-Devel/*.rpm \
  OpenFabrics/*.rpm OpenFabrics-Devel/*.rpm
```

Finally, install the prefixed InfiniPath-MPI RPMs in `/usr/mpi/qlogic`:

```
% rpm -Uvh --prefix /usr/mpi/qlogic InfiniPath-MPI-prefixed/*.rpm
```

The desired prefix should be made available in the `$MPICH_ROOT` environment variable, either by global shell configuration files or through third-party environment management utilities such as `mpi-selector` or the Environment Modules.

See the Using QLogic MPI section in the *QLogic HCA and InfiniPath Software User Guide* for more information on setting `$MPICH_ROOT` and using the `mpi-selector` utility.

Installing on an Unsupported Distribution

If you are running a kernel that does not match a supported kernel/distribution pair, you may need to provide an override during the install of the `infinipath-kernel` RPM. This override may be needed if you have a completely unsupported distribution, or if you have upgraded the kernel without upgrading the rest of the distribution. The InfiniPath install determines the distribution from either the `/etc/redhat-release` file or the `/etc/SuSE-release` file.

NOTE:

Using the override may not result in a buildable or working driver if your distribution/kernel combination is not similar enough to a tested and supported distribution/kernel pair.

The following example installation is for a Red Hat Enterprise Linux 4 Update 4 compatible kernel, where the `/etc/redhat-release` file indicates another distribution. If you are a `bash` or `sh` user, type:

```
# export IPATH_DISTRO=2.6.9_U4
```

Follow this with your normal `rpm` install commands, or run (as root):

```
# /usr/src/infinipath/drivers/make-install.sh
```

You can examine the list of supported distributions for this InfiniPath release by looking at this script:

```
/usr/src/infinipath/drivers/build-guards.sh
```

This is the current list of strings that can be passed to `IPATH_DISTRO`. The distributions to which each string is applied are shown in brackets:

```
2.6.22_FC6 [Fedora Core 6, 2.6.22 kernel]
2.6.18_EL5 [RHEL5, Scientific Linux 5.0, CentOS 5.0]
2.6.18_EL5.1 [RHEL5.1, Scientific Linux 5.1, CentOS 5.1]
2.6.16_sles10 [SLES10 GM]
2.6.16_sles10_sp1 [SLES10 SP1]
2.6.9_U4 [RHEL4 U4, CentOS 4.4, Scientific Linux 4.4]
2.6.9_U5 [RHEL4 U5, CentOS 4.5, Scientific Linux 4.5]
2.6.9_U6 [RHEL4 U6, CentOS 4.6, Scientific Linux 4.6]
```

If you try to install on an unsupported distribution or an unsupported distribution/kernel pair, you will see an error message. This example shows a case where 4.3 is still in `/etc/redhat-release`, although the kernel is Scientific Linux 4.4:

```
# rpm -Uv infinipath-kernel-2.2-3187.376_rhel4_psc.x86_64.rpm
Preparing packages for installation...
infinipath-kernel-2.2-3187.376_rhel4_psc
*** 2.6.9-42.0.10.ELsmp Scientific Linux SL release 4.3
(Beryllium) is not a supported InfiniPath distribution
error: %post(infinipath-kernel-2.2-3187.376_rhel4_psc.x86_64)
scriptlet failed, exit status 1
```

Managing and Installing Software Using Rocks

Rocks is a distribution designed for managing clusters from the San Diego Supercomputer center (SDSC).

Rocks is a way to manage the *kickstart* automated installation method created by Red Hat. By using the Rocks conventions, the installation process can be automated for clusters of any size. A *Roll* is an extension to the Rocks base distribution that supports different cluster types or provides extra functionality.

QLogic extends the normal Rocks compute node appliance xml file by adding two functions: one function installs the QLogic InfiniPath software, and the other function loads the drivers after kickstart reboots the machine.

This section provides an overview of one way of building a Rocks cluster using the recommend rolls, and a sample xml file that describes the contents of a kickstart config file. By reading this section and following the instructions on the Rocks web site, you can then install the InfiniPath RPMs on the required cluster nodes.

NOTE:

There are many ways to use Rocks to manage clusters. Familiarize yourself first with kickstart and then Rocks before using this method to install the InfiniPath RPMs.

Installing Rocks and InfiniPath RPMs

The following instructions are for building a Rocks 4.2.1 cluster, and then for installing InfiniPath. These instructions are only guidelines; see the material on the Rocks web site to complete an installation. If you want to use later versions of Rocks, these instructions will serve as general procedural steps.

1. Download the required rolls from the Rocks web site:

<http://www.rocksclusters.org/>

Follow the Downloads link to the following CDs:

```
Core Roll: Rocks 4.2.1 X86_64 ISO
(Area51+Base+Ganglia+grid+hpc+java+sge+web-server)
OS Roll - Disk 1
OS Roll - Disk 2
```

You may also need updates; look for the latest files with the `service-pack` prefix.

2. Build the front end node with the above CDs. For more details, see the Rocks installation documentation on the Rocks web site.
3. After building the front end node, make sure that the Rocks tools work. Add users and reload nodes. For more details, see the Rocks installation documentation on the Rocks web site.
4. In the directory `/home/install/site-profiles/4.2.1/nodes`, create the file:

```
extend-compute.xml
```

Use the following contents:

```
<?xml version="1.0" standalone="no"?>

<kickstart>

<description>
  A skeleton XML node file. This file is only a template
  and is intended as an example of how to customize your
  Rocks cluster and use InfiniPath InfiniBand software and
  MPI.

  We want to extend....
</description>

<changelog>
</changelog>

<main>
  <!-- kickstart 'main' commands go here,          -->
  <!--e.g., partitioning info                      -->
</main>
<!--
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! -->
<!-- Many older OFED packages may be included   -->
<!-- with CentOS/RHEL. We don't want to mix these -->
<!-- packages with QLogic packages              -->
<!-- About 26 of them could cause collisions.    -->
<!-- The - in front of the package name causes -->
<!-- to ignore them, so we negate the following -->
<!-- install the QLogic package set in the       -->
<!-- post-97-infinipath script.                 -->
<!-- This list is from CentOS; RedHat packages -->
<!-- slightly different.                        -->
<package>-dapl-devel</package>
<package>-dapl</package>
<package>-kernel-ib</package>
<package>-libibcm</package>
<package>-libibcommon-devel</package>
<package>-libibcommon</package>
<package>-libibmad-devel</package>
<package>-libibmad</package>
<package>-libibumad-devel</package>
<package>-libibumad</package>
<package>-libibverbs-devel</package>
<package>-libibverbs</package>
<package>-libibverbs-utils</package>
```



```

<package>-libipathverbs-devel</package>
<package>-libipathverbs</package>
<package>-libmthca</package>
<package>-librdmacm</package>
<package>-libsdp</package>
<package>-mstflint</package>
<package>-openib-diags</package>
<package>-opensm-devel</package>
<package>-opensm-libs</package>
<package>-opensm</package>
<package>-srptools</package>
<package>-tvflash</package>
<!-- skip lam -->
<package>-lam-gnu</package>

<post>

set -x

IPLOG=/root/InfiniPathextend-bash-compute.log
touch $IPLOG

echo =====InfiniPath-messages===== >> $IPLOG

echo '# INFINIPATH_ULIMIT_INSTALL_COMMENT; used by InfiniPath
install, do not remove' >> /etc/initscript
echo '# This allows locking of up to 128MB of memory per
process. It only' >> /etc/initscript
echo '# takes effect after a reboot, or "init 1", followed by
"init 3"' >> /etc/initscript
echo "ulimit -l 131072" >> /etc/initscript
echo 'eval exec "$4"' >> /etc/initscript

# the kernel must match the smp kernel you give below.
uname -r
export KVER="2.6.9-42.0.2.ELsmp"
/usr/src/infinipath/drivers/make-install.sh >> $IPLOG

chkconfig --level 123456 cpuspeed off
<!--
    Inserting a post installation script here. This
    code will be executed on the destination node AFTER the
    packages have been installed.
-->

<file name="/etc/rc.d/rocksconfig.d/post-97-infinipath"

```

```
mode="create" perms="a+rx">
#!/bin/sh
cd /home/install/contrib/4.2.1/x86_64/RPMS

rpm -Uvh --force infinipath*.rpm `ls mpi*rpm | grep -v
openmpi`

# If and Only IF OpenSM is needed and then please enable
OpenSM
# only on one node.
rpm -Uvh --force opensm-2.2-*_qlc.x86_64.rpm \
  libibcommon-2.2-*_qlc.x86_64.rpm \
  libibmad-2.2-*_qlc.x86_64.rpm \
  libibumad-2.2-*_qlc.x86_64.rpm \
  opensm-devel-2.2-*_qlc.x86_64.rpm \
  opensm-libs-2.2-*_qlc.x86_64.rpm

# Install any other OpenFabrics (OFED) packages that you want,
# then restart
/etc/init.d/infinipath restart
#Either move this file or remove it
mv /etc/rc.d/rocksconfig.d/post-97-infinipath /tmp
#rm /etc/rc.d/rocksconfig.d/post-97-infinipath
</file>

</post>
</kickstart>
```

In this file, the installation of the InfiniPath drivers is done in the `<post>` section, as it is a *live* install. This file can be used as a guideline: it may be cut and pasted, then modified to suit your needs.

5. On the front end node, download the InfiniPath 2.2 tarball for your distribution, RHEL4 or SLES10.

After downloading the tarball, type the following, where `xxx` is the distribution identifier, and `yyy` is the platform architecture:

```
$ tar zxvf InfiniPath2.2-xxx-yyy.tgz
```

The `tar` command will create a directory based on the `tar` file name and place the RPMs there. Next, copy the RPMs you want to install to the directory:

```
/home/install/contrib/4.2.1/x86_64/RPMS
```

It is easiest to copy all of the RPMs to this directory, then use the commands in the `extend-compute.xml` file to install the desired packages.

NOTE:

If you intend to use OpenFabrics and are using RHEL4 or RHEL5, make sure you install

`rhel4-ofed-fixup-2.2-4081.772.rhel4_psc.noarch.rpm`, which is in the `OpenFabrics` directory. This RPM will fix two conflicts. See “OpenFabrics Library Dependencies” on page A-4 for more information.

6. The completion of the installation is done using the `extend-compute.xml` file. More specific instructions for completing the install process can be found under the Documentation link on the Rocks web site:
<http://www.rocksclusters.org/>

Further Information on Rocks and kickstart

QLogic recommends checking the Rocks web site for updates. Extensive documentation on installing Rocks and custom rolls is on the Rocks web site. For more information on Rocks, see:

<http://www.rocksclusters.org/>

To find more information on Red Hat Enterprise Linux 4, and on using kickstart, see:

<http://www.redhat.com/>

Notes

A Installation Troubleshooting

The following sections contain information about issues that may occur during installation. Some of this material is repeated in the Troubleshooting appendix of the *QLogic HCA and InfiniPath Software User Guide*.

Many programs and files are available that gather information about the cluster, and can be helpful for debugging. See appendix D, Useful Programs and Files, in the *QLogic HCA and InfiniPath Software User Guide*.

Hardware Issues

Some of the hardware issues that may occur during installation are described in the following sections. Use the LEDs, as described in [“LED Link and Data Indicators” on page 5-29](#), to help diagnose problems.

Node Spontaneously Reboots

If a node repeatedly and spontaneously reboots when attempting to load the InfiniPath driver, it may be because the QLogic adapter is not installed correctly in the HTX or PCI Express slot.

Some HTX Motherboards May Need Two or More CPUs in Use

Some HTX motherboards may require that two or more CPUs be in use for the QLogic adapter to be recognized. This is most evident in four-socket motherboards.

BIOS Settings

This section covers issues related to BIOS settings. The two most important settings are:

- Advanced Configuration and Power Interface (ACPI). This setting must be enabled. If ACPI is disabled, it may cause initialization problems, as described in the Troubleshooting section of the *QLogic HCA and InfiniPath Software User Guide*.
- Memory Type Range Registers (MTRR) mapping must be set to *Discrete*. Using a different setting may result in reduced performance. See [“MTRR Mapping and Write Combining” on page A-2](#) for details.

MTRR Mapping and Write Combining

MTRR is used by the InfiniPath driver to enable write combining to the QLogic on-chip transmit buffers. Write combining improves write bandwidth to the QLogic chip by writing multiple words in a single bus transaction (typically 64 bytes). Write combining applies only to x86_64 systems. To see if write combining is working correctly and to check the bandwidth, run the following command:

```
$ ipath_pkt_test -B
```

With write combining enabled, the QLE7140 and QLE7240 report in the range of 1150–1500 MBps; the QLE7280 reports in the range of 1950–2960 MBps. The QHT7040/7140 adapters normally report in the range of 2300–2650 MBps.

You can also use `ipath_checkout` (use option 5) to check bandwidth.

In some cases, the InfiniPath driver cannot configure the CPU write combining attributes for QLogic InfiniPath. This case is normally seen with a new system, or after the system's BIOS has been upgraded or reconfigured.

If this error occurs, the interconnect operates, but in a degraded performance mode. The latency typically increases to several microseconds, and the bandwidth may decrease to as little as 200 MBps.

Upon driver startup, you may see these errors:

```
ib_ipath 0000:04:01.0: infinipath0: Performance problem: bandwidth  
to PIO buffers is only 273 MiB/sec  
  
infinipath: mtrr_add(feb00000,0x100000,WC,0) failed (-22)  
infinipath: probe of 0000:04:01.0 failed with error -22
```

If you do not see any of these messages on your console, but suspect this problem, check the `/var/log/messages` file. Some systems suppress driver load messages but still output them to the log file.

Two suggestions for fixing this problem are described in [“Edit BIOS Settings to Fix MTRR Issues”](#) on page A-2 and [“Use the ipath_mtrr Script to Fix MTRR Issues”](#) on page A-3.

See the Troubleshooting section of the *QLogic HCA and InfiniPath Software User Guide* for more details on a related performance issue.

Edit BIOS Settings to Fix MTRR Issues

You can edit the BIOS setting for MTRR Mapping. The BIOS setting looks similar to:

```
MTRR Mapping [Discrete]
```

For systems with very large amounts of memory (32GB or more), it may also be necessary to adjust the BIOS setting for the *PCI hole granularity* to 2GB. This setting allows the memory to be mapped with fewer MTRRs, so that there will be one or more unused MTRRs for the InfiniPath driver.

Some BIOS' do not have the MTRR mapping option. It may have a different name, depending on the chipset, vendor, BIOS, or other factors. For example, it is sometimes referred to as *32 bit memory hole*. This setting must be enabled.

If there is no setting for MTRR mapping or 32 bit memory hole, and you have problems with degraded performance, contact your system or motherboard vendor and ask how to enable write combining.

Use the `ipath_mtrr` Script to Fix MTRR Issues

QLogic also provides a script, `ipath_mtrr`, which sets the MTRR registers, enabling maximum performance from the InfiniPath driver. This Python script is available as a part of the InfiniPath software download, and is contained in the `infinipath*` RPM. It is installed in `/bin`.

To diagnose the machine, run it with no arguments (as root):

```
# ipath_mtrr
```

The test results will list any problems, if they exist, and provide suggestions on what to do.

To fix the MTRR registers, use:

```
# ipath_mtrr -w
```

Restart the driver after fixing the registers.

This script needs to be run after each system reboot.

See the `ipath_mtrr(8)` man page for more information on other options.

Issue with Supermicro® H8DCE-HTe and QHT7040

The QLogic adapter may not be recognized at startup when using the Supermicro H8DCE-HT-e and the QHT7040 adapter. To fix this problem, set the operating system selector option in the BIOS for Linux. The option will look like:

```
OS Installation [Linux]
```

Software Installation Issues

Some problems can be found by running `ipath_checkout`. Run `ipath_checkout` before contacting technical support.

Version Number Conflict with `opensm-*` on RHEL5 Systems

The older `opensm-*` packages that come with the RHEL5 distribution have a version number (3) that is greater than the InfiniPath version number (2.2). This prevents the newer InfiniPath packages from being installed. You may see an error message similar to this when trying to install:

```
Preparing packages for installation...
package opensm-3.1.8-0.1.ofed20080130 (which is newer than
opensm-2.2-33596.832.3_1_10.rhel4_qlc) is already installed
package opensm-devel-3.1.8-0.1.ofed20080130 (which is newer than
opensm-devel-2.2-33596.832.3_1_10.rhel4_qlc) is already installed
```

These older packages should be manually uninstalled before installing the InfiniPath and OpenFabrics 2.2 packages. Run the following command (as root):

```
# rpm -qa opensm\* | xargs rpm -e --nodeps --allmatches
```

If a package was not installed, you may see a warning message similar to:

```
rpm: no packages given for erase
```

OpenFabrics Dependencies

Install `sysfsutils` for your distribution before installing the OpenFabrics RPMs, as there are dependencies. If `sysfsutils` has not been installed, you will see error messages like this:

```
error: Failed dependencies:
libsysfs.so.1() (64bit) is needed by
libipathverbs-2.2-28581.811.1_1.rhel4_psc.x86_64
  libsysfs.so.1() (64bit) is needed by
libibverbs-utils-2.2-28581.811.1_1.rhel4_psc.x86_64
/usr/include/sysfs/libsysfs.h is needed by
libibverbs-devel-2.2-28581.811.1_1.rhel4_psc.x86_64
```

Check your distribution's documentation for information about `sysfsutils`.

OpenFabrics Library Dependencies

There are two issues with OpenFabrics on RHEL4 and RHEL5 distributions. Some of the OpenFabrics RPMs (most notably `ibutils`) have library dependencies that are distribution-specific. Additionally, the OpenFabrics packages contained in this release are newer than the ones that are distributed with RHEL4 U5 and RHEL5, and can cause conflicts during installation. An RPM, `rhel4-ofed-fixup-2.2-xxx.rhel4_psc.noarch.rpm` (where `xxx` is the build identifier), has been added to the RHEL tarball to work around the conflicts problem. It is contained in the OpenFabrics directory, and is installed when the other OpenFabrics RPMs are installed. Installation of this RPM will not affect FC6 users.

Missing Kernel RPM Errors

Install the `kernel-source`, `kernel-devel`, and, if using an older release, `kernel-smp-devel` RPMs for your distribution before installing the InfiniPath RPMs, as there are dependencies. Use `uname -a` to find out which kernel is currently running, to make sure that you install the version with which it matches.

If these RPMs have not been installed, you will see error messages like this when installing InfiniPath:

```
Building and installing InfiniPath modules for 2.6.16_sles10
2.6.16.21-0.8-debug kernel
*** ERROR: /lib/modules/2.6.16.21-0.8-debug/build/.config is
missing.
***          Is the kernel-source rpm for 2.6.16.21-0.8-debug
installed?
=====

=====

Building and installing InfiniPath modules for 2.6.9_U4
2.6.9-42.ELsmp kernel
*** ERROR: /lib/modules/2.6.9-42.ELsmp/build/.config is missing.
***          Is the kernel-smp-devel rpm for 2.6.9-42.ELsmp
installed?
=====

.
.
.
```

Install the correct RPMs by using the `yum` or `yast` commands, for example:

```
# yum install kernel-devel
```

NOTE:

Check your distribution's documentation for more information on installing these RPMs, and for usage of `yum` or `yast`.

Next, the `infinipath-kernel` package must be re-installed, with the `--replacepkgs` option included. Then InfiniPath can be restarted. To do so, type the following (as root):

```
# rpm -U --replacepkgs infinipath-kernel-*
# /etc/init.d/infinipath restart
```

Resolving Conflicts

Occasionally, conflicts may arise when trying to install "on top of" an existing set of files that may come from a different set of RPMs. For example, if you install the QLogic MPI RPMs after having previously installed Local Area Multicomputer (LAM)/MPI, there will be conflicts, since both installations have versions of some of the same programs and documentation. You would see an error message similar to the following:

```
# rpm -Uvh Documentation/*rpm InfiniPath/*rpm
InfiniPath-Devel/*rpm InfiniPath-MPI/*rpm OpenFabrics/*rpm
OpenFabrics-Devel/*rpm OpenSM/*rpm

Preparing...
##### [100%]

file /usr/share/man/man3/MPIO_Request_c2f.3.gz from install of
mpi-doc-2.2-4321.776_rhel4_psc conflicts with file from package
lam-7.1.2-8.fc6
```

Use the following command to remove previously installed conflicting packages. This command will remove all the available LAM packages:

```
# rpm -e --allmatches lam lam-devel lam-libs
```

After the packages have been removed, continue with the InfiniPath installation.

You can also use the `--prefix` option with the `rpm` command to relocate the install directory of any packages that you need to move. See [“Installing QLogic MPI in an Alternate Location” on page 5-34](#) for more information.

`mpirun` Installation Requires 32-bit Support

On a 64-bit system, 32-bit `glibc` must be installed before installing the `mpi-frontend-*` RPM. `mpirun`, which is part of the `mpi-frontend-*` RPM, requires 32-bit support.

If 32-bit `glibc` is not installed on a 64-bit system, an error like this displays when installing `mpi-frontend`:

```
# rpm -Uv ~/tmp/mpi-frontend-2.2-14729.802_rhel4_qlc.i386.rpm
error: Failed dependencies:
/lib/libc.so.6 is needed by mpi-frontend-2.2
-14729.802_rhel4_qlc.i386
```

In older distributions, such as RHEL4, the 32-bit `glibc` is contained in the `libgcc` RPM. The RPM will be named similarly to:

```
libgcc-3.4.3-9.EL4.i386.rpm
```

In newer distributions, `glibc` is an RPM name. The 32-bit `glibc` is named similarly to:

```
glibc-2.3.4-2.i686.rpm OR  
glibc-2.3.4-2.i386.rpm
```

Check your distribution for the exact RPM name.

`ifup` on `ipath_ether` on SLES 10 Reports "unknown device"

SLES 10 does not have all of the QLogic (formerly PathScale) hardware listed in its `pciutils` database. You may see error messages similar to the following after running the `ifup` command:

```
# ifup eth3  
eth3 device: QLogic Corp Unknown device 0010 (rev 02)
```

This has no effect on `ipath_ether`, so this error message can be safely ignored.

Lockable Memory Error on Initial Installation of InfiniPath

During the first installation of InfiniPath software, `/etc/initscript` is created or modified to increase the amount of lockable memory (up to 128 MB) for normal users. This change will not take effect until the system is rebooted, and jobs may fail with error messages about locking memory or failing `mmap`. This error is described in the QLogic MPI Troubleshooting section `Lock Enough Memory on Nodes When Using a Batch Queuing System` in the *QLogic HCA and InfiniPath Software User Guide*.

This is not an issue when upgrading to a newer version of InfiniPath software.

Notes

B Configuration Files

Table B-1 contains descriptions of the configuration and configuration template files used by the InfiniPath and OpenFabrics software.

Table B-1. Configuration Files

Configuration File Name	Description
/etc/infiniband/qlogic_vnic.cfg	VirtualNIC configuration file
/etc/modprobe.conf	Specifies options for modules when added or removed by the <code>modprobe</code> command. Also used for creating aliases. For Red Hat systems
/etc/modprobe.conf.local	Specifies options for modules when added or removed by the <code>modprobe</code> command. Also used for creating aliases. For SLES systems
/etc/sysconfig/hardware/hwcfg-eth-id-\$MAC	Network module file for <code>ipath_ether</code> network interface on SLES systems. \$MAC is replaced by the MAC address.
/etc/sysconfig/ics_inic.cfg	File where VNIC Ethernet interfaces are defined
/etc/sysconfig/infinipath	The primary configuration file for InfiniPath, OFED modules, and other modules and associated daemons. Automatically loads additional modules or changes IPoIB transport type.
/etc/sysconfig/network/ifcfg-<NAME>	Network configuration file for network interfaces. When used for <code>ipath_ether</code> , <NAME> is in the form <code>ethX</code> , where <code>X</code> is the number of the device, typically, 2, 3, etc. If used for VNIC configuration, name is in the form <code>eiocX</code> , where <code>X</code> is the number of the device. There will be one interface configuration file for each interface defined in <code>/etc/sysconfig/ics_inic.cfg</code> . For SLES systems

Table B-1. Configuration Files (Continued)

Configuration File Name	Description
<code>/etc/sysconfig/network-scripts/ifcfg-<NAME></code>	<p>Network configuration file for network interfaces. When used for <code>ipath_ether</code>, <code><NAME></code> is in the form <code>ethX</code>, where <code>X</code> is the number of the device, typically, 2, 3, etc.</p> <p>When used for VNIC configuration, the name is in the form <code>eiocX</code>, where <code>X</code> is the number of the device. There will be one interface configuration file for each interface defined in <code>/etc/sysconfig/ics_inic.cfg</code>.</p> <p>For Red Hat systems</p>
Sample and Template Files	Description
<code>/etc/sysconfig/network/ifcfg.template</code>	Template for the <code>ifcfg-ethX</code> files on SLES systems
<code>qlgcvnictools/qlgc_vnic.cfg</code>	Sample VNIC config file
<code>/usr/share/doc/init-scripts-*/sysconfig.txt</code>	File that explains many of the entries in the configuration files

C RPM Descriptions

The following sections contain detailed descriptions of the RPMs for InfiniPath and OpenFabrics software.

InfiniPath and OpenFabrics RPMs

For ease of installation, QLogic recommends that all RPMs are installed on all nodes. However, some RPMs are optional. Since cluster nodes can be used for different functions, it is possible to selectively install RPMs. For example, you can install the `opensm` package for use on the node that will act as a subnet manager. If you want to selectively install the RPMs, see the following tables for a comparison of required and optional packages.

Different Nodes May Use Different RPMs

In a cluster environment, different nodes may be used for different functions, such as launching jobs, software development, or running jobs. These nodes are defined as follows:

- **Front end node.** This node launches jobs. It is referred to as the *front end node* throughout this document.
- **Compute node.** These nodes run jobs.
- **Development or build node.** These are the machines on which examples or benchmarks can be compiled.

Any machine can serve any combination of these three purposes, but a typical cluster has many compute nodes and just a few (or only one) front end nodes. The number of nodes used for development will vary. Although QLogic recommends installing all RPMs on all nodes, not all InfiniPath software is required on all nodes. See [Table C-1](#), [Table C-2](#), [Table C-3](#), or [Table C-4](#) for information on installation of software RPMs on specific types of nodes.

RPM Version Numbers and Identifiers

The InfiniPath RPMs that are shipped have the InfiniPath release number and the build versions contained in the RPM name. The architecture is designated by `x86_64`, `noarch`, or `i386`, and is dependent upon the distribution. For example:

```
infinipath-2.2-33597.832_rhel14_qlc.x86_64.rpm
```

In this RPM, 2.2 is the InfiniPath release, and 33597.832 are the build versions.

Non-InfiniPath components may also have their own version number:

```
mvapich_gcc-2.2-33597.832.1_0_0.sles10_qlc.x86_64.rpm
```

1_0_0 is the 1.0.0 build for mvapich.

In all of the tables in this appendix, the build identifier is *xxx* and the distribution identifier is *yyy*. Using this convention, the previous RPMs would be listed as:

```
infinipath-2.2-xxx_yyy.x86_64.rpm
```

```
mvapich_gcc-2.2-xxx.1_0_0.yyy.x86_64.rpm
```

RPM Organization

The RPMs are organized as follows:

```
InfiniPath_license.txt, LEGAL.txt (top level)
Documentation/
InfiniPath/
InfiniPath-Devel/
InfiniPath-MPI/
OpenFabrics/
OpenFabrics-Devel/
OpenSM/
OtherHCAs/
OtherMPIs/
```

The tables in the following sections show the sample contents of each of the above directories.

To generate a list of the InfiniPath software package contents on each RPM, type:

```
$ rpm -qlp rpm_file_name
```

Documentation and InfiniPath RPMs

The documentation/RPMs for InfiniPath are listed in [Table C-1](#).

Table C-1. InfiniPath Documentation/RPMs

RPM Name	Front End	Compute	Development
infinipath-doc-2.2-xxx_yyy.noarch.rpm InfiniPath man pages and other documents	Optional	Optional	Optional
mpi-doc-2.2-xxx_yyy.noarch.rpm Man pages for MPI functions and other MPI documents	Optional	Optional	Optional

The InfiniPath/RPMs are listed in [Table C-2](#).

Table C-2. InfiniPath/RPMs

RPM Name	Front End	Compute	Development
<code>infinipath-2.2-xxx_yyy.x86_64.rpm</code> Utilities and source code InfiniPath configuration files Contains <code>ipath_checkout</code> and <code>ipathbug-helper</code> ^a	Optional	Required	Optional
<code>infinipath-kernel-2.2-xxx_yyy.x86_64.rpm</code> InfiniPath drivers, OpenFabrics kernel modules	Optional	Required	Optional
<code>infinipath-libs-2.2-xxx_yyy.i386.rpm</code> InfiniPath protocol shared libraries for 32-bit and 64-bit systems	Optional	Required	Optional

Table Notes

^a If you want to use `ipath_checkout` and `ipathbug-helper`, install this RPM wherever you install `mpi-frontend`.

The InfiniPath-Devel/RPMs are listed in [Table C-3](#).

Table C-3. InfiniPath-Devel/RPMs

RPM Name	Front End	Compute	Development
<code>infinipath-devel-2.2-xxx_yyy.x86_64.rpm</code> Development files for InfiniPath	Optional	Optional	Optional
<code>mpi-devel-2.2-xxx_yyy.noarch.rpm</code> Source code for the MPI development environment, including headers and libs, MPI examples and benchmarks. Use to build the examples or rebuild the benchmarks.	Optional	Optional	Required

The InfiniPath-MPI/RPMs are listed in [Table C-4](#).

Table C-4. InfiniPath-MPI/RPMs

RPM Name	Front End	Compute	Development
mpi-benchmark-2.2-xxx_yyy.x86_64.rpm MPI benchmark binaries	Optional	Required	Optional
mpi-frontend-2.2-xxx_yyy.i386.rpm MPI job launch scripts and binaries, including mpirun and MPD	Required	Required	Optional
mpi-libs-2.2-xxx_yyy.i386.rpm Shared libraries for MPI	Optional	Required	Required

OpenFabrics RPMs

OpenFabrics and OpenSM are optional components. For ease of installation, QLogic recommends that all of the OpenFabrics RPMs listed in [Table C-5](#) and [Table C-6](#) be installed on all nodes. The development RPMs in [Table C-7](#) are only needed on the nodes where OFED programs are compiled. The `opensm` package in [Table C-8](#) should be installed only on the node that will be used as a subnet manager. The packages in [Table C-9](#) should be installed only if other HCAs are used. The packages in [Table C-11](#) should be installed if other MPI implementations are desired.

Table C-5. OpenFabrics Documentation/RPMs

RPM Name	Comments
ofed-docs-2.2-xxx.1_3.yyy.x86_64.rpm OpenFabrics documentation	Optional for OpenFabrics

Table Notes

OpenFabrics documentation is installed in the same directory as the InfiniPath documentation.

Table C-6. OpenFabrics/RPMs

RPM Name	Comments
dapl-2.2-xxx.1_2_5.yyy.x86_64.rpm uDAPL 1.2 support	Optional for OpenFabrics
dapl-2.2-xxx.2_0_7.yyy.x86_64.rpm uDAPL 2.0 support	Optional for OpenFabrics

Table C-6. OpenFabrics/RPMs (Continued)

RPM Name	Comments
dapl-utils-2.2-xxx.2_0_7.yyy.x86_64.rpm uDAPL support	Optional for OpenFabrics
ib-bonding-2.2-xxx.0_9_0.yyy.x86_64.rpm Utilities to manage and control the driver operation	Optional for OpenFabrics
ibsim-2.2-xxx.0_4.yyy.x86_64.rpm Voltaire InfiniBand Fabric Simulator	Optional for OpenFabrics
ibutils-2.2-xxx.1_2.yyy.x86_64.rpm ibutils provides InfiniBand (IB) network and path diagnostics.	Optional for OpenFabrics
ibvexdmtools-2.2-xxx.0_0_1.yyy.x86_64.rpm Discover and use QLogic Virtual NIC devices via VNIC protocol over InfiniBand	Optional for OpenFabrics
infiniband-diags-2.2-xxx.1_3_6.yyy.x86_64.rpm Diagnostic tools	Optional for OpenFabrics
iscsi-initiator-utils-2.2-xxx.2_0.yyy.x86_64.rpm ^a Server daemon and utility programs for iSCSI. Also iSER support For Red Hat systems	Optional for OpenFabrics
libibcm-2.2-xxx.1_0_2.yyy.x86_64.rpm Along with the OpenFabrics kernel drivers, libibcm provides a user-space IB connection management API.	Optional for OpenFabrics
libibcommon-2.2-xxx.1_0_8.yyy.x86_64.rpm Common utility functions for IB diagnostic and management tools	Required for OpenSM
libibmad-2.2-xxx.1_1_6.yyy.x86_64.rpm Low-layer IB functions for use by the IB diagnostic and management programs. These include management datagrams (MADs), SA, SMP, and other basic IB functions.	Required for OpenSM
libibumad-2.2-xxx.1_1_7.yyy.x86_64.rpm Provides the user MAD library functions that sit on top of the user MAD modules in the kernel. These functions are used by IB diagnostic and management tools, including OpenSM.	Required for OpenSM

Table C-6. OpenFabrics/RPMs (Continued)

RPM Name	Comments
<p><code>libibverbs-2.2-xxx.1_1_1.yyy.x86_64.rpm</code> Library that allows userspace processes to use InfiniBand <i>verbs</i> as described in the <i>InfiniBand Architecture Specification</i>. This library includes direct hardware access for fast path operations. For this library to be useful, a device-specific plug-in module must also be installed.</p>	Required for OpenFabrics
<p><code>libibverbs-utils-2.2-xxx.1_1_1.yyy.x86_64.rpm</code> Useful <code>libibverbs</code> example programs such as <code>ibv_devinfo</code>, which displays information about IB devices</p>	Required for OpenFabrics
<p><code>libipathverbs-2.2-xxx.1_1.yyy.x86_64.rpm</code> Provides device-specific userspace driver for QLogic HCAs</p>	Required for OpenFabrics
<p><code>librdmacm-2.2-xxx.1_0_6.yyy.x86_64.rpm</code> Support for the new connection manager</p>	Optional for OpenFabrics
<p><code>librdmacm-utils-2.2-xxx.1_0_6.yyy.x86_64.rpm</code> Utilities for the new connection manager</p>	Optional for OpenFabrics
<p><code>libsdp-2.2-xxx.1_1_99.yyy.x86_64.rpm</code> Can be LD_PRELOAD-ed to have a sockets application use IB Sockets Direct Protocol (SDP) instead of TCP, transparently and without recompiling the application</p>	Required for OpenFabrics
<p><code>ofed-scripts-2.2-xxx.1_3.yyy.x86_64.rpm</code> OpenFabrics scripts</p>	Optional for OpenFabrics
<p><code>openib-diags-2.2-xxx.1_2_7.yyy.x86_64.rpm</code> Useful programs for troubleshooting and checking the state of the adapter, IB fabric, and its components</p>	Optional for OpenFabrics
<p><code>open-iscsi-2.2-xxx.2_0.yyy.x86_64.rpm</code>^a Transport independent, multi-platform implementation of RFC3720 iSCSI with iSER support For SLES systems</p>	Optional for OpenFabrics
<p><code>opensm-libs-2.2-xxx.3_1_10.yyy.x86_64.rpm</code> Provides the library for OpenSM</p>	Required for OpenSM
<p><code>perftest-2.2-xxx.1_2.yyy.x86_64.rpm</code> IB performance tests</p>	Optional for OpenFabrics

Table C-6. OpenFabrics/RPMs (Continued)

RPM Name	Comments
qlgc_vnic_daemon-2.2-xxx.0_0_1.yyy.x86_64.rpm Used with VNIC ULP service	Optional for OpenFabrics
qlvnictools-2.2-xxx.0_0_1.yyy.x86_64.rpm Startup script, sample config file, and utilities	Optional for OpenFabrics
qperf-2.2-xxx.0_4_0.yyy.x86_64.rpm IB performance tests	Optional for OpenFabrics
rds-tools-2.2-xxx.1_1.yyy.x86_64.rpm Supports RDS	Optional for OpenFabrics
rhel4-ofed-fixup-2.2-xxx.2_2.yyy.noarch.rpm Fixes conflicts with older versions of OpenFabrics for RHEL4 and RHEL5	Required for OpenFabrics
sdpnetstat-2.2-xxx.1_60.yyy.x86_64.rpm Provides network statistics for SDP	Optional for OpenFabrics
srptools-2.2-xxx.0_0_4.yyy.x86_64.rpm Support for SRP	Optional for OpenFabrics

Table Notes

There are two versions of the `dapl*` packages: version `1_2_5` and version `2_0_7`. QLogic recommends installing the `1_2_5` version for compatibility with most `dapl` applications.

^a `iscsi-initiator-utils-2.2-xxx.2_0.yyy.x86_64.rpm` and `open-iscsi-2.2-xxx.yyy.x86_64.rpm` are essentially the same, except that the former is for Red Hat and the latter is for SLES.

Table C-7. OpenFabrics-Devel/RPMs

RPM Name	Comments
dapl-devel-2.2-xxx.2_0_7.yyy.x86_64.rpm Development files for uDAPL support	Optional for OpenFabrics
ibsim-2.2-xxx.0_4.yyy.x86_64.rpm InfiniBand fabric simulator from Voltaire	Optional for OpenFabrics
libibcm-devel-2.2-xxx.1_0_2.yyy.x86_64.rpm Development files for the <code>libibcm</code> library	Optional for OpenFabric

Table C-7. OpenFabrics-Devel/RPMs (Continued)

RPM Name	Comments
libibcommon-devel-2.2-xxx.1_0_8.yyy.x86_64.rpm Development files for the libibcommon library	Optional for OpenFabrics
libibmad-devel-2.2-xxx.1_1_6.yyy.x86_64.rpm Development files for the libibmad library	Optional for OpenFabrics
libibumad-devel-2.2-xxx.1_1_7.yyy.x86_64.rpm Development files for the libibumad library	Optional for OpenFabrics
libibverbs-devel-2.2-xxx.1_1_1.yyy.x86_64.rpm Libraries and header files for the libibverbs verbs library	Optional for OpenFabrics
libipathverbs-devel-2.2-xxx.1_1.yyy.x86_64.rpm Libraries and header files for the libibverbs verbs library	Optional for OpenFabrics
librdmacm-devel-2.1-xxx.1_0_6.yyy.x86_64.rpm Development files for the new connection manager	Optional for OpenFabrics
libsdp-devel-2.2-xxx.1_1_99.yyy.x86_64.rpm Can be LD_PRELOAD-ed to have a sockets application use Sockets Direct Protocol (SDP) instead of TCP	Optional for OpenFabrics
opensm-devel-2.2-xxx.3_1_10.yyy.x86_64.rpm Development files for OpenSM	Optional for OpenSM

Table C-8. OpenSM/RPMs

RPM Name	Comments
opensm-2.2-xxx.3_1_10.yyy.x86_64.rpm OpenSM provides an implementation of an InfiniBand subnet manager and administrator. At least one per each InfiniBand subnet is required to initialize the InfiniBand hardware.	Required for OpenSM

Table C-9. Other HCAs/RPMs

RPM Name	Comments
libcxgb3-2.2-xxx.1_1_3.yyy.x86_64.rpm Support for the Chelsio 10GbE HCA	Optional for OpenFabrics

Table C-9. Other HCAs/RPMs (Continued)

RPM Name	Comments
libmlx4-2.2-xxx.1_0.yyy.x86_64.rpm Userspace driver for Mellanox® ConnectX™ InfiniBand HCAs	Optional for OpenFabrics
libmthca-2.2-xxx.1_0_4.yyy.x86_64.rpm Provides a device-specific userspace driver for Mellanox HCAs for use with the libibverbs library	Optional for OpenFabrics
libnes-2.2-xxx.0_5.yyy.x86_64.rpm Provides a userspace driver for NetEffect RNICs for use with the libibverbs library	Optional for OpenFabrics
mstflint-2.2-xxx.1_3.yyy.x86_64.rpm Firmware update tool for other HCAs	Optional for OpenFabrics
tvflash-2.2-xxx.0_9_0.yyy.x86_64.rpm Query and update the firmware flash memory for other HCAs	Optional for OpenFabrics

Table C-10. Other HCAs-Devel/RPMs

RPM Name	Comments
libcxgb3-devel-2.2-xxx.1_1_3.yyy.x86_64.rpm Development files for the Chelsio 10GbE HCA	Optional for OpenFabrics

Table C-11. Other MPIs/RPMs

RPM Name	Front End	Compu te	Developme nt
mpi-selector-2.2-xxx.1_0_0.yyy.x86_64.rpm Tool to select MPI compiled with different compilers	Optional	Optional	Optional
mpitests_mvapich_gcc-2.2-xxx.3_0.yy.x86_64.rpm MVAPICH MPI tests compiled with GNU	Optional	Optional	Optional
mpitests_mvapich_intel-2.2-xxx.3_0.yyy.x86_64.rpm ^a MVAPICH MPI tests compiled with Intel	Optional	Optional	Optional

Table C-11. OtherMPIs/RPMs (Continued)

RPM Name	Front End	Compu te	Developme nt
mpitests_mvapich_pathscale-2.2-xxx .3_0.yyy.x86_64.rpm MVAPICH MPI tests compiled with PathScale	Optional	Optional	Optional
mpitests_mvapich_pgi-2.2-xxx.3_0.y yy.x86_64.rpm MVAPICH MPI tests compiled with PGI	Optional	Optional	Optional
mpitests_openmpi_gcc-2.2-xxx.3.0.y yy.x86_64.rpm Open MPI MPI tests compiled with GNU	Optional	Optional	Optional
mpitests_openmpi_intel-2.2-xxx.3_0 .yyy.x86_64.rpm ^a Open MPI MPI tests compiled with Intel	Optional	Optional	Optional
mpitests_openmpi_pathscale-2.2-xxx .3_0.yyy.x86_64.rpm Open MPI MPI tests compiled with PathScale	Optional	Optional	Optional
mpitests_openmpi_pgi-2.2-xxx.3_0.y yy.x86_64.rpm Open MPI MPI tests compiled with PGI	Optional	Optional	Optional
mvapich_gcc-2.2-xxx.1_0_0.yyy.x86_ 64.rpm MVAPICH compiled with GNU	Optional	Optional	Optional
mvapich_intel-2.2-xxx.1_0_0.yyy.x8 6_64.rpm ^a MVAPICH compiled with Intel	Optional	Optional	Optional
mvapich_pathscale-2.2-xxx.1_0_0.yy y.x86_64.rpm MVAPICH compiled with PathScale	Optional	Optional	Optional
mvapich_pgi-2.2-xxx.1_0_0.yyy.x86_ 64.rpm MVAPICH compiled with PGI	Optional	Optional	Optional
openmpi_gcc-2.2-xxx.1_2_5.yyy.x86_ 64.rpm Open MPI compiled with GNU	Optional	Optional	Optional

Table C-11. OtherMPIs/RPMs (Continued)

RPM Name	Front End	Compute	Development
openmpi_intel-2.2-xxx.1_2_5.yyy.x86_64.rpm ^a Open MPI compiled with Intel	Optional	Optional	Optional
openmpi_pathscale-2.2-xxx.1_2_5.yyy.x86_64.rpm Open MPI compiled with PathScale	Optional	Optional	Optional
openmpi_pgi-2.2-xxx.1_2_5.yyy.x86_64.rpm Open MPI compiled with PGI	Optional	Optional	Optional
qlogic-mpi-registry-2.2-xxx.1_0.yyy.x86_64.rpm ^b Helps QLogic MPI interoperate with other MPIs through the <code>mpi-selector</code> utility	Optional	Optional	Optional

Table Notes

The compiler versions used are: GNU 4.1., PathScale 3.0, Intel 10.1.012, and PGI 7.1-5.

^a The Intel-compiled version requires that the Intel compiler be installed and that paths to the Intel compiler runtime libraries be resolvable from the user's environment. The version used is Intel 10.1.012.

^b Install the `mpi-devel-*` RPM before installing this RPM, as there are dependencies

Notes

Index

A

ACPI 4-4, A-1
Adapter, *see* HCA

B

BIOS
 configuring 4-4
 settings A-1
 settings to fix MTRR issues A-2

C

-c 5-32
Cables supported 4-3
Compiler support 2-4
Configuration
 files B-1
 ib_ipath 5-12
 ipath_ether on Fedora, RHEL4, RHEL5
 5-12
 ipath_ether on SLES 5-14
 OpenSM 5-23
 VNIC 5-18
--continue 5-32
CPUs, HTX motherboards may require two or
 more CPUs A-1

D

-d 5-32
--debug 5-32
Distribution override, setting 5-35
Distributions supported 2-3

Document conventions 1-3
Documentation for InfiniPath 1-4
Documentation/RPMs C-2, C-4
Downloading software files 5-3
Drivers
 configuration and loading information 5-29
 InfiniPath and OpenFabrics overview 5-11
 IPoIB configuration 5-16, 5-17
 rebuilding or reinstalling after a kernel
 upgrade 5-28
 rebuilding or reinstalling if a different kernel
 is installed 5-29
 starting, stopping, and restarting 5-26

E

Error
 /lib/modules/2.6.16.21-0.8-debug
 /build/.config is missing error
 message A-5
 Failed dependencies error message
 A-4, A-6

F

Features, new 2-2
file/usr/share/man/man3/MPIO_Req
 st_c2f.3.gz from install of
 mpi-doc-2.1-4321.776_rhel4_p
 sc conflicts with file from
 package lam-7.1.2-8.fc6 error
 message A-6
Files, missing 5-5
Form factors for HCAs 4-2

H

- h 5-32
- Hardware installation overview 3-1
- Hardware requirements 4-1
- HCA
 - form factors 4-2
 - other HCAs-Devel/RPMs C-9
 - model numbers 2-1
 - not recognized A-1
 - other HCAs/RPMs C-8
 - QHT7140 installation with HTX riser 4-12
 - QHT7140 installation without an HTX riser 4-16
 - QLE7140/7240/7280 installation with PCI Express riser 4-9
 - QLE7140/7240/7280 installation without a PCI Express riser 4-15
 - supported 1-1
 - tuning for performance 5-30
- help 5-32
- HTX motherboards may required two or more CPUs A-1

I

- ib_ipath
 - module 5-11
 - 0000:04:01.0:infinipath0:Performance problem: error message A-2
 - configuration 5-12
- IBA6110 4-8
- IBA6120 4-7
- IBA7220 4-7
- ifup on iaph_ether reports unknown device on SLES 10 A-7
- InfiniPath
 - see *also* InfiniPath software
 - cluster 1-2
 - devel/RPMs C-3
 - documentation 1-4
 - documentation and InfiniPath RPMs C-2
 - downgrading RPMs 5-33
 - environment 5-2

- interconnect overview 1-2
- MPI/RPMs C-4
- OpenFabrics interoperability 1-3
- restarting the service 5-10
- RPMs C-3
- RPMs, installing 5-6
- scripts, using to start, stop, or restart drivers 5-26
- starting the service 5-10
- tar file, unpacking 5-4
- uninstalling RPMs 5-33
- using rpm to install 5-8
- InfiniPath software
 - installed layout 5-9
 - list of 2-4
 - starting, stopping, and restarting 5-26
- Installation
 - hardware overview 3-1
 - hardware requirements for 4-1–4-2
 - Lustre 5-34
 - on an unsupported distribution 5-35
 - QHT7140 with HTX riser 4-12
 - QHT7140 without an HTX riser 4-16
 - QLE7140/7240/7280 with PCI Express riser 4-9
 - QLE7140/7240/7280 without a PCI Express riser 4-15
 - QLogic MPI in an alternate location 5-34
 - using Rocks for 5-36–5-41
 - software overview 3-2
 - verification of 5-31
- Interoperability, InfiniPath OpenFabrics 1-3
- ipath
 - checkout 5-31, A-3
 - checkout options 5-32
 - ether, configuration on Red Hat and Fedora 5-12
 - ether, configuration on SLES 5-14
 - ether, configuring the network interface for 5-12
 - mtrr script to fix MTRR issues A-3
- IPoIB driver configuration 5-16, 5-17

K

- k 5-32
- keep 5-32
- Kernel
 - supported 2-3
- Kernel, missing kernel RPM errors A-5
- Kernels supported 2-3

L

- LEDs, blink patterns 5-29
- Linux, supported distributions 5-2
- Lockable memory error A-7
- Lustre, installing 5-34

M

- Model numbers for HCAs 2-1
- MPI, other MPIs/RPMs C-9
- MPI over uDAPL 5-26
- mpirun, installation requires 32-bit support A-6
- mpi-selector 5-34
- MTRR 4-4
 - editing BIOS settings to fix A-2
 - mapping and write combining A-2
 - using `ipath_mtrr` script to fix issues A-3
- MTU, changing the size 5-26

N

- Node repeatedly reboots error A-1
- Node types 5-1

O

- OFED software, uninstalling 5-33
- OFED SRP 5-24
- OpenFabrics
 - devel/RPMs C-4, C-7

- configuration 5-16
- documentation/RPMs C-4
- downgrading RPMs 5-33
- environment 5-2
- interoperability with InfiniPath 1-3
- library dependencies with RHEL4 and RHEL5 A-4
- RPMs C-4
- RPMs, installing 5-6
- RPMs, uninstalling 5-33
- sysfsutils dependencies A-4
 - using `rpm` to install 5-8
- OpenSM 5-23
- OpenSM/RPMs C-8
- `opensm-*` packages, version number conflict on RHEL5 A-4
- `openssh` 5-3
- `openssh-server` 5-3
- Optical media converters 4-4

P

- Package contents 4-5
- package `opensm-3.1.8...` <which is newer than `opensm-2.2-33596.832.3...`> is already installed error message A-4
- Performance tuning 5-30
- Protocols supported 2-5
- Protocols, InfiniBand subnet management 1-3

Q

- QHT7040 4-8
- QHT7140 4-8
 - installation with HTX riser 4-12
 - installation without an HTX riser 4-16
- QLE7140 4-7
- QLE7140/7240/7280
 - installation with PCI Express riser 4-9
 - installation without a PCI Express riser 4-15

QLE7280 4-7

QLogic MPI, installing in an alternate location
5-34

R

RD 2-5

RHEL4 and RHEL5, OpenFabrics library
dependencies A-4

RHEL5 `opensm-*` packages, version number
conflict A-4

Rocks, managing and installing software with
5-36

RPMs

choosing which to install 5-7

different sets on the same system A-6

documentation C-2

downgrading 5-33

other HCAs C-8

other HCAs-Devel C-9

InfiniPath C-3

InfiniPath-devel C-3

InfiniPath-MPI C-4

installing InfiniPath and OpenFabrics 5-6

list of InfiniPath C-2

missing 5-5

missing kernel RPM errors A-5

other MPIs C-9

no packages given for erase
warning message A-4

OpenFabric C-4

OpenFabrics-devel C-4, C-7

OpenSM C-8

using to install InfiniPath and OpenFabrics
5-8

version numbers and identifiers C-1

`--run=LIST` 5-32

S

Safety precautions 4-5

`--skip=LIST` 5-32

SLES 10, `ifup` on `ipath_ether` reports
unknown device A-7

SLES, `ipath_ether` configuration 5-14

Software

files to download 5-3

InfiniPath 2-4

InfiniPath layout 5-9

installation overview 3-2

package requirements 5-3

SRP 5-24

SRP, OFED SRP 5-24

Subnet management 1-3

Supermicro H8DCE-HTe, problems with
QHT7040 A-3

Support, technical 1-5

Switch configuration and monitoring 4-17

Switches supported 1-2, 4-3

T

`tar` file, unpacking 5-4

TCP/IP, enabling networking over the
InfiniPath link 5-12

Technical support 1-5

Terminology 1-3

Transport services supported 2-5

U

uDAPL, with MPI 5-26

Uninstalling InfiniPath and OpenFabrics RPMs
5-33

Uninstalling OFED software 5-33

V

-v 5-32

--verbose 5-32

VNIC, configuration 5-18

-vv 5-32

--vverbose 5-32

-vvv 5-32

--vvverbose 5-32

W

--workdir=DIR 5-32

Notes



Corporate Headquarters QLogic Corporation 26650 Aliso Viejo Parkway Aliso Viejo, CA 92656 949.389.6000 www.qlogic.com
Europe Headquarters QLogic (UK) LTD. Quatro House Lyon Way, Frimley Camberley Surrey, GU16 7ER UK +44 (0) 1276 804 670

© 2006–2008 QLogic Corporation. Specifications are subject to change without notice. All rights reserved worldwide. QLA, QLogic, SANsurfer, the QLogic logo, InfiniPath, SilverStorm, and EKOPath are trademarks or registered trademarks of QLogic Corporation. AMD Opteron is a trademark of Advanced Microdevices Inc. BladeCenter and IBM are registered trademarks of International Business Machines Corporation. DataDirect Networks is a trademark of DataDirect Networks, Inc. EMCORE is a trademark of EMCORE Corporation. HTX is a trademark of the HyperTransport Technology Consortium. IBM and BladeCenter are registered trademarks of International Business Machines Corporation. InfiniBand is a trademark and service mark of the InfiniBand Trade Association. Intel is a registered trademark of Intel Corporation. Linux is a registered trademark of Linus Torvalds. LSI Logic and Engenio are trademarks or registered trademarks of LSI Logic Corporation. Lustre is a registered trademark of Cluster File Systems, Inc. Mellanox is a registered trademark and ConnectX is a trademark of Mellanox Technologies, Inc. PathScale is a trademark of PathScale LLC. PCI Express is a registered trademark of PCI-SIG Corporation. Red Hat and Enterprise Linux are registered trademarks of Red Hat, Inc. Supermicro is a registered trademark of Super Micro Computer Inc. SUSE is a registered trademark of Novell Inc. Zarlink is a trademark of Zarlink Semiconductor Inc. All other brand and product names are trademarks or registered trademarks of their respective owners. Information supplied by QLogic Corporation is believed to be accurate and reliable. QLogic Corporation assumes no responsibility for any errors in this brochure. QLogic Corporation reserves the right, without notice, to make changes in product design or specifications.

Free Manuals Download Website

<http://myh66.com>

<http://usermanuals.us>

<http://www.somanuals.com>

<http://www.4manuals.cc>

<http://www.manual-lib.com>

<http://www.404manual.com>

<http://www.luxmanual.com>

<http://aubethermostatmanual.com>

Golf course search by state

<http://golfingnear.com>

Email search by domain

<http://emailbydomain.com>

Auto manuals search

<http://auto.somanuals.com>

TV manuals search

<http://tv.somanuals.com>