



Intel® Omni-Path Fabric Suite FastFabric

User Guide

Rev. 8.0

October 2017



You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or visit <http://www.intel.com/design/literature.htm>.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at intel.com.

No computer system can be absolutely secure.

Intel, the Intel logo, Intel Xeon Phi, and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2015–2017, Intel Corporation. All rights reserved.



Revision History

For the latest documentation, go to <http://www.intel.com/omnipath/FabricSoftwarePublications>.

Date	Revision	Description
October 2017	8.0	<p>Updates to this document include:</p> <ul style="list-style-type: none"> This document has been restructured to include the <i>Intel® Omni-Path Fabric Suite FastFabric Command Line Interface Reference Guide</i> content. All references have been updated appropriately. Moved non-FastFabric CLI tools to the <i>Intel® Omni-Path Fabric Host Software User Guide</i>: opa-arptbl-tuneup, opaautoconfig, opa-init-kernel, opa_osd_dump, opa_osd_exercise, opa_osd_perf, opa_osd_query, opacapture, opacconfig, opafabricinfo, opagetvf, opagetvf_env, opahfirev, opainfo, opapacketcapture, opapmaquery, opaportconfig, opaportinfo, oparesolvehfiport, opasaquery, opasmaquery, and opatmmtool Updated the following CLI commands: <ul style="list-style-type: none"> opaallanalysis opacmdall opafequery (deprecated) opashowallports Updated the following port counters: <ul style="list-style-type: none"> Link Quality Indicator (LQI) LocalLinkIntegrityErrors (LLI) Counter PortRcvErrors (RxE) Counter ExcessiveBufferOverrunErrors (EBO) Counter UncorrectableErrors (Unc) Counter FMConfigErrors Counter (FMC)
August 2017	7.0	<p>Updates to this document include:</p> <ul style="list-style-type: none"> Updated Performing Single Host Verification with two new prompts. Updated "failed ports" and "failed nodes" to "no response ports" and "no response nodes", respectively, in text and screenshots in the following sections: <ul style="list-style-type: none"> Accessing the Fabric Performance Monitor Fabric Performance Monitor TUI Overview How to Use the Fabric Performance Monitor TUI Viewing the Fabric Performance Monitoring Summary Screen Viewing Image Information Viewing Bandwidth Utilization Bookmarking a Sweep Fabric Configuration and PM Image Information PM Port Group's Performance Utilization and Statistical Data Sorted Lists of Links based upon Statistical Criteria
April 2017	6.0	<p>Updates to this document include:</p> <ul style="list-style-type: none"> Global filepath changes: <ul style="list-style-type: none"> From /usr/lib/opa/samples/ to /usr/share/opa/samples/ From /usr/lib/opa/src/ to /usr/src/opa/ From /etc/sysconfig/ to /etc/
continued...		



Date	Revision	Description
		<ul style="list-style-type: none">Added new section Intel® Omni-Path Architecture Overview.Updated Focused Fabric Feature Analysis.Deprecated <code>opaxlattopology_cust</code> in Topology Analysis and Verifying OPA Fabric Status and TopologyChanged How to Use the Fabric Performance Monitor TUI and sections within Monitoring Fabric Performance:<ul style="list-style-type: none">"Group BW Summary" to "Group Performance (P)""Group Err Summary" to "Group Statistics (S)""Bandwidth Statistics" to "Bandwidth Utilization""Error Statistics" to "Statistics Category"Added new section Using Fabric Performance Data.Added new chapter FastFabric Troubleshooting including new section Switching to the Intel P-State Driver to Run Certain FastFabric Tools.Added new chapter FastFabric Diagnostics Capabilities.
December 2016	5.0	Updates to this document include: <ul style="list-style-type: none">Document has been restructured and rewritten for usability.Added Cluster Configurator for Intel® Omni-Path Fabric to Preface.Globally, updated filepath from <code>/opt/opa</code> to <code>/usr/lib/opa</code>.Added Intel® Omni-Path Software Overview to Overview.Added note to Starting Up the Tools that you must have root privilege to run FastFabric commands.
August 2016	4.0	Document has been updated.
May 2016	3.0	Document has been updated.
February 2016	2.0	Document has been updated.
November 2015	1.0	Document has been updated.

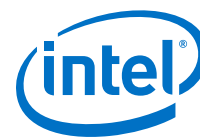


Contents

Revision History.....	3
Preface.....	13
Intended Audience.....	13
Intel® Omni-Path Documentation Library.....	13
Cluster Configurator for Intel® Omni-Path Fabric.....	15
Documentation Conventions.....	15
License Agreements.....	16
Technical Support.....	16
1.0 Introduction.....	17
1.1 Documentation Organization.....	17
2.0 Overview.....	18
2.1 Intel® Omni-Path Architecture Overview.....	18
2.1.1 Host Fabric Interface.....	20
2.1.2 Intel® OPA Switches.....	20
2.1.3 Intel® OPA Management.....	21
2.2 Intel® Omni-Path Software Overview.....	21
2.3 FastFabric Overview.....	23
2.3.1 FastFabric Architecture.....	23
2.3.2 FastFabric Capabilities.....	25
3.0 Getting Started.....	31
3.1 Important Note on First-Time Installations.....	31
3.2 Working with TUI Menus.....	31
3.2.1 Starting Up the Tools.....	31
3.2.2 Intel FastFabric OPA Tools TUI Overview.....	34
3.2.3 How to Use the FastFabric TUI.....	34
3.2.4 Fabric Performance Monitor TUI Overview.....	36
3.2.5 How to Use the Fabric Performance Monitor TUI.....	37
3.3 Working with CLI Commands.....	40
3.3.1 Common Tool Options.....	40
3.3.2 Selection of Devices.....	41
3.4 Configuration of IPoIB Name Mapping.....	49
3.5 Sample Files.....	49
3.5.1 List of Files.....	49
3.5.2 opagentopology.....	51
3.5.3 topology.xlsx Overview.....	53
3.6 Configuration Files for FastFabric	56
3.6.1 FastFabric Configuration File.....	57
3.6.2 Ports List Configuration File.....	57
3.6.3 Chassis List Configuration Files.....	58
3.6.4 Externally-Managed Switch List Configuration File.....	60
3.6.5 Hosts List Configuration Files.....	62
3.6.6 Port Statistics Thresholds Configuration File.....	63
3.6.7 Signal Integrity Thresholds Configuration File.....	63
3.6.8 Fabric Topology Input File.....	64



4.0 FastFabric TUI Menus.....	66
4.1 Managing the Chassis Configuration.....	66
4.1.1 Editing the Configuration Files for Chassis Setup.....	68
4.1.2 Verifying Chassis via Ethernet Ping.....	70
4.1.3 Updating the Chassis Firmware.....	71
4.1.4 Setting Up Chassis Basic Configuration.....	71
4.1.5 Setting Up Password-less ssh/scp.....	72
4.1.6 Rebooting the Chassis.....	73
4.1.7 Getting Basic Chassis Configuration.....	73
4.1.8 Configuring Chassis Fabric Manager.....	74
4.1.9 Updating the Chassis FM Security Files.....	78
4.1.10 Getting Chassis FM Security Files.....	79
4.1.11 Checking the OPA Fabric Status.....	79
4.1.12 Controlling Chassis Fabric Manager.....	80
4.1.13 Generating All Chassis Problem Report Information.....	81
4.1.14 Running a Command on All Chassis.....	82
4.1.15 Viewing opachassisadmin Result Files.....	82
4.2 Managing the Switch Configuration.....	83
4.2.1 Editing the Configuration Files for Externally-Managed Switch Setup.....	85
4.2.2 Generating or Updating Switch File.....	87
4.2.3 Testing for Switch Presence.....	88
4.2.4 Verifying Switch Firmware.....	88
4.2.5 Updating Switch Firmware.....	89
4.2.6 Setting Up Switch Basic Configuration.....	90
4.2.7 Rebooting the Switch.....	92
4.2.8 Reporting Switch Firmware and Hardware Information.....	92
4.2.9 Getting Basic Switch Configuration.....	93
4.2.10 Reporting Switch VPD Information.....	93
4.2.11 Viewing opaswitchadmin Result Files.....	94
4.3 Managing the Host Configuration.....	95
4.3.1 Editing the Configuration Files for Host Setup.....	96
4.3.2 Verifying Hosts are Pingable.....	98
4.3.3 Setting Up Password-Less SSH/SCP.....	99
4.3.4 Copying /etc/hosts to All Hosts.....	99
4.3.5 Showing uname -a for All Hosts.....	100
4.3.6 Installing/Upgrading OPA Software.....	100
4.3.7 Configuring IPoIB IP Address.....	101
4.3.8 Building Test Applications and Copying to Hosts.....	102
4.3.9 Rebooting Hosts.....	103
4.3.10 Refreshing SSH Known Hosts.....	103
4.3.11 Rebuilding MPI Library and Tools.....	104
4.3.12 Running a Command on All Hosts.....	106
4.3.13 Copying a File to All Hosts.....	107
4.3.14 Viewing opahostadmin Result Files.....	108
4.4 Verifying the Host.....	108
4.4.1 Editing the Configuration Files for Host Verification.....	110
4.4.2 Viewing a Summary of Fabric Components.....	112
4.4.3 Verifying Hosts Pingable, SSHable, and Active.....	113
4.4.4 Performing Single Host Verification.....	114
4.4.5 Verifying OPA Fabric Status and Topology.....	116



4.4.6 Verifying Hosts See Each Other.....	117
4.4.7 Verifying Hosts Ping via IPoIB.....	118
4.4.8 Refreshing SSH Known Hosts.....	119
4.4.9 Checking MPI Performance.....	120
4.4.10 Checking Overall Fabric Health.....	121
4.4.11 Starting or Stopping Bit Error Rate Cable Test.....	122
4.4.12 Generating All Hosts Problem Report Information.....	123
4.4.13 Running a Command on All Hosts.....	125
4.4.14 Viewing opahostadmin Result Files.....	126
5.0 Descriptions of Command Line Tools.....	128
5.1 High-Level TUIs.....	128
5.1.1 opafastfabric.....	128
5.1.2 opatop.....	129
5.2 Health Check and Baselining Tools.....	130
5.2.1 Usage Model.....	130
5.2.2 Common Operations and Options.....	131
5.2.3 opafabricanalysis.....	133
5.2.4 opachassisanalysis.....	137
5.2.5 opahostsmanalysis.....	142
5.2.6 opaesmanalysis.....	143
5.2.7 opaallanalysis.....	145
5.2.8 Manual and Automated Usage.....	147
5.2.9 Re-Establishing Health Check Baseline	147
5.2.10 Interpreting the Health Check Results.....	148
5.2.11 Interpreting Health Check .changes Files.....	151
5.3 Verification, Analysis, and Control CLIs.....	154
5.3.1 opacabletest.....	154
5.3.2 opaextractbadlinks.....	156
5.3.3 opaextractlink.....	158
5.3.4 opaextractmissinglinks.....	160
5.3.5 opaextractsellinks.....	162
5.3.6 opaextractstat2.....	164
5.3.7 opafindgood.....	166
5.3.8 opalinkanalysis.....	168
5.3.9 opareport.....	171
5.3.10 opareports.....	181
5.3.11 opareport Detailed Information.....	183
5.3.12 opaverifyhosts.....	208
5.3.13 opaxlattopology.....	210
5.3.14 opaxlattopology_cust.....	213
5.4 Detailed Fabric Data Gathering.....	214
5.4.1 opaextracterror.....	214
5.4.2 opaextractlids.....	216
5.4.3 opaextractperf.....	218
5.4.4 opaextractstat.....	220
5.4.5 opashowallports.....	223
5.5 Configuration and Control for Chassis, Switch, and Host	224
5.5.1 opagenswitches.....	224
5.5.2 opagenchassis.....	226
5.5.3 opagenesmchassis.....	227



5.5.4 opachassisadmin.....	228
5.5.5 opaswitchadmin.....	234
5.5.6 opahostadmin.....	239
5.5.7 Interpreting the opahostadmin, opachassisadmin, and opaswitchadmin log files.....	247
5.6 Basic Setup and Administration Tools.....	248
5.6.1 opapingall.....	248
5.6.2 opasetupssh.....	249
5.6.3 opacmdall.....	252
5.6.4 opacaptureall.....	255
5.7 File Management Tools.....	258
5.7.1 opascpall.....	258
5.7.2 opauploadall.....	259
5.7.3 opadownloadall.....	261
5.7.4 Simplified Editing of Node-Specific Files.....	262
5.7.5 Simplified Setup of Node-Generic Files.....	263
5.8 Fabric Link and Port Control.....	263
5.8.1 opadisableports.....	263
5.8.2 opaenableports.....	265
5.8.3 opadisablehosts.....	266
5.8.4 opaswdisableall.....	267
5.8.5 opaswenableall.....	268
5.8.6 opaledports.....	269
5.9 Fabric Debug.....	270
5.9.1 opafequery.....	270
5.9.2 opapaquery.....	277
5.9.3 opashowmc.....	283
5.10 FastFabric Utilities.....	284
5.10.1 opa2rm.....	284
5.10.2 opaexpandfile.....	286
5.10.3 opafirmware.....	286
5.10.4 opasorthosts.....	287
5.10.5 opaxmlextract.....	288
5.10.6 opaxmlfilter.....	291
5.10.7 opaxmlindent.....	292
5.10.8 opaxmlgenerate.....	292
5.10.9 opacheckload.....	294
6.0 Performance Monitoring.....	296
6.1 Monitoring Fabric Performance.....	296
6.1.1 Viewing the Fabric Performance Monitoring Summary Screen.....	296
6.1.2 Viewing the PM Configuration.....	298
6.1.3 Viewing Image Information.....	300
6.1.4 Viewing Bandwidth Utilization.....	302
6.1.5 Viewing Statistics Category.....	304
6.1.6 Viewing Configuration Information.....	307
6.1.7 Viewing Focus Information.....	309
6.1.8 Viewing Port Statistics.....	312
6.1.9 Navigating PM Sweeps.....	315
6.1.10 Bookmarking a Sweep.....	316
6.1.11 Using the opatop Command Line Options.....	317



6.2 Using Fabric Performance Data.....	318
6.2.1 Top Level Data.....	318
6.2.2 Mid-Tier Data.....	320
6.2.3 Lowest Tier Data.....	321
6.3 Port Counters Overview.....	322
6.3.1 Utilization.....	322
6.3.2 Link Integrity.....	322
6.3.3 Congestion.....	324
6.3.4 SMA Congestion.....	325
6.3.5 Bubble.....	325
6.3.6 Security.....	326
6.3.7 Routing.....	326
6.3.8 Other.....	327
7.0 FastFabric Diagnostics Capabilities.....	328
7.1 Overview.....	328
7.2 Accessing Link Quality Indicator Values.....	328
7.3 Topology Verification.....	329
7.3.1 Interpreting Output of Topology Verification Tools.....	331
7.4 Port Type Information.....	334
7.5 Link Down Reason.....	337
8.0 MPI Sample Applications.....	340
8.1 Building and Running Sample Applications.....	340
8.1.1 Building MPI Sample Applications.....	340
8.1.2 Running MPI Sample Applications.....	341
8.2 Latency/Bandwidth Deviation Test.....	342
8.3 OSU Tests.....	344
8.3.1 OSU Latency.....	344
8.3.2 OSU Latency2.....	344
8.3.3 OSU Latency 3.....	344
8.3.4 OSU Multi Latency3.....	345
8.3.5 OSU Bandwidth.....	345
8.3.6 OSU Bandwidth2.....	345
8.3.7 OSU Bandwidth3.....	346
8.3.8 OSU Multi Bandwidth3.....	346
8.3.9 OSU Bidirectional Bandwidth.....	346
8.3.10 OSU Bidirectional Bandwidth3.....	346
8.3.11 OSU All to All 3.....	347
8.3.12 OSU Broadcast 3.....	347
8.3.13 OSU Multiple Bandwidth/Message Rate.....	347
8.4 Latency Tests.....	348
8.4.1 Multi-Threaded Latency Test	348
8.4.2 Multi-Pair Latency Test.....	349
8.4.3 Broadcast Latency Test.....	349
8.4.4 One-Sided Put Latency Test	349
8.4.5 One-Sided Get Latency Test	349
8.4.6 One-Sided Accumulate Latency Test	349
8.5 Bandwidth Tests.....	349
8.5.1 Bidirectional Bandwidth Test.....	350
8.5.2 Multiple Bandwidth / Message Rate Test.....	350



8.5.3 One-Sided Put Bandwidth Test.....	350
8.5.4 One-Sided Get Bandwidth Test	350
8.5.5 One-Sided Put Bidirectional Bandwidth Test	350
8.6 mpi_stress Test.....	351
8.7 High Performance Linpack (HPL2).....	352
8.8 Intel® MPI Benchmarks (IMB).....	353
8.9 Pallas MPI Benchmark (PMB).....	353
8.10 MPI Fabric Stress Tests.....	354
8.10.1 All HFI Latency.....	354
8.10.2 run_cabletest.....	355
8.10.3 run_batch_cabletest.....	356
8.10.4 gen_group_hosts.....	358
8.10.5 run_multibw.....	358
8.10.6 run_nxnlatbw.....	359
8.11 MPI Batch run_* Scripts.....	359
8.11.1 SHMEM Batch run_* scripts.....	360
9.0 FastFabric Troubleshooting.....	361
9.1 Switching to the Intel P-State Driver to Run Certain FastFabric Tools.....	361
Appendix A Map of Intel® Omni-Path Architecture Commands.....	363



Figures

1	Intel® OPA Fabric.....	19
2	Intel® OPA Building Blocks.....	20
3	Intel® OPA Fabric and Software Components.....	22
4	FastFabric Architecture.....	24
5	Fabric Performance Monitor TUI Screen (Example).....	37
6	Fabric Performance Monitoring TUI Navigation.....	39
7	Topology Workflow.....	53
8	topology.xlsx Example.....	54



Tables

1	FastFabric Methods.....	25
2	FastFabric OPA Fabric Monitoring Menu Descriptions.....	33
3	Fabric Performance Monitor TUI Descriptions.....	37
4	Common Tool Options.....	41
5	Core Full Statement Definitions.....	55
6	Present Leaf Statement Definitions.....	56
7	Omitted Spines Statement Definitions.....	56
8	SM Statement Definition.....	56
9	FastFabric OPA Chassis Setup/Admin Menu Descriptions.....	67
10	FastFabric OPA Switch Setup/Admin Menu Descriptions.....	84
11	FastFabric OPA Host Setup Menu Descriptions.....	96
12	FastFabric OPA Host Verification/Admin Menu Descriptions.....	109
13	Performance Impact.....	121
14	Possible issues found in health check .changes files.....	152
15	Summary Screen Field Descriptions.....	297
16	PM Configuration Field Descriptions.....	299
17	Image Information Field Descriptions.....	301
18	Bandwidth Statistics Field Descriptions.....	303
19	Statistics Field Descriptions.....	306
20	Configuration Information Field Descriptions.....	308
21	Focus Information Field Descriptions.....	311
22	Port Statistics Field Descriptions.....	313
23	Link Quality Values and Description.....	323
24	Rank Assignment.....	347
25	Map of InfiniBand*, Intel® True Scale, and Intel® OPA Commands.....	363



Preface

This manual is part of the documentation set for the Intel® Omni-Path Fabric (Intel® OP Fabric), which is an end-to-end solution consisting of Intel® Omni-Path Host Fabric Interfaces (HFIs), Intel® Omni-Path switches, and fabric management and development tools.

The Intel® OP Fabric delivers a platform for the next generation of High-Performance Computing (HPC) systems that is designed to cost-effectively meet the scale, density, and reliability requirements of large-scale HPC clusters.

Both the Intel® OP Fabric and standard InfiniBand* are able to send Internet Protocol (IP) traffic over the fabric, or *IPoFabric*. In this document, however, it is referred to as *IP over IB* or *IPoIB*. From a software point of view, IPoFabric and IPoIB behave the same way and, in fact, use the same `ib_ipoib` driver to send IP traffic over the `ib0` and/or `ib1` ports.

Intended Audience

The intended audience for the Intel® Omni-Path (Intel® OP) document set is network administrators and other qualified personnel.

Intel® Omni-Path Documentation Library

Intel® Omni-Path publications are available at the following URLs:

- Intel® Omni-Path Switches Installation, User, and Reference Guides
<http://www.intel.com/omnipath/SwitchPublications>
- Intel® Omni-Path Software Installation, User, and Reference Guides (includes HFI documents)
<http://www.intel.com/omnipath/FabricSoftwarePublications>
- Drivers and Software (including Release Notes)
<http://www.intel.com/omnipath/Downloads>

Use the tasks listed in this table to find the corresponding Intel® Omni-Path document.

Task	Document Title	Description
Key: Shading indicates the URL to use for accessing the particular document.		
• Intel® Omni-Path Switches Installation, User, and Reference Guides:	http://www.intel.com/omnipath/SwitchPublications	
• Intel® Omni-Path Software Installation, User, and Reference Guides (includes HFI documents):	http://www.intel.com/omnipath/FabricSoftwarePublications (no shading)	
• Drivers and Software (including Release Notes):	http://www.intel.com/omnipath/Downloads	
<i>continued...</i>		



Task	Document Title	Description
Using the Intel® OPA documentation set	<i>Intel® Omni-Path Fabric Quick Start Guide</i>	A roadmap to Intel's comprehensive library of publications describing all aspects of the product family. It outlines the most basic steps for getting your Intel® Omni-Path Architecture (Intel® OPA) cluster installed and operational.
Setting up an Intel® OPA cluster	<i>Intel® Omni-Path Fabric Setup Guide</i> (Old title: <i>Intel® Omni-Path Fabric Staging Guide</i>)	Provides a high level overview of the steps required to stage a customer-based installation of the Intel® Omni-Path Fabric. Procedures and key reference documents, such as Intel® Omni-Path user guides and installation guides are provided to clarify the process. Additional commands and BKM's are defined to facilitate the installation process and troubleshooting.
Installing hardware	<i>Intel® Omni-Path Fabric Switches Hardware Installation Guide</i>	Describes the hardware installation and initial configuration tasks for the Intel® Omni-Path Switches 100 Series. This includes: Intel® Omni-Path Edge Switches 100 Series, 24 and 48-port configurable Edge switches, and Intel® Omni-Path Director Class Switches 100 Series.
	<i>Intel® Omni-Path Host Fabric Interface Installation Guide</i>	Contains instructions for installing the HFI in an Intel® OPA cluster. A cluster is defined as a collection of nodes, each attached to a fabric through the Intel interconnect. The Intel® HFI utilizes Intel® Omni-Path switches and cabling.
Installing host software Installing HFI firmware Installing switch firmware (externally-managed switches)	<i>Intel® Omni-Path Fabric Software Installation Guide</i>	Describes using a Text-based User Interface (TUI) to guide you through the installation process. You have the option of using command line interface (CLI) commands to perform the installation or install using the Linux* distribution software.
Managing a switch using Chassis Viewer GUI Installing switch firmware (managed switches)	<i>Intel® Omni-Path Fabric Switches GUI User Guide</i>	Describes the Intel® Omni-Path Fabric Chassis Viewer graphical user interface (GUI). It provides task-oriented procedures for configuring and managing the Intel® Omni-Path Switch family. Help: GUI online help.
Managing a switch using the CLI Installing switch firmware (managed switches)	<i>Intel® Omni-Path Fabric Switches Command Line Interface Reference Guide</i>	Describes the command line interface (CLI) task information for the Intel® Omni-Path Switch family. Help: -help for each CLI.
Managing a fabric using FastFabric	<i>Intel® Omni-Path Fabric Suite FastFabric User Guide</i> (Merged with: <i>Intel® Omni-Path Fabric Suite FastFabric Command Line Interface Reference Guide</i>)	Provides instructions for using the set of fabric management tools designed to simplify and optimize common fabric management tasks. The management tools consist of TUI menus and command line interface (CLI) commands. Help: -help and man pages for each CLI. Also, all host CLI commands can be accessed as console help in the Fabric Manager GUI.
Managing a fabric using Fabric Manager	<i>Intel® Omni-Path Fabric Suite Fabric Manager User Guide</i>	The Fabric Manager uses a well defined management protocol to communicate with management agents in every Intel® Omni-Path Host Fabric Interface (HFI) and switch. Through these interfaces the Fabric Manager is able to discover, configure, and monitor the fabric.
	<i>Intel® Omni-Path Fabric Suite Fabric Manager GUI User Guide</i>	Provides an intuitive, scalable dashboard and set of analysis tools for graphically monitoring fabric status and configuration. It is a user-friendly alternative to traditional command-line tools for day-to-day monitoring of fabric health. Help: Fabric Manager GUI Online Help.
continued...		



Task	Document Title	Description
Configuring and administering Intel® HFI and IPoIB driver Running MPI applications on Intel® OPA	<i>Intel® Omni-Path Fabric Host Software User Guide</i>	Describes how to set up and administer the Host Fabric Interface (HFI) after the software has been installed. The audience for this document includes both cluster administrators and Message-Passing Interface (MPI) application programmers, who have different but overlapping interests in the details of the technology.
Writing and running middleware that uses Intel® OPA	<i>Intel® Performance Scaled Messaging 2 (PSM2) Programmer's Guide</i>	Provides a reference for programmers working with the Intel® PSM2 Application Programming Interface (API). The Performance Scaled Messaging 2 API (PSM2 API) is a low-level user-level communications interface.
Optimizing system performance	<i>Intel® Omni-Path Fabric Performance Tuning User Guide</i>	Describes BIOS settings and parameters that have been shown to ensure best performance, or make performance more consistent, on Intel® Omni-Path Architecture. If you are interested in benchmarking the performance of your system, these tips may help you obtain better performance.
Designing an IP or storage router on Intel® OPA	<i>Intel® Omni-Path IP and Storage Router Design Guide</i>	Describes how to install, configure, and administer an IPoIB router solution (Linux* IP or LNet) for inter-operating between Intel® Omni-Path and a legacy InfiniBand* fabric.
Building a Lustre* Server using Intel® OPA	<i>Building Lustre* Servers with Intel® Omni-Path Architecture Application Note</i>	Describes the steps to build and test a Lustre* system (MGS, MDT, MDS, OSS, OST, client) from the HPDD master branch on a x86_64, RHEL*/CentOS* 7.1 machine.
Building Containers for Intel® OPA fabrics	<i>Building Containers for Intel® Omni-Path Fabrics using Docker* and Singularity* Application Note</i>	Provides basic information for building and running Docker* and Singularity* containers on Linux*-based computer platforms that incorporate Intel® Omni-Path networking technology.
Writing management applications that interface with Intel® OPA	<i>Intel® Omni-Path Management API Programmer's Guide</i>	Contains a reference for programmers working with the Intel® Omni-Path Architecture Management (Intel OPAMGT) Application Programming Interface (API). The Intel OPAMGT API is a C-API permitting in-band and out-of-band queries of the FM's Subnet Administrator and Performance Administrator.
Learning about new release features, open issues, and resolved issues for a particular release	<i>Intel® Omni-Path Fabric Software Release Notes</i>	
	<i>Intel® Omni-Path Fabric Manager GUI Release Notes</i>	
	<i>Intel® Omni-Path Fabric Switches Release Notes (includes managed and externally-managed switches)</i>	

Cluster Configurator for Intel® Omni-Path Fabric

The Cluster Configurator for Intel® Omni-Path Fabric is available at: <http://www.intel.com/content/www/us/en/high-performance-computing-fabrics/omni-path-configurator.html>.

This tool generates sample cluster configurations based on key cluster attributes, including a side-by-side comparison of up to four cluster configurations. The tool also generates parts lists and cluster diagrams.

Documentation Conventions

The following conventions are standard for Intel® Omni-Path documentation:

- **Note:** provides additional information.
- **Caution:** indicates the presence of a hazard that has the potential of causing damage to data or equipment.



- **Warning:** indicates the presence of a hazard that has the potential of causing personal injury.
- Text in [blue](#) font indicates a hyperlink (jump) to a figure, table, or section in this guide. Links to websites are also shown in blue. For example:
See [License Agreements](#) on page 16 for more information.
For more information, visit www.intel.com.
- Text in **bold** font indicates user interface elements such as menu items, buttons, check boxes, key names, key strokes, or column headings. For example:
Click the **Start** button, point to **Programs**, point to **Accessories**, and then click **Command Prompt**.
Press **CTRL+P** and then press the **UP ARROW** key.
- Text in `Courier` font indicates a file name, directory path, or command line text. For example:
Enter the following command: `sh ./install.bin`
- Text in *italics* indicates terms, emphasis, variables, or document titles. For example:
Refer to *Intel® Omni-Path Fabric Software Installation Guide* for details.
In this document, the term *chassis* refers to a managed switch.

Procedures and information may be marked with one of the following qualifications:

- **(Linux)** – Tasks are only applicable when Linux* is being used.
- **(Host)** – Tasks are only applicable when Intel® Omni-Path Fabric Host Software or Intel® Omni-Path Fabric Suite is being used on the hosts.
- **(Switch)** – Tasks are applicable only when Intel® Omni-Path Switches or Chassis are being used.
- Tasks that are generally applicable to all environments are not marked.

License Agreements

This software is provided under one or more license agreements. Please refer to the license agreement(s) provided with the software for specific detail. Do not install or use the software until you have carefully read and agree to the terms and conditions of the license agreement(s). By loading or using the software, you agree to the terms of the license agreement(s). If you do not wish to so agree, do not install or use the software.

Technical Support

Technical support for Intel® Omni-Path products is available 24 hours a day, 365 days a year. Please contact Intel Customer Support or visit <http://www.intel.com/omnipath/support> for additional detail.



1.0 Introduction

This manual provides instructions for using the Intel® Omni-Path Fabric Suite FastFabric, a set of fabric management tools designed to simplify and optimize common fabric management tasks.

For details about the other documents for the Intel® Omni-Path product line, refer to [Intel® Omni-Path Documentation Library](#) on page 13 of this document.

The management tools consist of TUI menus and command line interface (CLI) commands. All of the functions that the TUI menus perform can also be performed using CLI commands. To aid in learning the commands, the TUI shows each CLI command as it executes it.

Throughout this document and the FastFabric Tools, "chassis" refers to managed switches and "switches" refers to externally-managed switches.

Note: This manual assumes that you have already installed the Intel® Omni-Path Software as prescribed in the *Intel® Omni-Path Fabric Software Installation Guide*

1.1 Documentation Organization

This manual is organized as follows:

- This **Introduction** provides an overview of this document and its structure.
- **Overview** provides an overview of the Intel® Omni-Path and FastFabric architecture and capabilities.
- **Getting Started** provides instructions and information for starting up and using the FastFabric TUI and CLI tools as well as an introduction to configuration files.
- **FastFabric TUI Menus** provide instructions for setting up, managing, and verifying managed chassis, externally-managed switches, and hosts.
- **Descriptions of Command Line Tools** provides complete descriptions of each CLI tool and its parameters.
- **Performance Monitoring** provides instructions for monitoring the performance, congestion, and statistics information of a fabric as well as the port categories and counters used by the Intel® Omni-Path Fabric.
- **FastFabric Diagnostics Capabilities** provides information about the FastFabric features that help you diagnose fabric issues.
- **MPI Sample Applications** provides a variety of sample applications that can be used to perform basic tests and performance analysis.
- **FastFabric Troubleshooting** provides you with instructions and tips for troubleshooting common issues when operating FastFabric tools.
- Appendix A, **Map of Intel® Omni-Path Architecture Commands**, provides a mapping of commands between InfiniBand*, Intel® True Scale, and Intel® Omni-Path Architecture.

2.0 Overview

This section provides an overview of the Intel® Omni-Path Architecture and Intel® Omni-Path Fabric Suite FastFabric.

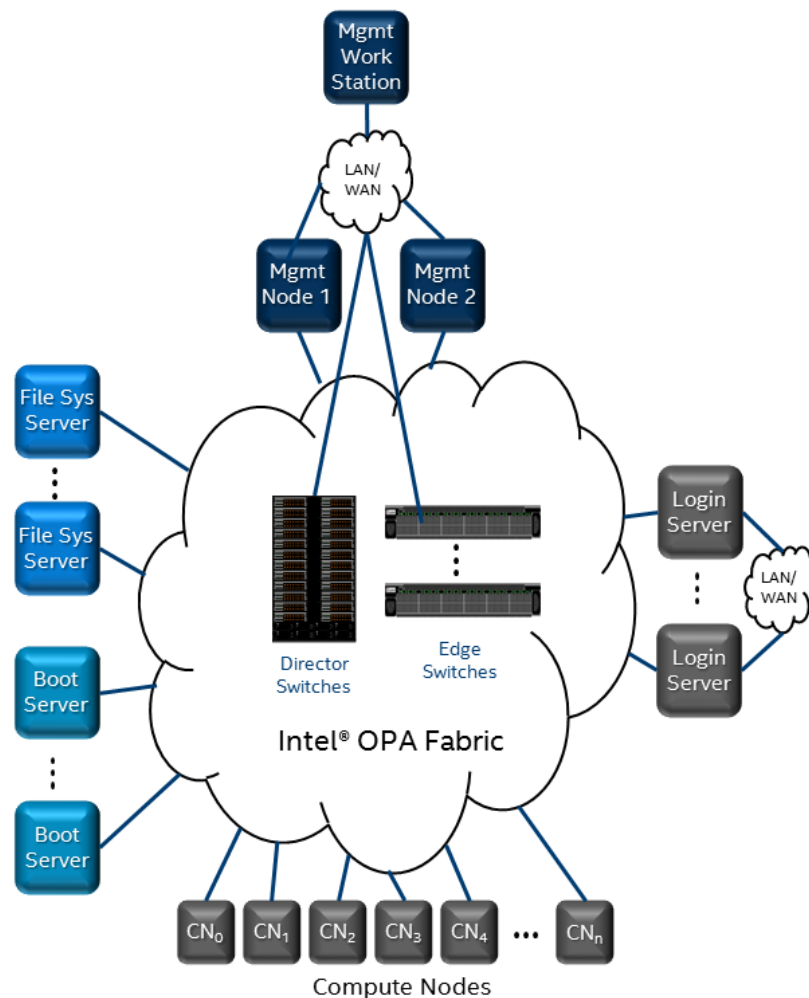
2.1 Intel® Omni-Path Architecture Overview

The Intel® Omni-Path Architecture (Intel® OPA) interconnect fabric design enables a broad class of multiple node computational applications requiring scalable, tightly-coupled processing, memory, and storage resources. Options for close "on-package" integration between Intel® OPA family devices, Intel® Xeon® Processors, and Intel® Xeon Phi™ Processors, enable significant system level packaging and network efficiency improvements. When coupled with open standard APIs developed by the OpenFabrics Alliance* (OFA) Open Fabrics Interface (OFI) workgroup, host fabric interfaces (HFIs) and switches in the Intel® OPA family systems are optimized to provide the low latency, high bandwidth, and high message rate needed by large scale High Performance Computing (HPC) applications.

Intel® OPA provides innovations for a multi-generation, scalable fabric, including link layer reliability, extended fabric addressing, and optimizations for many-core processors. High performance datacenter needs are also a core Intel® OPA focus, including link level traffic flow optimization to minimize datacenter-wide jitter for high priority packets, robust partitioning support, quality of service support, and a centralized fabric management system.

The following figure shows a sample Intel® OPA-based fabric, consisting of different types of nodes and servers.

Figure 1. Intel® OPA Fabric



To enable the largest scale systems in both HPC and the datacenter, fabric reliability is enhanced by combining the link level retry typically found in HPC fabrics with the conventional end-to-end retry used in traditional networks. Layer 2 network addressing is extended for systems with over ten million endpoints, thereby enabling use on the largest scale datacenters for years to come.

To enable support for a breadth of topologies, Intel® OPA provides mechanisms for packets to change virtual lanes as they progress through the fabric. In addition, higher priority packets are able to preempt lower priority packets to provide more predictable system performance, especially when multiple applications are running simultaneously. Finally, fabric partitioning is provided to isolate traffic between jobs or between users.

The software ecosystem is built around OFA software and includes four key APIs.

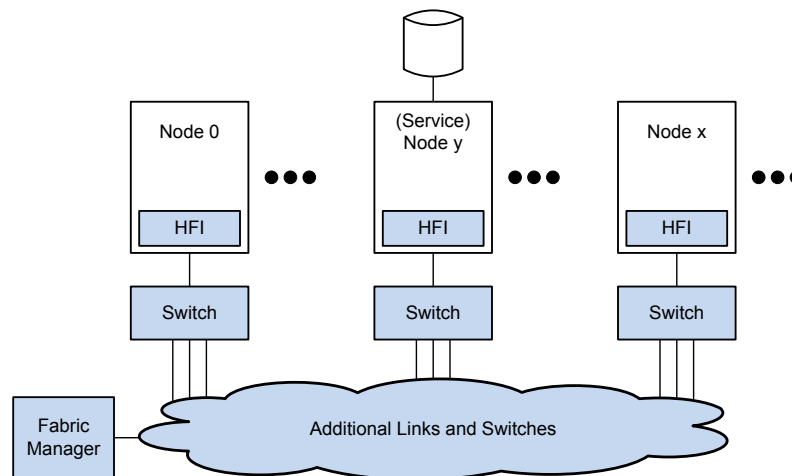
1. The OFA OFI represents a long term direction for high performance user level and kernel level network APIs.

2. The Performance Scaled Messaging 2 (PSM2) API provides HPC-focused transports and an evolutionary software path from the Intel® True Scale Fabric.
3. OFA Verbs provides support for existing remote direct memory access (RDMA) applications and includes extensions to support Intel® OPA fabric management.
4. Sockets is supported via OFA IPoFabric (also called IPoIB) and rSockets interfaces. This permits many existing applications to immediately run on Intel® Omni-Path as well as provide TCP/IP features such as IP routing and network bonding.

Higher level communication libraries, such as the Message Passing Interface (MPI), and Partitioned Global Address Space (PGAS) libraries, are layered on top of these low level OFA APIs. This permits existing HPC applications to immediately take advantage of advanced Intel® Omni-Path features.

Intel® Omni-Path Architecture is an end-to-end solution consisting of Intel® Omni-Path Host Fabric Interfaces (HFIs), Intel® Omni-Path switches, and fabric management and development tools. These building blocks are shown in the following figure.

Figure 2. Intel® OPA Building Blocks



2.1.1 Host Fabric Interface

Each host is connected to the fabric through a Host Fabric Interface (HFI) adapter. The HFI translates instructions between the host processor and the fabric. The HFI includes the logic necessary to implement the physical and link layers of the fabric architecture, so that a node can attach to a fabric and send and receive packets to other servers or devices. HFIs also include specialized logic for executing and accelerating upper layer protocols.

2.1.2 Intel® OPA Switches

Intel® OPA switches are OSI Layer 2 (link layer) devices, and act as packet forwarding mechanisms within a single Intel® OPA fabric. Intel® OPA switches are responsible for implementing Quality of Service (QoS) features, such as virtual lanes, congestion management, and adaptive routing. Switches are centrally managed by the Intel® Omni-Path Fabric Suite Fabric Manager software, and each switch includes a



management agent to handle management transactions. Central management means that switch configurations are programmed by the FM software, including managing the forwarding tables to implement specific fabric topologies, configuring the QoS and security parameters, and providing alternate routes for adaptive routing. As such, all OPA switches must include management agents to communicate with the Intel® OPA Fabric Manager.

2.1.3 Intel® OPA Management

The Intel® OPA fabric is centrally managed and supports redundant Fabric Managers that manage every device (server and switch) in the fabric through management agents associated with those devices. The Primary Fabric Manager is an Intel® OPA fabric software component selected during the fabric initialization process.

The Primary Fabric Manager is responsible for:

1. Discovering the fabric's topology.
2. Setting up Fabric addressing and other necessary values needed for operating the fabric.
3. Creating and populating the Switch forwarding tables.
4. Maintaining the Fabric Management Database.
5. Monitoring fabric utilization, performance, and statistics rates.

The fabric is managed by sending management packets over the fabric. These packets are sent *in-band* (that is, over the same wires as regular network packets) using dedicated buffers on a specific virtual lane (VL15). End-to-end reliability protocols are used to detect lost packets.

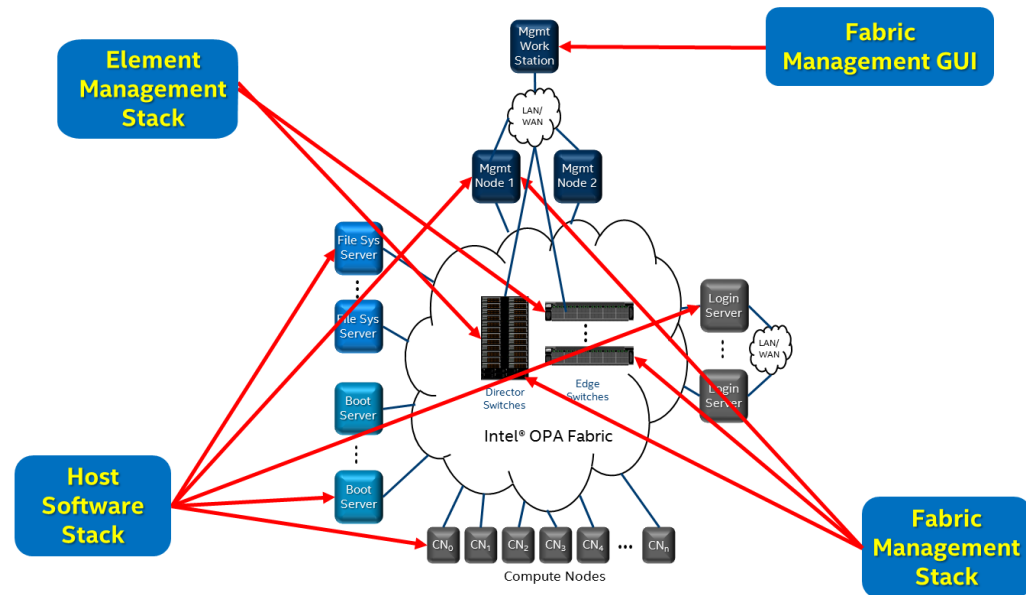
2.2 Intel® Omni-Path Software Overview

For software applications, Intel® OPA maintains consistency and compatibility with existing Intel® True Scale Fabric and InfiniBand* APIs utilizing the open source OpenFabrics Alliance* (OFA) software stack on Linux* distribution releases.

Software Components

The key software components and their usage models are shown in the following figure and described in the following paragraphs.

Figure 3. Intel® OPA Fabric and Software Components



Software Component Descriptions

Element Management Stack

- Runs on an embedded Intel processor included in managed Intel® OP Edge Switch 100 Series and Intel® Omni-Path Director Class Switch 100 Series switches.
- Provides system management capabilities, including signal integrity, thermal monitoring, and voltage monitoring, among others.
- Accessed via Ethernet* port using command line interface (CLI) or graphical user interface (GUI).

User documents:

- *Intel® Omni-Path Fabric Switches GUI User Guide*
- *Intel® Omni-Path Fabric Switches Command Line Interface Reference Guide*

Host Software Stack

- Runs on all Intel® OPA-connected host nodes and supports compute, management, and I/O nodes.
- Provides a rich set of APIs including OFI, PSM2, sockets, and OFA verbs.
- Provides high performance, highly scalable MPI implementation via OFA, PSM2, and an extensive set of upper layer protocols.
- Includes Boot over Fabric mechanism for configuring a server to boot over Intel® Omni-Path using the Intel® OP HFI Unified Extensible Firmware Interface (UEFI) firmware.

User documents:

- *Intel® Omni-Path Fabric Host Software User Guide*

continued...



Software Component Descriptions
<ul style="list-style-type: none"> Intel® Performance Scaled Messaging 2 (PSM2) Programmer's Guide
Fabric Management Stack <ul style="list-style-type: none"> Runs on Intel® OPA-connected management nodes or embedded Intel processor on the switch. Initializes, configures, and monitors the fabric routing, QoS, security, and performance. Includes a toolkit for configuration, monitoring, diagnostics, and repair. User documents: <ul style="list-style-type: none"> Intel® Omni-Path Fabric Suite Fabric Manager User Guide Intel® Omni-Path Fabric Suite FastFabric User Guide
Fabric Management GUI <ul style="list-style-type: none"> Runs on laptop or workstation with a local screen and keyboard. Provides interactive GUI access to Fabric Management features such as configuration, monitoring, diagnostics, and element management drill down. User documents: <ul style="list-style-type: none"> Intel® Omni-Path Fabric Suite Fabric Manager GUI Online Help Intel® Omni-Path Fabric Suite Fabric Manager GUI User Guide

2.3 FastFabric Overview

Intel® Omni-Path Fabric Suite FastFabric is a set of fabric management tools designed to simplify and optimize common fabric management tasks. FastFabric includes the following capabilities:

- Monitoring and diagnostic tools
- Fabric deployment and verification
- Switch management
- Host management

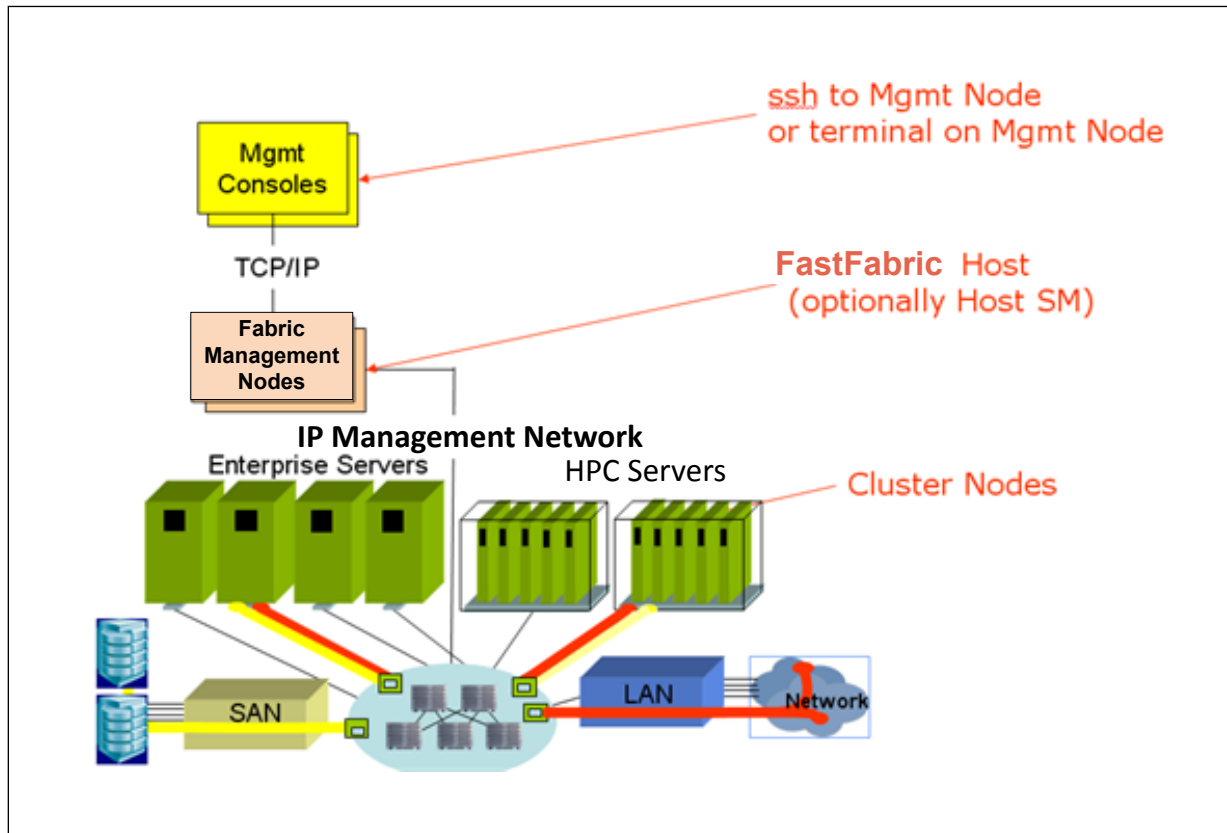
FastFabric consists of a hierarchy of commands and tools. In order to simplify learning and use, these tools all have similar command line arguments. Many of the FastFabric tools are designed to be easily extended via scripting or exporting data into other formats, such as spreadsheets.

The higher level tools allow you to focus on the names assigned to devices, and avoid the need to figure out LIDs or remember GUIDs for basic operations. As such, Intel recommends that you establish a naming convention for the cluster, and assign names to all the hosts and switches in the cluster.

2.3.1 FastFabric Architecture

FastFabric is typically installed on one or more Fabric Management Nodes. The Fabric Management Node must be connected to the rest of the cluster through the Intel® Omni-Path Fabric and a management network. The management network may be the primary Internet Protocol over InfiniBand* (IPoIB) network or Ethernet*. The management network is used for FastFabric host setup and administration tasks. It may also be used for other aspects of server administration or operation. Refer to the following figure for a high-level block diagram of the FastFabric architecture.

Figure 4. FastFabric Architecture



Depending on cluster size and design, the Fabric Management node may also be used as the master node for starting Message Passing Interface (MPI) jobs. It may also be used to run an Intel® Omni-Path Fabric Suite Fabric Manager and other management software. Refer to the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide* for details and what combinations are valid.

Note: When IPoIB is used as the management network, FastFabric is not able to install host software or configure IPoIB. However in this configuration, FastFabric is able to support host software upgrades, verification, and all its other features.

If remote access to FastFabric is desired, set up remote access to the Fabric Management Node using the Intel® Omni-Path Fabric Suite Fabric Manager GUI, ssh, Telnet, X-Windows, VNC or any other mechanism that will allow the remote user to access a Linux* Command Line shell. Typically FastFabric is used only by cluster administrators.

2.3.1.1 How FastFabric Works

FastFabric manages two types of switching devices that are managed by the "Chassis Setup/Admin" and "Externally Managed Switch Setup/Admin" menus.

The Chassis menu allows management of switching devices that are termed "managed." These include both edge and director class switching devices that have one or more management cards in place. The management card provides an



environment that exposes various TCP/IP services. This includes a command line interpreter login shell environment, with which FastFabric communicates. The device has an active Ethernet connection for LAN connectivity. The user is instructed to build a list of chassis in a "chassis" file, listing either the IP addresses or host names of the chassis to be managed; FastFabric provides tools to help discover such devices in the fabric and construct such a file. Communication with these devices is primarily out-of-band.

The Externally Managed Switch menu allows management of switching devices that are edge switches without management cards. Consequently, there is no environment present to provide TCP/IP services. There is no active Ethernet connection on the device. Therefore, communication to these devices must be accomplished in-band, via Intel® OPA management protocols designed specifically for this purpose. The user is instructed to build a list of switches in a "switches" file, listing the GUIDs of the switches to be managed; FastFabric provides tools to help discover such devices in the fabric and construct such a file.

FastFabric consists of a variety of tools to administrate hosts, chassis and externally managed switches. Depending on the tool, the method of accessing and administering the target devices may differ.

The following table describes the access methods that FastFabric uses.

Table 1. FastFabric Methods

Method	Examples
Inband access	Fabric performance, statistics, and congestion monitoring. Fabric topology reports, SA database queries, fabric error and link speed analysis, tools for externally managed switches, etc.
Log in through a management network	Host setup and installation, tools for managed chassis, etc.
MPI job startup (can be inband or through a management network)	Verify MPI performance, running sample MPI benchmarks, host-to-switch cable test.

Tools that log into other hosts will do so in a password-less manner using ssh. Tools that log into managed chassis can also use ssh. Chassis tools can prompt for a single password for all chassis, use password-less ssh, or can be pre-configured with the password. These approaches permit the tools to operate with minimal user interaction, and for this reason reduce the time to perform operations against many hosts or chassis.

After initial installation, FastFabric can be configured to use IPoIB instead of the management network.

Note: IPoIB cannot be used to reconfigure IPoIB or install new hosts.

2.3.2 FastFabric Capabilities

2.3.2.1 FastFabric Command Hierarchy

FastFabric provides numerous powerful commands. These commands can be best understood as a hierarchy of capabilities permitting operations at high, mid and low levels.



2.3.2.1.1 Monitoring and Diagnostics

At the highest level, FastFabric provides an interactive Text-based User Interface (TUI), called `opafastfabric`. The TUI provides an easy and efficient way to perform fabric deployment and verification, and diagnosis of typical fabrics. The TUI is structured in the typical sequence of operations for fabric verification. All of the functions that the TUI performs are also available using command line interface (CLI) commands. To aid in learning the commands, the TUI shows each CLI command as it executes it.

Other high level tools can provide an initial view of fabric status and health. These include the Intel® Omni-Path Fabric Suite Fabric Manager GUI, the interactive cluster statistics and performance display tool (`opatop`), and the tools to verify cluster status as compared to a previous baseline (`opaallanalysis` and its sub-tools: `opalinkanalysis`, `opafabricanalysis`, `opachassisanalysis`, `opaesmanalysis`, and `opahostsmanalysis`).

When analyzing the fabric at a mid-tier of information, the next tier of tools include: `opafabricinfo`, `opareports`, `opareport`, `opaextractbadlinks`, `opaextractlink`, `opaextractsellinks`, and `opaextractstat2`. These tools provide very powerful ways to query the fabric. The `opaextract*` family of tools are all scripts that take advantage of `opareport` to generate delimited files that can be easily parsed or exported into spreadsheets for offline analysis. These scripts can also be good samples for the creation of site-specific sysadmin scripts.

At the next level of lower analysis there are additional tools. These provide direct access to more of the raw fabric information, such as port counters, LIDs, and other configured parameters. Tools in this tier include: `opashowallports`, `opaextractlids`, `opaextracterror`, `opaextractperf`, `opaextractstat`. Many of these tools are scripts that are also built on top of `opareport`. `opareport` is a foundational tool in FastFabric that provides a rich set of fabric analysis capabilities, and can provide both high level and very detailed output.

At the lowest level of analysis are tools that can access the management protocols directly. This can permit all the details of a given port or device to be viewed or analyzed. Typically, these tools only need to be used when debugging subtle issues. These tools include: `opahfirev`, `opashowmc`, `opafirmware`, `opasaquery`, `opapaquery`, `opafequery`, `opasmaquery`, `opaportinfo`, and `opapmaquery`.

2.3.2.1.2 Benchmark and Stress Tests

FastFabric includes a number of benchmarks and stress tests. These can be found in `/usr/src/opa/mpi_apps` and `/usr/mpi/*/*/tests`. The `opacabletest` tool also provides a simple way to create high stress on all links in the fabric to aid in the verification of fabric stability.

In addition, other existing Intel® Omni-Path benchmarks and test programs may also be used to exercise the `libfabric` and `verbs` interfaces.

2.3.2.2 Host and Switch Management

FastFabric includes tools to manage both managed and externally-managed switches, as well as hosts. These tools are in addition to the fundamental operational controls that the Fabric Manager provides for all devices in the fabric. Many of these capabilities are also available in the `opafastfabric` TUI.



For externally-managed switches, `opagenswitches` can assist in generating a list of the devices currently in the fabric, and `opaswitchadmin` provides the primary control and query functions to manage firmware, check status, and reboot.

For managed switches, `opagenchassis` can assist in generating a list of the devices currently in the fabric (`opagenesmchassis` will generate the list of those currently running an embedded subnet manager (ESM)), and `opachassisadmin` provides the primary control and query functions to manage firmware, check status, and root. In addition `opapingall`, `opacmdall`, and `opasetupssh` can provide direct access to the switch CLI.

For hosts, `opahostadmin` provides typical control and query functions to manage host software and configuration. `opafindgood` and `opaverifyhosts` can provide analysis of the host status. In addition, `opapingall`, `opacmdall`, `opascpall`, `opadownloadall`, `opauploadall`, and `opasetupssh` are tools that are included to perform basic ssh and scp operations against the hosts.

2.3.2.3 Topology Analysis

FastFabric includes a rich set of topology analysis and verification capabilities. This can start with a pre-assembly description of the cluster design, from which `opaxlattopology` or `opaxlattopology_cust` (deprecated) can generate a `topology.xml` file for use by FastFabric and the Fabric Manager.

`opareport` has a number of reports for verifying the topology (`-o verify*`) and can do analysis of the current routing for credit loops, degree of path balance, and so forth. In addition, reports such as `opareport -o links`, `opaextractlink` and `opaextractsellinks` can provide an in-depth view of the fabric connectivity and design.

2.3.2.4 Link and Port Management

From a fabric perspective, a fabric consists of numerous fabric ports. The Fabric Manager controls and configures these ports, but FastFabric also includes a rich set of tools to analyze ports and perform some basic control functions, such as bouncing a port.

`opainfo` is an easy-to-use tool that can provide the primary status of the current host's ports. For controlling ports, `opaportconfig` can control a single port, while `opaenableports` and `opadisableports` can use the output from `opaextractbadlinks` or `opaextractsellinks` to disable a list of ports. `opadisablehosts` can also disable the ports on a list of hosts. `opaenableports` can reen able ports. For switches, `opaswdisableall` can disable unused ports on switches, while `opaswenableall` can re-enable them. To get all the low-level details of port status and configuration, `opaportinfo`, `opasmaquery` and `opapmaquery` may be used.

2.3.2.5 Focused Fabric Feature Analysis

Tools and reports are available to provide in-depth analysis of various fabric features.

Link quality, signal integrity, security errors, routing errors, and other issues can be analyzed using the following:



- `opafastfabric TUI`
- Intel® Omni-Path Fabric Suite Fabric Manager GUI
- `opatop`
- `opaallanalysis`
- `opaextractbadlinks`
- `opareport` (such as `-o errors`, `-o slow*`, and `-o mis*` reports)
- `opashowallports`

The details of Quality of Service (QoS) configuration and operation can be reviewed using the following:

- Intel® Omni-Path Fabric Suite Fabric Manager GUI
- `opasaquery -o vfinfo`
- `opasaquery -o path`
- various `opareport` options (such as `-ovfinfo`, `-o vfmember`, and `-o bfrctrl`)
- all the low-level details can be reviewed using `opasmaquery opareport -V -o comps -d 10`, and assorted `opasaquery` reports that show SL, SC and VL tables

Fabric routing can be analyzed using various `opareport` options, such as:

- `-o portusage`
- `-o treepathusage`
- `-o pathusage`
- `-o portgroups`
- `-o validateroutes`
- `-o validatepgs`
- `-o validatecreditloops`
- `-o linear`
- `-o mcast`

To view or analyze Link Width Downgrade (LWD), the following commands can be used:

- `opareport -o slowlinks`
- `opafabricanalysis` (uses `opareport -o slowlinks`)
- `opaextracterror` (uses `opareport -o comps`, shows main error counters)
- `opaextractperf` (uses `opareport -o comps`, shows per port counters)
- `opalinkanalysis slowlinks`



2.3.2.6 Scripting and Integration Enablement

Various additional tools can facilitate extending FastFabric, or integrating it with other tools. Among these are the XML processing tools (`opaxmlextract`, `opaxmlfilter`, and `opaxmlindent`), which can permit the XML output formats from `opareport` and/or the `opafm.xml` file itself to be easily parsed and analyzed in other scripts. The `opaextract*` scripts can provide samples of how to effectively use these tools.

`opagetvf` and `opagetvf_env` provide an easy way to extract key virtual fabric parameters to aid job scheduler and job launch integration with virtual fabrics.

2.3.2.7 Scripting on Top of FastFabric

Intel® Omni-Path Fabric Suite FastFabric was designed to make the scripting of OEM or site-specific tools easy to use. However, to ensure forward compatibility, scripts should be created using tools and arguments that are documented in [Descriptions of Command Line Tools](#) on page 128.

A number of the tools, such as the `opa*analysis` set of tools, are designed for easy use through exit code checks. These tools can easily be scripted to be run, and then, on bad exit codes, to issue emails or other forms of alerts to system administrators. Such mechanisms can be scheduled for regular execution by way of cron jobs. The file that is created by these tools can then be analyzed by the system administrators.

`opareport` is a powerhouse tool that provides a wide range of fabric data-gathering and analysis capabilities. The best way to script with this tool is to take advantage of its `-x` option to output XML. That output can then be easily parsed by `opaxmlextract` to extract sets of fields into delimited formats that can then be easily parsed by scripts, or exported to external tools such as spreadsheets. The `opa*extract` set of scripts are all built on top of `opareport -x` and `opaxmlextract`. These scripts can provide a great starting point by copying them and then creating new variations to meet your unique needs.

Intel recommends against creating scripts that attempt to directly parse `opareport -o` snapshot output. This format cannot be guaranteed to be forward-compatible with future FastFabric software releases. Most of the information in an `opareport -o` snapshot is also available in a forward-compatible format via `opareport -x -o comps -d 10 -s`. The remainder can be found in other [opareport](#) output by using different options.

Intel also recommends against creating scripts that attempt to parse the human-readable output formats produced by the tools. Intel reserves the right to refine these formats in future FastFabric software releases, and therefore, these formats cannot be guaranteed to be forward-compatible.

2.3.2.8 Customer Support Data Gathering

Detailed information about the current fabric status and configuration can be quickly obtained using `opacapture` for a single node, or `opacaptureall` for multiple nodes, to aid customer support.



2.3.2.9 Other Tools and Capabilities

In addition, a number of the non-Infiniband*-specific OpenFabrics Alliance* (OFA) tools will continue to function on an Intel® Omni-Path Fabric, and can provide additional information. Among these are `ibv_devinfo` (note that MTU will not correctly report MTUs beyond 4K), `ibstat` (note that some of the extended physical states for Intel® Omni-Path ports, such as offline, will not be reported properly), `ibsrpdm`, and `ibv_devices`.

The `ibacm` Distributed Subnet Administrator Provider (DSAP) plugin may be queried and debugged using `opa_osd_dump`, `opa_osd_exercise`, `opa_osd_perf`, and `opa_osd_query`.

`opapacketcapture` can provide traces of many of the verbs packets, including all the management packets, to enable offline analysis and debug in Wireshark using the Intel® Omni-Path dissector.



3.0 Getting Started

This section provides instructions and information for getting started with the Intel® Omni-Path Fabric Suite FastFabric tools.

3.1 Important Note on First-Time Installations

This user guide is not an installation guide.

If you are installing and configuring the fabric for the first time, you must refer to the *Intel® Omni-Path Fabric Software Installation Guide*.

3.2 Working with TUI Menus

One method for working with the FastFabric toolset is through the TUI menus. This method provides a more guided, task-oriented approach for using the tools, prompting the user for information and values.

3.2.1 Starting Up the Tools

Note: To run the Intel® Omni-Path Fabric Suite FastFabric tools described in this manual, you must have root privileges.

3.2.1.1 Accessing the Intel FastFabric OPA Tools Menu

The Intel FastFabric OPA Tools menu allows you to configure and manage the Intel® Omni-Path Fabric.

Using the `opafastfabric` Command

To start up the Intel FastFabric OPA Tools menu from the command prompt, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter `opafastfabric`.

The Intel FastFabric OPA Tools menu is displayed.

```
Intel FastFabric OPA Tools
Version: X.X.X.X.X

1) Chassis Setup/Admin
2) Externally Managed Switch Setup/Admin
3) Host Setup
4) Host Verification/Admin
5) Fabric Monitoring

X) Exit
```



From the Intel OPA Software Menu

To start up the Intel FastFabric OPA Tools menu from the Intel OPA Software main menu, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opaconfig**.

The Intel OPA [version] Software main menu is displayed.

```
Intel OPA X.X.X.X.X Software

  1) Show Installed Software
  2) Reconfigure OFA IP over IB
  3) Reconfigure Driver Autostart
  4) Generate Supporting Information for Problem Report
  5) FastFabric (Host/Chassis/Switch Setup/Admin)
  6) Uninstall Software

X) Exit
```

3. At the cursor, type 5.

The Intel FastFabric OPA Tools menu is displayed.

```
Intel FastFabric OPA Tools
Version: X.X.X.X.X

  1) Chassis Setup/Admin
  2) Externally Managed Switch Setup/Admin
  3) Host Setup
  4) Host Verification/Admin
  5) Fabric Monitoring

X) Exit
```

3.2.1.2 Accessing the Fabric Performance Monitor

The Fabric Performance Monitor allows you to monitor performance, congestion, and statistics information in a fabric.

Using the **opatop** Command

To start up the Fabric Performance Monitor from the command prompt, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opatop**.

The Fabric Performance Monitor Summary screen is displayed.

```
opatop: Img: 10s @ Wed Sep 14 11:29:52 2016, Live
Summary:  SW:      0 Ports: SW:      0 HFI:      2      Link:      1
          SM:      1 Node NRsp:      0 Skip:      0 Port NRsp:      0 Skip:      0
          AvgMBps  MinMBps  MaxMBps  AvgKPps  MinKPps  MaxKPps
0 All      Int      0        0        0        0        0        0
   Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min
1 HFIs     Int      0        0        0        0        0        0
   Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min
2 SWs      No ports in group
```




```

Master-SM: LID: 0x0001 Port: 1 Priority: 0 State: Master
          Name: phcppriv10 hfi1_0
          PortGUID: 0x0011750101575300
Secondary-SM: none

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help |
sS Pmcfg Imginfo View 0-n:

```

From the Intel FastFabric OPA Tools Menu

To start up the Fabric Performance Monitor menu from the Intel FastFabric OPA Tools menu, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opafastfabric**.

The Intel FastFabric OPA Tools menu is displayed.

```

Intel FastFabric OPA Tools
Version: X.X.X.X.X

1) Chassis Setup/Admin
2) Externally Managed Switch Setup/Admin
3) Host Setup
4) Host Verification/Admin
5) Fabric Monitoring

X) Exit

```

3. At the cursor, type 5.

The FastFabric OPA Fabric Monitoring menu is displayed.

```

FastFabric OPA Fabric Monitoring Menu

0) Fabric Performance Monitoring          [ Skip ]

P) Perform the Selected Actions           N) Select None
X) Return to Previous Menu (or ESC)

```

Table 2. FastFabric OPA Fabric Monitoring Menu Descriptions

Menu Item	Description
0) Fabric Performance Monitoring	Allows you to access the TUI that monitors the performance, congestion, and statistics information about a fabric. Associated CLI Command: opatop

4. Type 0 to toggle to the [Perform] option.

5. Type P to perform the operation.

The Fabric Performance Monitor information is displayed.

```

opatop: Img: 10s @ Fri Sep 16 11:35:24 2016, Live
Summary: SW:      0 Ports: SW:      0 HFI:      2      Link:      1
          SM:      1 Node NRsp:    0 Skip:    0 Port NRsp:    0 Skip:    0
          AvgMBps  MinMBps  MaxMBps  AvgKPps  MinKPps  MaxKPps

```



```
0 All      Int      0      0      0      0      0      0
   Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min
1 HFIs     Int      0      0      0      0      0      0
   Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min
2 SWs      No ports in group

Master-SM: LID: 0x0001 Port: 1 Priority: 0 State: Master
           Name: phcppriv10 hf1l_0
           PortGUID: 0x0011750101575300
Secondary-SM: none

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help |
sS Pmcfg Imginfo View 0-n:
```

3.2.2 Intel FastFabric OPA Tools TUI Overview

The Intel FastFabric OPA Tools TUI allows you to perform common fabric management tasks including setting up and managing the chassis, switches, and hosts.

Note: For detailed information on associated CLI tools and options, refer to [Descriptions of Command Line Tools](#) on page 128.

The following is an example of the Intel FastFabric OPA Tools main menu.

```
Intel FastFabric OPA Tools
Version: X.X.X.X.X

1) Chassis Setup/Admin
2) Externally Managed Switch Setup/Admin
3) Host Setup
4) Host Verification/Admin
5) Fabric Monitoring

X) Exit
```

3.2.3 How to Use the FastFabric TUI

The FastFabric TUI menus are set up for ease of use. The submenus are designed to present operations in the order they would typically be used during an installation.

Note: All FastFabric TUI menu alpha-based options are case-insensitive.

Selecting Menu Items and Performing Operations

1. From the Intel FastFabric OPA Tools main menu, select the target menu item (0-4¹).

```
Intel FastFabric OPA Tools
Version: X.X.X.X.X

1) Chassis Setup/Admin
2) Externally Managed Switch Setup/Admin
```

1 For menu item 5, refer to [How to Use the Fabric Performance Monitor TUI](#) on page 37.



```

3) Host Setup
4) Host Verification/Admin
5) Fabric Monitoring

X) Exit

```

The target menu is displayed as shown in the example below:

```

FastFabric OPA Chassis Setup/Admin Menu
Chassis File: /etc/opa/chassis
Setup:
0) Edit Config and Select/Edit Chassis File [ Skip ]
1) Verify Chassis via Ethernet Ping [ Skip ]
2) Update Chassis Firmware [ Skip ]
3) Set Up Chassis Basic Configuration [ Skip ]
4) Set Up Password-Less SSH/SCP [ Skip ]
5) Reboot Chassis [ Skip ]
6) Get Basic Chassis Configuration [ Skip ]
7) Configure Chassis Fabric Manager (FM) [ Skip ]
8) Update Chassis FM Security Files [ Skip ]
9) Get Chassis FM Security Files [ Skip ]
Admin:
a) Check OPA Fabric Status [ Skip ]
b) Control Chassis Fabric Manager (FM) [ Skip ]
c) Generate All Chassis Problem Report Info [ Skip ]
d) Run a Command on All Chassis [ Skip ]
Review:
e) View opachassisadmin Result Files [ Skip ]

P) Perform the Selected Actions N) Select None
X) Return to Previous Menu (or ESC)

```

2. Type the key corresponding to the target menu item (0-9, a-d) to toggle the Skip/Perform selection.

More than one item may be selected.

3. Type P to perform the operations that were selected.

Notes:

- If more than one menu item is selected, the operations are performed in the order shown in the menu. This is the typical order desired during fabric setup.
- If you want to perform operations in a different order, you must select the first target menu item, type P to perform the operation, then repeat this process for the next menu item operation to be performed, and so on.

4. Type N to clear all selected items.
5. Type X or press Esc to exit this menu and return to the Main Menu.

Aborting Operations

While multiple menu items are performing, you have an opportunity to abort individual operations as they come up. After each operation completes and before the next operation begins, you are prompted as shown below:

```
Hit any key to continue...
```

- Press Esc to stop the sequence of operations return to the previous menu.
- Any unperformed operations are still highlighted in the menu. To complete the selected operations, type P.



- Press any other key to perform the next selected menu item being performed.
This prompt is also shown after the last selected item completes, providing an opportunity to review the results before the screen is cleared to display the menu.

Submenu Configuration Files

On each FastFabric submenu, item 0 permits a different file to be selected and edited (using the editor selected by the EDITOR environment variable). It also permits reviewing and editing of the `opafastfabric.conf` file. The `opafastfabric.conf` file guides the overall configuration of FastFabric and describes cluster-specific attributes of how FastFabric operates. It is discussed in greater detail in [Configuration Files for FastFabric](#) on page 56.

At the top of each FastFabric submenu screen beneath the title, the directory and configuration file containing the components on which to operate are shown.

In the example below, the configuration file is noted in bold.

```
FastFabric OPA Host Setup Menu
Host File: /etc/opa/hosts
Setup:
0) Edit Config and Select/Edit Host File      [ Skip ]
1) Verify Hosts Pingable                     [ Skip ]
2) Set Up Password-Less SSH/SCP              [ Skip ]
```

Note: During the execution of each menu selection, the actual FastFabric command line tool being used is shown. This can be used as an educational aid to learn the command line tools.

The example snippet below shows how the CLI is displayed in the TUI execution.

```
Performing Chassis Admin: Verify Chassis via Ethernet Ping
Executing: /usr/sbin/opapingall -C -p -F /etc/opa/chassis
```

3.2.4 Fabric Performance Monitor TUI Overview

The Fabric Performance Monitor TUI allows you to monitor performance, congestion, and statistics information about a fabric.

The following is an example of the (`opatop`) Fabric Performance Monitor TUI.

```
opatop: Img: 10s @ Wed Sep 14 11:29:52 2016, Live
Summary: SW:      0 Ports: SW:      0 HFI:      2      Link:      1
          SM:      1 Node NRsp:      0 Skip:      0 Port NRsp:      0 Skip:      0
          AvgMBps  MinMBps  MaxMBps  AvgKPps  MinKPps  MaxKPps
0 All      Int      0         0         0         0         0         0
  Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min
1 HFIs     Int      0         0         0         0         0         0
  Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min
2 SWs      No ports in group

Master-SM: LID: 0x0001 Port: 1 Priority: 0 State: Master
           Name: phcppriv10 hfi1_0
           PortGUID: 0x0011750101575300
Secondary-SM: none
```



```
Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help |
sS Pmcfg Imginfo View 0-n:
```

3.2.5 How to Use the Fabric Performance Monitor TUI

The Fabric Performance Monitor TUI allow you to view and interact with live performance data.

Reading the TUI Screens

The figure below shows the major sections common to all Fabric Performance Monitor TUI screens.

Figure 5. Fabric Performance Monitor TUI Screen (Example)

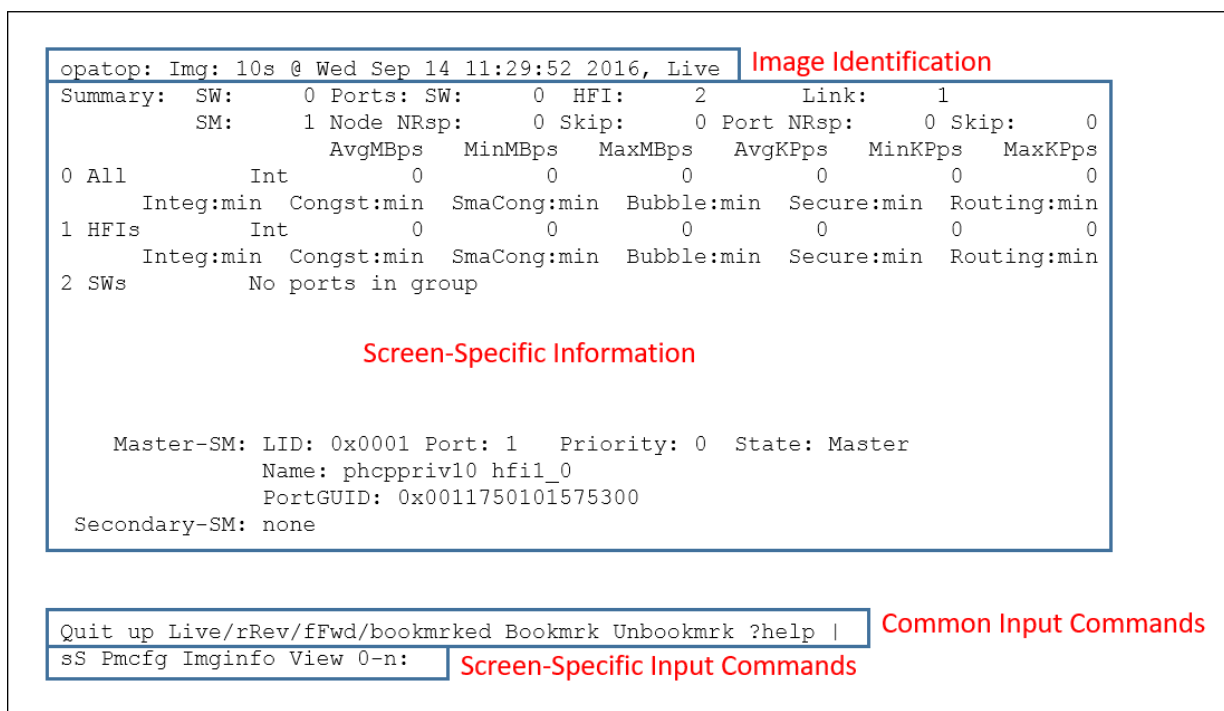


Table 3. Fabric Performance Monitor TUI Descriptions

Section of Screen	Description
opatop	Refers to the CLI command that initiates to the Fabric Performance Monitoring TUI.
<i>continued...</i>	



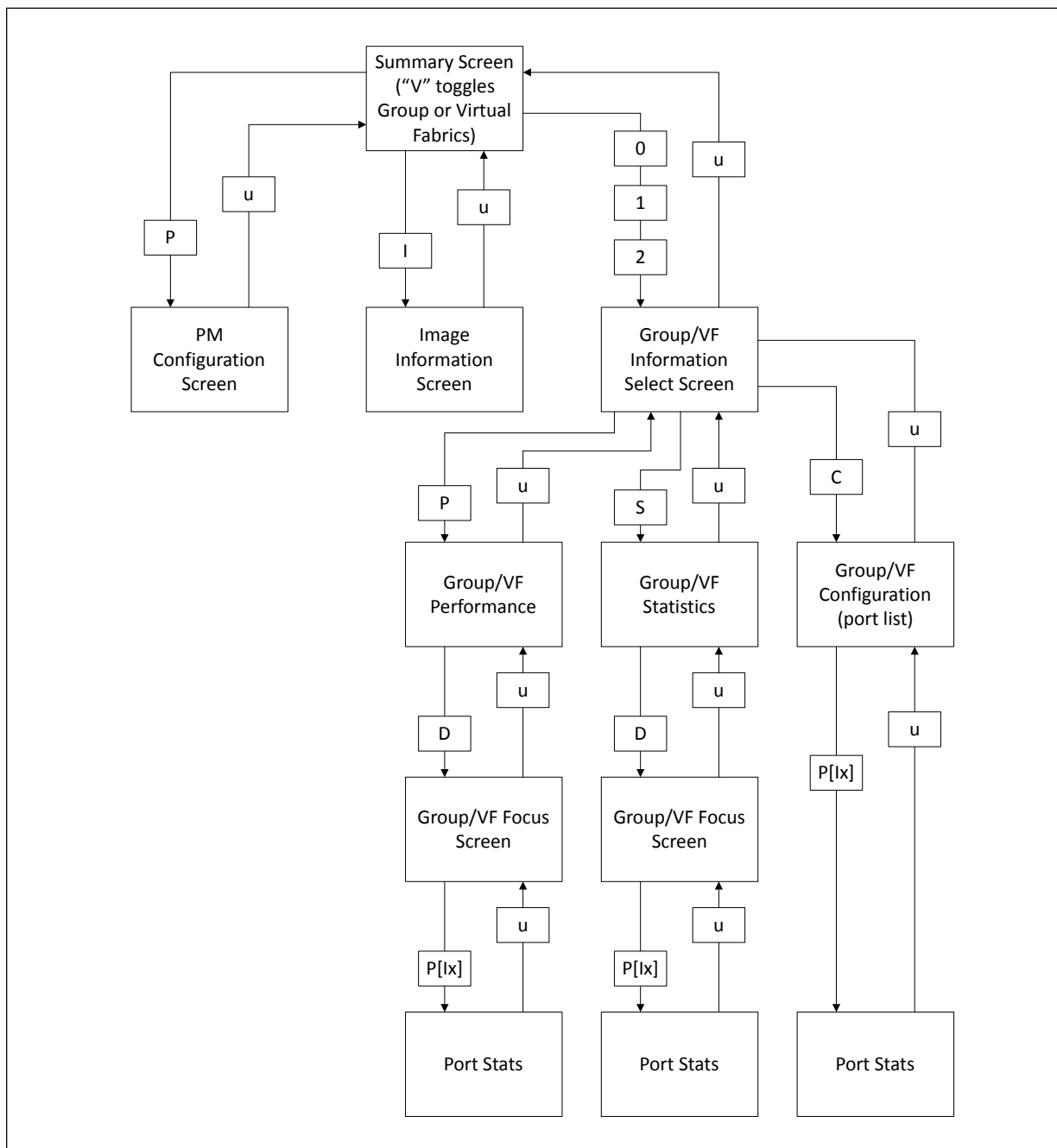
Section of Screen	Description
	NOTE: opatop may be used interchangeably with Intel® Fabric Performance Monitoring TUI within this manual.
Image Identification	<p>Displays the following image (Img) information:</p> <ul style="list-style-type: none">Image interval (II): The time over which this image data is relevant.<ul style="list-style-type: none">For in-memory images, this value is equal to the PM Sweep Interval.For images stored on disk (Short Term History), the interval is equal to the sum of all the intervals for each image compounded into the composite (disk) image. <p>NOTE: The interval can change when transitioning between images stored in memory and images stored on disk.</p> <ul style="list-style-type: none">Timestamp for the image being displayed in the format Day Month Date HR:MIN:SEC YYYY (example, Wed Sep 14 11:29:52 2016) If a Live image is not being displayed, the current time ('Now:') is also shown.Type of image<ul style="list-style-type: none">LiveHist (History)Bkmk (Bookmark)
Screen-Specific Information	<p>Displays information and layout of the selected screen.</p> <p>NOTE: Each screen is different and will be discussed in subsequent sections.</p>
Common Input Commands	<p>Displays the common input commands that appear on every screen and perform the same action.</p> <ul style="list-style-type: none">Q/q – Quit programu – Up to previous screenL – Select Live imager – Navigate reverse 1 sweepR – Navigate reverse 5 sweepsf – Navigate forward 1 sweepF – Navigate forward 5 sweepsb – Select (previously) bookmarked imageB – Bookmark currently selected imageU – Unbookmark image? – Help provides information about the screen contents and input commands. <p>Commands are case insensitive except where specifically noted otherwise. The ENTER key must be pressed after multi-character commands and for Quit.</p>
Screen-Specific Input Commands	Displays the screen-specific commands.

Navigating the Screens

The Fabric Performance Monitoring TUI allows you to access various screens in a hierarchal manner to examine the state of a fabric. Through the screen-specific commands, each screen will provide access to the next screen or back to the parent screen.

The Fabric Performance Monitoring TUI screen navigational hierarchy is shown below.

Figure 6. Fabric Performance Monitoring TUI Navigation



As an example, if you want to navigate from the Group Info Sel screen to the Group BW Stats screen, perform the following steps:

1. The Group Info Sel screen is shown below.

```
opatopt: Img: 10s @ Thu Sep 22 15:44:47 2016, Live
Group Info Sel: HFIs
Int NumPorts: 2   Rate Min: 100g   Max: 100g
```



```
Ext NumPorts: 0
Group Performance (P)
Group Statistics (S)
Group Config (C)

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | P S C:
```

The selections for the next level of screens are displayed as:

```
Group Performance (P)
Group Statistics (S)
Group Config (C)
```

The menu options are shown in the screen-specific commands as:

```
Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | P S C:
```

2. From the Group Info Sel screen, enter **P**.

The Group BW Stats screen is displayed.

```
opatop: Img: 10s @ Thu Sep 22 15:52:27 2016, Live
Group Performance: HFIs Criteria: Util-High Number: 10
Int: TotMBps AvgMBps MinMBps MaxMBps TotKPps AvgKPps MinKPps
MaxKPps
0 0 0 0 0 0 0
0 Buckt 0+% 10+% 20+% 30+% 40+% 50+% 60+% 70+% 80+% 90+%
2 0 0 0 0 0 0 0 0 0 0
NoResp Int Ports: PMA: 0 Topo: 0
Int Congestion Max 0+ 25+ 50+ 75+ 100+
0 2 0 0 0 0
Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | cC N0-n Detail:
```

3. Type **u** (lowercase) to return to the Group Info Sel screen.
4. Type **u** (lowercase) to return to the Summary screen.

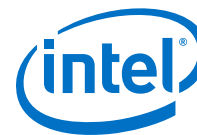
Important: To switch between Port and Virtual Fabric Grouping screens, press **v** at the Summary screen and navigate through the hierarchy.

3.3 Working with CLI Commands

Another method for working with the FastFabric toolset is through CLI commands. This method requires more advanced knowledge of FastFabric, and provides more control of the tools through individual parameters.

3.3.1 Common Tool Options

The following table lists the common CLI options that are applicable to most of the tools.

**Table 4. Common Tool Options**

Command	Description
-?	Displays basic usage information for any of the commands. An invalid option also displays this information.
--help	Displays complete usage information for most of the commands.
-p	<p>Runs the operation/command in parallel. This means the operation is performed simultaneously on batches of FF_MAX_PARALLEL hosts. (Default = 1000.) This option allows the overall time of an operation to be much lower. However, a side effect is that any output from the command is bursty and intermingled. Therefore, this option should be used for commands where there is no output or the output is of limited interest. For some commands (such as <code>opascpall</code>), this performs the operation in a quiet mode to limit output. If you want to change the number of parallel operations, export <code>FF_MAX_PARALLEL=#</code> where # is the new number (such as 500).</p> <p>For more advanced operations (such as <code>opahostadmin</code>, <code>opachassisadmin</code>, and <code>opaswitchadmin</code>), parallel operation is the default mode.</p> <p>Parallel operation can also be disabled by setting <code>FF_MAX_PARALLEL</code> to 1.</p>
-S	<p>Prompts for password for admin on chassis or root on host. By default, Intel® Omni-Path Fabric Suite FastFabric toolset operations against Intel® Omni-Path Chassis (such as <code>opacmdall</code>, <code>opacaptureall</code>, and <code>opachassisadmin</code>) obtain the chassis admin password from the <code>FF_CHASSIS_ADMIN_PASSWORD</code> environment variable which may be directly exported or part of <code>opafastfabric.conf</code>. Alternatively, you can use the <code>-S</code> option to be interactively prompted for the chassis admin password. The password is prompted for once, and the same password is then used to log in to each chassis during the operation.</p> <p>For hosts, this option is only applicable to <code>opasetupssh</code>.</p> <p>Note: All versions of Intel® Omni-Path Chassis firmware permit SSH keys to be configured within the chassis for secure password-less login. In this case, there is no need to configure a <code>FF_CHASSIS_ADMIN_PASSWORD</code> environment variable, and <code>FF_CHASSIS_LOGIN_METHOD</code> can be set to SSH. Intel recommends you set up a secure SSH password-less login using <code>opasetupssh -C</code>. Refer to the <i>Intel® Omni-Path Fabric Switches GUI User Guide</i> for more information.</p>
-C	Specifies that the given operation should be performed against chassis. By default, many Intel® Omni-Path Fabric Suite FastFabric toolset operations are performed against hosts. However, selected FastFabric toolset commands (such as <code>opacmdall</code> , <code>opapingall</code> , and <code>opacaptureall</code>) can also operate against Intel® Omni-Path managed chassis. When <code>-C</code> is specified, the operation is performed against chassis instead of hosts. Refer to Selection of Devices for details about the selection of chassis.
-h	Select which local HFI to use.
-p	Select which local HFI port to use.
-v	Produces verbose output.

3.3.2 Selection of Devices

This chapter describes how you choose devices for CLI commands. In general, you can select a number of devices through list files or explicitly identify devices by their names or formats within the command.

3.3.2.1 Selection of Hosts

To perform operations against a set of hosts, you can specify the hosts on which to operate using one of the following methods:

- On the command line, using the `-h` option.
- Using the environment variable `HOSTS` to specify a space-separated list of hosts. Useful when multiple commands are performed against the same small set of hosts.

- Using the `-f` option or the `HOSTS_FILE` environment variable to specify a file containing the set of hosts. Useful for groups of hosts that are used often. The file is located here: `/etc/opa/hosts` by default. The file must list all hosts in the cluster except the host running the FastFabric toolset itself.

Within the tools, the options are considered in the following order:

1. `-h` option
2. `HOSTS` environment variable
3. `-f` option
4. `HOSTS_FILE` environment variable
5. `/etc/opa/hosts` file

For example, if the `-h` option is used and the `HOSTS_FILE` environment variable is also exported, the command operates only on hosts specified using the `-h` option.

3.3.2.1.1 Host List Files

You can use the `-f` option to provide the name of a file containing the list of hosts on which to operate. The default location is `/etc/opa/hosts`.

It may be useful to create multiple files in `/etc/opa` representing different subsets of the fabric. For example:

- `/etc/opa/hosts-mpi` – list of MPI hosts
- `/etc/opa/hosts-fs` – list of file server hosts
- `/etc/opa/hosts` – list of all hosts except for the FastFabric toolset node
- `/etc/opa/allhosts` – list of all hosts including the FastFabric toolset node

Host List File Format

Sample host list file:

```
# this is a comment
192.168.0.4 # host identified by IP address
n001 # host identified by resolvable TCP/IP name
include /etc/opa/hosts-mpi # included file
```

Each line of the host list file may specify a single host, a comment, or another host list file to include.

Hosts may be specified by IP address or a resolvable TCP/IP host name. Typically, host names are used for readability. Also, some FastFabric toolset commands translate the supplied host names to IPoIB hostnames, in which case names are generally easier to translate than numeric IP addresses. Typically management network hostnames are specified. However, if desired, IPoIB hostnames or IP addresses may be used to accelerate large file transfers and other operations.

Files to be included may be specified using an `include` directive followed by a file name. File names specified should generally be absolute pathnames. If relative pathnames are used, they are searched for in the current directory first, then `/etc/opa`.



Comments may be placed on any line by using a # to precede the comment. On lines with hosts or include directives, the # must be white space-separated from any preceding hostname, IP address, or included file name.

3.3.2.1.2 Explicit Host Names

When hosts are explicitly specified using the `-h` option or the `HOSTS` environment variable, a space-separated list of host names (or IP addresses) may be supplied. For example: `-h 'host1 host2 host3'`

3.3.2.2 Selection of Chassis

To perform operations against a set of chassis, you can specify the chassis on which to operate using one of the following methods:

- On the command line, using the `-H` option.
- Using the environment variable `CHASSIS` to specify a space-separated list of chassis. Useful when multiple commands are performed against the same small set of chassis.
- Using the `-F` option or the `CHASSIS_FILE` environment variable to specify a file containing the set of chassis. Useful for groups of chassis that will be used often. The file is located here: `/etc/opa/chassis` by default. The file must list all chassis in the cluster.

Within the tools, the options are considered in the following order:

1. `-H` option
2. `CHASSIS` environment variable
3. `-F` option
4. `CHASSIS_FILE` environment variable
5. `/etc/opa/chassis` file

For example, if the `-H` option is used and the `CHASSIS_FILE` environment variable is also exported, the command operates only on chassis specified by the `-H` option.

3.3.2.2.1 Chassis List Files

You can use the `-F` option to provide the name of a file containing the list of chassis on which to operate. The default is `/etc/opa/chassis`.

It may be useful to create multiple files in `/etc/opa` representing different subsets of the fabric. For example:

- `/etc/opa/chassis-core`: list of core switching chassis
- `/etc/opa/chassis-edge`: list of edge switching chassis
- `/etc/opa/esm_chassis`: list of chassis running an SM
- `/etc/opa/chassis`: list of all chassis

If a relative path is specified for the `-F` option, the current directory is checked first, followed by `/etc/opa/`.

Chassis List File Format

Sample chassis file:

```
# this is a comment
192.168.0.5 # chassis IP address
edge1 # chassis resolvable TCP/IP name
include /etc/opa/chassis-core # included file
```

Each line of the chassis list file may specify a single chassis, a comment, or another chassis list file to include.

A chassis may be specified by chassis management network IP address or a resolvable TCP/IP name. Typically, names are used for readability.

Files to be included may be specified using an `include` directive followed by a file name. File names specified should be absolute path names. If relative path names are used, they are searched for in the current directory first, then `/etc/opa`.

Comments may be placed on any line using a `#` to precede the comment. On lines with chassis or `include` directives, the `#` must be white space-separated from any preceding name, IP address, or included file name.

The chassis file can also be generated using the `opagenchassis` command.

3.3.2.2.2 Explicit Chassis Names

When chassis are explicitly specified using the `-H` option or the `CHASSIS` environment variable, a space-separated list of names (or IP addresses) may be supplied. For example: `-H chassis1 chassis2 chassis3`.

3.3.2.2.3 Selection of Slots within a Chassis

Typically, operations are performed against the primary management module (MM) in the chassis. For operations such as `opacmdall`, you can specify the management module for the given chassis, if there is a redundant/secondary MM.

To perform operations against a specific subset of cards within the chassis, you can augment the chassis IP address or name within a chassis list or a chassis file with a list of slot numbers on which to operate. Use the form:

```
chassis:slot1,slot2,...
```

For example:

```
i9k229:0
i9k229:0,1,5
192.168.0.5:0,1,5
```

Note: No spaces can be used within the chassis name and slot list.

This format may be used whenever a chassis name or IP address is valid, such as the `-H` option, the `CHASSIS` environment variable, or chassis list files.

The slot number specified may be ignored on some operations.



Only slots containing MM may be specified with this format. Use the `chassisQuery` command to identify MM slots.

Note: For any operation, be careful that a given chassis is listed only once with all relevant slots. This prevents conflicting concurrent operations against a given chassis.

3.3.2.3 Selection of Switches

To perform operations against a set of externally-managed switches, you can specify the switch on which to operate using one of the following methods:

- On the command line, using the `-N` option.
- Using the environment variable `SWITCHES` to specify a space-separated list of switches. Useful when multiple commands are performed against the same small set of switches.
- Using the `-L` option or the `SWITCHES_FILE` environment variable to specify a file containing the set of switches. Useful for groups of switches that are used often. The file is located here: `/etc/opa/switches` by default. The file must list all switches in the cluster.

Within the tools, the options are considered in the following order:

1. `-N` option
2. `SWITCHES` environment variable
3. `-L` option
4. `SWITCHES_FILE` environment variable
5. `/etc/opa/switches` file

For example, if the `-N` option is used and the `SWITCHES_FILE` environment variable is also exported, the command operates only on switches specified using the `-N` option.

3.3.2.3.1 Switch List Files

You can use the `-L` option to provide the name of a file containing the list of switches on which to operate. The default is `/etc/opa/switches`.

It may be useful to create multiple files in `/etc/opa` representing different subsets of the fabric.

If a relative path is specified for the `-L` option or `SWITCHES_FILE` environment variable, the current directory is checked first, followed by `/etc/opa/`.

Switch List File Format

Sample switch list file:

```
# this is a comment
0x00117500d9000138,i9k138 # Node GUID with desired Name
0x00117500d9000139,i9k139 # Node GUID with desired Name
0x00117500d9000140:1:2,i9k140 # Node GUID with port and Name
0x00117500d9000141,i9k141,1 # Node GUID with desired Name, short distance
0x00117500d9000142,i9k142,5 # Node GUID with desired Name, longer distance
include /etc/opa/moreswitches # included file
```



Each line of the switch list file may specify a single switch, a comment, or another switch list file to include.

Switches can be specified by node GUID, optionally followed by a colon and the hfi:port, optionally followed by a comma and the Node Description (nodename) to be assigned to the switch, and optionally followed by the distance value indicating the relative distance from the FastFabric node for each switch.

You can use `opagenswitches` to locate externally-managed switches in the fabric and generate a `switches` file. By default, `opagenswitches` provides the proper distance value relative to the FastFabric node from which it was run. Alternatively, the `opagenswitches -R` option suppresses generation of this field.

When you use `opagenswitches` in conjunction with a topology file created during fabric design, you can associate switch names in the topology file with NodeGUIDs of the actual devices. This facilitates subsequent use of `opaswitchadmin` to configure the node descriptions for all switches according to the fabric design plan.

In a typical pure fat tree topology with externally managed switches as edge switches and managed switches as core switches, you can also manually specify proper distance by simply specifying 1 for the distance value of the switch next to the FastFabric node. Note that in such a topology, all other Edge switches are an equal length from the FastFabric node and a missing distance value causes them to be treated as having a distance value which is larger than any other found in the file. Therefore, the other switches would be rebooted first and the FastFabric node's switch would be rebooted last.

The GUID is used to select the switch and, on firmware update operations, the node description is written to the switch such that other FastFabric tools (such as `opasaquery` and `opareport`) can provide a more easily readable name for the switch. The node description can also be updated as part of switch basic configuration.

The hfi:port may be used to specify which local port (subnet) to use to access the switch. If this is omitted, all local ports specified are checked for the switch and the first port found to be able to access the switch is used to access it. Refer to [Descriptions of Command Line Tools](#) on page 128 for more information about how to specify the hfi:port value.

Files to be included may be specified using an `include` directive followed by a file name. File names specified should be absolute path names. If relative path names are used, they are searched for within the current directory first, then `/etc/opa`.

Comments may be placed on any single line by using a `#` to precede the comment. On lines with chassis or include directives, the `#` must be white space-separated from any preceding GUID, name, or included file name.

Intel recommends that a unique node description is specified for each switch. This name should follow typical naming rules and use the characters a-z, A-Z, 0-9, and underscore. No spaces are allowed in the node description. Additionally, names should not start with a digit.



For externally-managed switches, the node GUID can be found on a label on the bottom of the switch. Alternately, the node GUIDs for switches in the fabric can be found using a command such as:

```
opasaquery -t sw -o nodeguid
```

Note: The `opasaquery` command reports all switch node GUIDs, including those of managed chassis such as the Intel® Omni-Path Switch 100 Series. GUIDs for managed chassis cannot be used in the `switches` file.

3.3.2.3.2 Explicit Switch Names

When switches are explicitly specified using the `-N` option or the `SWITCHES` environment variable, a space-separated list of GUIDs (optionally with `hfi:port` and/or name) may be supplied. For example: `-N '0x00117500d9000138,i9k138 0x00117500d9000139,i9k139'`

3.3.2.4 Selection of Local Ports (Subnets)

Many commands permit a specific set of local Intel® Omni-Path Host Fabric Interface (HFI) ports to be used for fabric access. For example, `opareports`, `opafabricinfo`, `opaswitchadmin`, `opafabricanalysis`, and `opaallanalysis`. The default is to use the first active port. However, for Fabric Management nodes connected to more than one subnet, you must specify the local HFI and port so that the desired subnet is analyzed.

You can specify the local ports on which to operate using one of the following methods:

- On the command line, using the `-p` option.
- Using the environment variable `PORTS` to specify a space-separated list of ports. Useful when multiple commands are performed against the same small set of ports.
- Using the `-t` option or the `PORTS_FILE` environment variable to specify a file containing the set of ports. Useful for groups of ports that are used often. The file is located here: `/etc/opa/ports` by default. The file must list all local ports connected to unique subnets.

Within the tools, the options are considered in the following order:

1. `-p` option
2. `PORTS` environment variable
3. `-t` option
4. `PORTS_FILE` environment variable
5. `/etc/opa/ports` file
6. Default of the first active port on system. (0:0 port specification)

For example, if the `-p` option is used and the `PORTS_FILE` environment variable is also exported, the command operates only on ports specified using the `-p` option.

3.3.2.4.1 Port List Files

You can use the `-t` option or the `PORTS_FILE` environment variable to provide the name of a file containing the list of local HFI ports to use. The default is `/etc/opa/ports`.

It may be useful to create multiple files in `/etc/opa` representing different subsets of the ports. For example:

- `/etc/opa/ports-primary` - ports for which this node is primary
- `/etc/opa/ports-plane1` - port(s) for plane1 subnet
- `/etc/opa/ports` - list of all unique subnet ports

If a relative path is specified for the `-t` option or `PORTS_FILE` environment variable, the current directory is checked first, followed by `/etc/opa/`.

Port List File Format

Note: Intel® Omni-Path Host Fabric Interface has 1 port.

Sample port list file:

```
# this is a comment
1:1 # first port on 1st HFI
2:1 # first port on 2nd HFI
3:0 # first active port on 3rd HFI
include /etc/opa/ports-plane2 # included file
```

Each line of the port list file may specify a single port, a comment, or include another port list file.

Ports are specified as `hfi:port`. No spaces are permitted. The first HFI is 1 and the first port is 1. The value 0 for HFI or port has special meaning. The allowed formats are:

```
0:0 = 1st active port in system
0:y = port y within system
x:0 = 1st active port on HFI x
x:y = HFI x, port y
```

Files to be included may be specified using an `include` directive followed by a file name. File names specified should be absolute pathnames. If relative pathnames are used, they are searched for within the current directory first, then `/etc/opa`.

Comments may be placed on any line by using a `#` to precede the comment. On lines with a port or `include` directive, the `#` must be white space-separated from any preceding port or included filename.

3.3.2.4.2 Explicit Ports

When ports are explicitly specified using the `-p` option or the `PORTS` environment variable, a space-separated list of ports may be supplied. For example: `-p '1:1 2:1'`.



3.4 Configuration of IPoIB Name Mapping

The FastFabric tools support the concept of a management network and an IPoIB network. For some clusters, the management network will be a low-speed network such as 1 GB or 10 GB Ethernet. For other clusters, IPoIB may serve double duty as the host management network.

Note: When using IPoIB as the management network, the initial installation of Fabric software cannot be done using FastFabric.

The various FastFabric tools will translate from host names provided to and from IPoIB names as needed. This permits the given host names to be either management network or IPoIB network names.

- The default configuration file assumes that IPoIB host names are formed by adding a `-opa` suffix to the management network name.
- If a different suffix is desired, `FF_IPOIB_SUFFIX` can be changed.
- If IPoIB is also being used as the management network, `FF_IPOIB_SUFFIX` can be set to an empty string `""`.

The translation is driven by the following functions within `opafastfabric.conf`:

- `ff_host_basename` – Given a management network or IPoIB hostname, translate to management network name; should match `hostname -s`
- `ff_host_basename_to_ipoib` – Given a management network name, translate to IPoIB hostname

More complex mappings can be specified by implementing alternate algorithms for these functions.

Note: When managing a cluster where the IPoIB settings on the compute nodes are incompatible with the Fabric Management node, Intel recommends that you do not run IPoIB on the Fabric management nodes.

3.5 Sample Files

This section describes the files that are installed in the `/usr/share/opa/samples` directory, including the `opagentopology` sample script.

3.5.1 List of Files

This section describes the files that are installed in the `/usr/share/opa/samples` directory.

Configuration and Control Files

Files used by commands that analyze the fabric and perform multi-step initialization and verification operations. See [Configuration and Control for Chassis, Switch, and Host](#) for command details.

- `allhosts-sample`: all hosts in fabric, including management nodes. See `opahostadmin` for details.



- chassis-sample: all managed switches (reachable out-of-band). See [opagenchassis](#) and [opachassisadmin](#) for details.
- esm_chassis-sample: all managed switches running the embedded FM. See [opagenesmchassis](#) for details.
- hosts-sample: all hosts in the fabric. See [opahostadmin](#) for details.
- ports-sample: HFI and port configuration to use for communication with fabric. See [opagenswitches](#), [opagenchassis](#), [opagenesmchassis](#) and [opaswitchadmin](#) for details.
- switches-sample: all externally managed switches (reachable in-band). See [opagenswitches](#) and [opaswitchadmin](#) for details.

Packet Capture Files

Files for use with `opapacketcapture`:

- `filterFile.txt` - packet filter configuration.
- `triggerFile.txt` - trigger configuration for condition to terminate packet capture.

For more information on `opapacketcapture`, refer to the *Intel® Omni-Path Fabric Host Software User Guide*.

Topology Files

Files related to topology:

- `README.topology`
- `README.xlat_topology`
- `opagentopology` - script to generate topology file. See [opagentopology](#) for details.
- `opatopology_links.txt` - text CSV values for LinkSummary information.
- `opatopology_FIs.txt` - text CSV values for HFI Nodes information.
- `opatopology_SWs.txt` - text CSV values for Switch Nodes information.
- `opatopology_SMs.txt` - text CSV values for SM information.
- `linksum_swd06.csv`, `linksum_swd24.csv` - sample CSV configurations. See `README.xlat_topology` for explanation.
- `topology.xlsx`, `topology_cust.xlsx` - topology MS Excel files.
- `opamon.conf-sample`, `opamon.si.conf-sample` - port counter threshold files for use with [opareport](#).

Miscellaneous Files

- `hostverify.sh` - bash script to help verify configuration and performance of host nodes.
- `mac_to_dhcp` - script to help generate DHCP stanzas to append to `dhcpd.conf`. Uses host and MAC addresses.
- `opafastfabric.conf-sample` - configuration file for [opafastfabric](#). Used in `/etc/opa`.



3.5.2 opagentopology

Generates sample topology verification XML. Provides an example of using `opaxmlgenerate` and is a prototype for customization.

Uses CSV input files `opatopology_links.txt`, `opatopology_FIs.txt`, and `opatopology_SWs.txt` to generate LinkSummary, Node FIs, and Node SWs information respectively. These files are samples of what might be produced as part of translating a user custom file format into temporary intermediate CSV files.

LinkSummary information includes Link, Cable, and Port information. Note that `opagentopology` (not `opaxmlgenerate`) generates the XML version string as well as the `<Topology>` and `<LinkSummary>` lines. Also note that the indent level is at the default value of zero (0). The portions of the script that call `opaxmlgenerate` follow:

```
opaxmlgenerate -X /usr/share/opa/samples/opatopology_1.txt -d \; -h Link
-g Rate -g Rate_Int -g MTU -g LinkDetails -h Cable -g CableLength -g CableLabel
-g CableDetails -e Cable -h Port -g NodeGUID -g PortNum -g NodeDesc -g PortGUID
-g NodeType -g NodeType_Int -g PortDetails -e Port -h Port -g NodeGUID -g PortNum
-g NodeDesc -g PortGUID -g NodeType -g NodeType_Int -g PortDetails -e Port -e Link

opaxmlgenerate -X /usr/share/opa/samples/opatopology_2.txt -d \;
-h Node -g NodeGUID -g NodeDesc -g NodeDetails -g HostName -g NodeType
-g NodeType_Int -g NumPorts -e Node
```

opatopology_links.txt

This file can be found in `/usr/share/opa/samples/`. For brevity, this sample shows only two links. The second link shows an example of omitting some information. In the second line, the MTU, LinkDetails, and other fields are not present, which is indicated by an empty value for the field (no entry between the semicolon delimiters).

Note: The following example exceeds the available width of the page. For readability, a blank line is shown between lines to make it clear where the line ends. In an actual link file, no blank lines are used.

```
25g;2048;0;IO Server Link;11m;S4567;cable model 456;0x0002c9020020e004;1;bender
HFI-1;0x0002c9020020e004;FI;Some info about port;0x0011750007000df6;7;Switch 1234
Leaf 4;;SW;

25g;;0;;;0x0002c9020025a678;1;mindy2 HFI-1;;FI;;0x0011750007000e6d;4;Switch
2345 Leaf 5;;SW;
```

opatopology_FIs.txt

This file can be found in `/usr/share/opa/samples/`. For brevity, this sample shows only two nodes.

```
0x0002c9020020e004;bender HFI-1;More details about node
0x0002c9020025a678;mindy2 HFI-1;Node details
```



opatopology_SWs.txt

This file can be found in /usr/share/opa/samples/. For brevity, this sample shows only two nodes.

```
0x0011750007000df6;Switch 1234 Leaf 4;  
0x0011750007000e6d;Switch 2345 Leaf 5;
```

opatopology_SMs.txt

This file can be found in /usr/share/opa/samples/. For brevity, this sample shows only one node.

```
0x0002c9020025a678;1;mindy2 HFI-1;0x0011750007000e6d;FI;details about SM
```

Example

When run against the supplied topology input files, opagentopology produces:

```
<?xml version="1.0" encoding="utf-8" ?>  
<Topology>  
  <LinkSummary>  
    <Link>  
      <Rate>25g</Rate>  
      <MTU>2048</MTU>  
      <Internal>0</Internal>  
      <LinkDetails>IO Server Link</LinkDetails>  
      <Cable>  
        <CableLength>11m</CableLength>  
        <CableLabel>S4567</CableLabel>  
        <CableDetails>cable model 456</CableDetails>  
      </Cable>  
    <Port>  
      <NodeGUID>0x0002c9020020e004</NodeGUID>  
      <PortNum>1</PortNum>  
      <NodeDesc>bender HFI-1</NodeDesc>  
      <PortGUID>0x0002c9020020e004</PortGUID>  
      <NodeType>FI</NodeType>  
      <PortDetails>Some info about port</PortDetails>  
    </Port>  
    <Port>  
      <NodeGUID>0x0011750007000df6</NodeGUID>  
      <PortNum>7</PortNum>  
      <NodeDesc>Switch 1234 Leaf 4</NodeDesc>  
      <NodeType>SW</NodeType>  
    </Port>  
  </Link>  
  <Link>  
    <Rate>25g</Rate>  
    <Internal>0</Internal>  
    <Cable>  
      </Cable>  
    <Port>  
      <NodeGUID>0x0002c9020025a678</NodeGUID>  
      <PortNum>1</PortNum>  
      <NodeDesc>mindy2 HFI-1</NodeDesc>  
      <NodeType>FI</NodeType>  
    </Port>  
    <Port>  
      <NodeGUID>0x0011750007000e6d</NodeGUID>  
      <PortNum>4</PortNum>  
      <NodeDesc>Switch 2345 Leaf 5</NodeDesc>  
      <NodeType>SW</NodeType>  
    </Port>  
  </Link>  
</Topology>
```



```

</Link>
</LinkSummary>
<Nodes>
  <FIs>
    <Node>
      <NodeGUID>0x0002c9020020e004</NodeGUID>
      <NodeDesc>bender HFI-1</NodeDesc>
      <NodeDetails>More details about node</NodeDetails>
    </Node>
    <Node>
      <NodeGUID>0x0002c9020025a678</NodeGUID>
      <NodeDesc>mindy2 HFI-1</NodeDesc>
      <NodeDetails>Node details</NodeDetails>
    </Node>
  </FIs>
  <Switches>
    <Node>
      <NodeGUID>0x0011750007000df6</NodeGUID>
      <NodeDesc>Switch 1234 Leaf 4</NodeDesc>
    </Node>
    <Node>
      <NodeGUID>0x0011750007000e6d</NodeGUID>
      <NodeDesc>Switch 2345 Leaf 5</NodeDesc>
    </Node>
  </Switches>
  <SMs>
    <SM>
      <NodeGUID>0x0002c9020025a678</NodeGUID>
      <PortNum>1</PortNum>
      <NodeDesc>mindy2 HFI-1</NodeDesc>
      <PortGUID>0x0011750007000e6d</PortGUID>
      <NodeType>FI</NodeType>
      <SMDetails>details about SM</SMDetails>
    </SM>
  </SMs>
</Nodes>
</Topology>

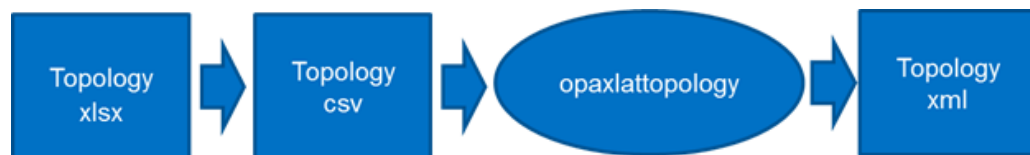
```

3.5.3 topology.xlsx Overview

This section describes the `topology.xlsx` file that is installed in the `/usr/share/opa/samples` directory.

A multi-step process generates the required `topology.xml` file, as shown in the following figure. You must edit the sample `topology.xlsx` file, save the edited information in CSV format, and run the `opaxlattopology` script, which produces the `topology.xml` file.

Figure 7. Topology Workflow



`topology.xlsx` provides a standard format for representing each external link in a cluster. Each link contains **Source**, **Destination**, and **Cable** fields with one link per row of the spreadsheet. The cells cannot contain commas.



Figure 8. topology.xlsx Example

A				B		C		D		E		F		G		H		I		J		K		L		M		N		O	
Standard-Format Topology Spread Sheet																															
Source																Destination										Cable					
Rack Group		Rack		Name		Name-2		Port		Type		Rack Group		Rack		Name		Name-2		Port		Type		Label		Length		Details			
row1	rack1	host01		gw				FI				row1	rack1	sw11						1		SW		host01 sw11P1		1m		Cable CU			
		host02								sw12						2				host02 sw12P2		1m		Cable CU							
		host03								sw13						3				host03 sw13P3		1m		Cable CU							
		host04								sw14						4				host04 sw14P4		1m		Cable CU							
	rack2	host05						FI					rack3	core1		L101A		1		CL		host05 core1L101P1		5m		Cable Fiber					
		host06								core1				L102B		2				host06 core1L102P2		5m		Cable Fiber							
		host07								core1				L103B		3				host07 core1L103P3		5m		Cable Fiber							
row2	rack1	host08										row2	rack3	core1		L104A		4				host08 core1L104P4		5m		Cable Fiber					
		sw11				19		SW		core1				L108A		9		CL		sw11P19 core1L108P9		1m		Cable CU							
		sw12				20				core1				L108A		10				sw12P20 core1L108P10		1m		Cable CU							
		sw13				21				core1				L108A		11				sw13P21 core1L108P11		1m		Cable CU							
	rack4	sw14				22				core1			L108A		12				sw14P22 core1L108P12		1m		Cable CU								
		host201		lsw				FI					rack4	sw21						1		SW		host201 sw21P1		1m		Cable CU			
		host202								sw22						2				host202 sw22P2		1m		Cable CU							
host203								sw23				3				host203 sw23P3		1m		Cable CU											
row3	rack5	host204										row3	rack6	sw24						4				host204 sw24P4		1m		Cable CU			
		host205						FI						core2		L101B		1		CL		host205 core2L101P1		5m		Cable Fiber					
		host206								core2				L102A		2				host206 core2L102P2		5m		Cable Fiber							
		host207								core2				L103A		3				host207 core2L103P3		5m		Cable Fiber							
	rack4	host208											rack6	core2		L104B		4				host208 core2L104P4		5m		Cable Fiber					
		sw21				19		SW		core2				L108B		9		CL		sw21P19 core2L108P9		1m		Cable CU							
		sw22				20				core2				L108B		10				sw22P20 core2L108P10		1m		Cable CU							
row4	rack5	sw23				21						row4	rack6	core2		L108B		11				sw23P21 core2L108P11		1m		Cable CU					
		sw24				22				core2				L108B		12				sw24P22 core2L108P12		1m		Cable CU							
		Xhost						FI						Xrow	Xrack	Xswitch				1		SW									
	Core Name: core1		Core Group: core1		Core Rack: rack3		Core Size: 288		Core Full: 1																						
	Core Name: core2		Core Group: core2		Core Rack: rack6		Core Size: 192		Core Full: 0																						
	Present Leafs		L305		L306		L110		L111		L112		L113																		
Desired Spines		S203		S205																											
Core Name: core1																															
Core Name: core2																															
Hosts		host01		host02																											



The first row must have a value. If the Type field is empty on any row, the script defaults the value to the closest previous value. The type values are:

- **Host:** FI for HFI adapter
- **Edge Switch:** SW
- **Core Leaf:** CL for Director switch core leaf module
- Cable (optional)

Cable values are optional and have no special syntax.

The **Cable** fields have the following columns:

- Label - Max characters = 57. (In release 10.2 and earlier, this field is limited to 20 characters.)
- Length
- Details

Core Full Statement

At the bottom of the `/usr/lib/opa/sample/topology.xlsx` file, there is a core full statement to indicate if the Intel® OP Director Class Switch 100 Series is fully populated with all spine and leaf modules installed. If there are multiple 6-slot or 24-slot Director switches in the fabric, each Director switch should have an entry in the `topology.xlsx` file as shown in the following table.

Table 5. Core Full Statement Definitions

Core Name:Core01	Core Group:row1	Core Rack:rack01	Core Size:1152	Core Full:0
Core Name:Core02	Core Group:row1	Core Rack:rack02	Core Size:1152	Core Full:0
Core Name: Specified in "Name" Column of topology.xlsx	Core Group: Specified in "Rack Group" Column of topology.xlsx	Core Rack: Specified in "Rack" Column of topology.xlsx	Core Size: Set to 1152 for 24 slot Director switch, 288 for 6 slot Director switch. Represents all internal (spine) links for fully populated Director.	0: Use for partially populated director. 1: Use for fully populated director.

Present Leaf Statement

This section should be used when the Core is partially populated (Core Full:0). Present Leaf Statement is used to specify the list of all present Leafs in the Core. This section can have multiple rows for each partially populated Core in the fabric.

There is no need to list the leaf names that have already been listed in the external link section as either Source Name or Destination Name.

Table 6. Present Leaf Statement Definitions

Core Name:core2	L105	L106	L110	L111	L112	L113
Core Name: Specified in "Name" Column of topology.xlsx	Name of Leaf Present	Name of Leaf Present	Name of Leaf Present			

Omitted Spine Statement

This section should also be used when the Core is partially populated (Core Full:0). Omitted Spine Statement is used to list all the missing Spines from the Core. This section can have multiple rows for each partially populated Core in the fabric.

Table 7. Omitted Spines Statement Definitions

Core Name:core2	S203	S205
Core Name: Specified in "Name" Column of topology.xlsx	Name of Missing Spine	Name of Missing Spine

SM Statement

This section can be used to list all the expected SMs in the fabric. The section can have multiple cells to indicate any number of SMs in a fabric.

Table 8. SM Statement Definition

SM	host01	host02	Name of Host running SM
----	--------	--------	-------------------------

Known Issue: Expected SM cannot be added with the detail levels, such as within a specific rack or group.

3.6 Configuration Files for FastFabric

The FastFabric configuration files allow you to configure and change the basic settings and variables for the fabric and each of its components. These files are pushed out across the network ensuring that each component is synchronized.

Configuration files are located under the `/etc/opa` directory.

Sample files are installed into `/usr/share/opa/samples` with the suffix `-sample`. These files show the defaults of the given release.

Note: Do not edit the sample files.

Configuration files are self-documented as shown in the example snippet below.

```
#!/bin/bash
# [ICS VERSION STRING: @(#) ./fastfabric/samples/opafastfabric.conf-sample
10_3_0_0_51 [09/20/16 23:52]
# This is a bash sourced config file which defines variables used in
# fast fabric tools. Command line arguments will override these settings.
```




```
# Assignments should be scripted such that this file does not override
# exported environment settings, as shown in the defaults below

if [ "$CONFIG_DIR" = "" ]
then
    if [ -d /etc ]
    then
        CONFIG_DIR=/etc
    else
        CONFIG_DIR=/etc
    fi
    export CONFIG_DIR
fi

# Override default location for HOSTS_FILE
export HOSTS_FILE=${HOSTS_FILE:-$CONFIG_DIR/opa/hosts}

# Override default location for CHASSIS_FILE
export CHASSIS_FILE=${CHASSIS_FILE:-$CONFIG_DIR/opa/chassis}
```

You can find more information about the various configuration variables in the "Environment Variables" section for the applicable CLI commands.

3.6.1 FastFabric Configuration File

The FastFabric configuration file allows you to view the default settings and modify the variables for most of the FastFabric command line options.

The file is located under `/etc/opa/opafastfabric.conf`.

A sample file is provided, and matches the internal defaults of the FastFabric tools.

Note: Command line arguments will override these settings.

Modifying the FastFabric Configuration File

- To modify the configuration file, refer to the following FastFabric TUI procedures:
 - [Editing the Configuration Files for Chassis Setup](#) on page 68
 - [Editing the Configuration Files for Externally-Managed Switch Setup](#) on page 85
 - [Editing the Configuration Files for Host Setup](#) on page 96
 - [Editing the Configuration Files for Host Verification](#) on page 110
- Adhere to the following requirements when editing the file:
 - The configuration file is a bash shell script that will be included by each tool. As such, the file should be implemented so that the environment variables defined prior to execution will not be altered.

The sample code below shows the bash syntax that allows only uninitialized variables to be overwritten by the configuration file:

```
var= "${var:-value}"
```

3.6.2 Ports List Configuration File

The Ports List configuration file allows you to specify the local HFI ports (i.e., subnets) that FastFabric will use in assorted commands for fabric access.

The file is located under `/etc/opa/ports`.

A sample file is provided, and matches the internal defaults of the FastFabric tools.

Alternate filenames may be specified in `opafastfabric.conf` using environment variables, or on the command line. Refer to the [Descriptions of Command Line Tools](#) on page 128 for more information.

Modifying the Ports List Configuration File

1. To modify the configuration file, refer to the following FastFabric TUI procedures:
 - [Editing the Configuration Files for Chassis Setup](#) on page 68
 - [Editing the Configuration Files for Externally-Managed Switch Setup](#) on page 85
 - [Editing the Configuration Files for Host Verification](#) on page 110
2. Adhere to the following requirements when editing the file:
 - Each line of the port list file may specify a single port, a comment, or another port list file to include.
 - Ports are specified as `hfi:port`. No spaces are permitted.

The first Host Fabric Interface Adapter is 1, and the first port is 1. The value 0 for Host Fabric Interface or port has special meaning. The allowed formats are shown in the example below.

```
# [ICS VERSION STRING: @(#) ./fastfabric/samples/ports-sample 10_3_0_0_51
[09/20/16 23:52]
# This file defines the local HFI ports to use to access the fabric(s)
#
# specify one line per HFI port of the form hfi:port such as:
#   0:0 = 1st active port in system
#   0:y = port y within system
#   x:0 = 1st active port on HFI x
#   x:y = HFI x, port y
# The first HFI in the system is 1. The first port on an HFI is 1.
0:0
```

- Files to be included may be specified using an `include` directive followed by a file name.

In general, specified file names should be absolute path names. If relative path names are used, they will be searched for within the current directory, then `/etc/opa`.

- Comments may be placed on any line by using a `"#"` to precede the comment. On lines with a port or `include` directive, the `"#"` must be white-space separated from any preceding port or included file name.

3.6.3 Chassis List Configuration Files

The Chassis List configuration files allow you to specify the Intel chassis that FastFabric will operate against for many operations.

The `opagenchassis` command can be used to help locate chassis in the fabric and generate a chassis file.

The files are located under `/etc/opa/chassis` and `/etc/opa/esm_chassis`.



A sample file is provided, and matches the internal defaults of the FastFabric tools.

Alternate filenames may be specified in `opafastfabric.conf`, using environment variables or on the command line. Refer to [Descriptions of Command Line Tools](#) on page 128 for more information.

Modifying the Chassis List Configuration Files

1. To modify the configuration files, refer to [Editing the Configuration Files for Chassis Setup](#) on page 68.
2. Adhere to the following requirements when editing the file:
 - Each line of the chassis list file may specify a single chassis, a comment, or another chassis list file to include.
 - Chassis are specified by the chassis management network IP address or by a resolvable TCP/IP name.
Note: Typically, names are used for readability.
 - If Ethernet is being used for the management network, specify the name corresponding to the ethernet IP address of the chassis.
 - Files to be included may be specified using an `include` directive followed by a file name.
In general, specified file names should be absolute path names. If relative path names are used, they will be searched for within the current directory, then `/etc/opa` directory.
 - Comments may be placed on any line by using a `"#"` to precede the comment.
On lines with chassis or `include` directives, the `#` must be white-space separated from any preceding name, IP address, or included filename.

3.6.3.1 Performing Operations Against a Selection of Slots Within a Chassis

Normally, operations are performed against the management card in the chassis. For operations such as `opacmdall`, the command is executed against the management interface for the given chassis. For more sophisticated operations, such as firmware update, a directory with firmware for each chassis card type can be supplied and all cards in the chassis will be updated with the appropriate firmware from that directory. However, in some cases it may be desirable to perform operations against a specific subset of cards within the chassis.

1. Augment the chassis IP address, a name within a chassis list, or a chassis file with a list of slot numbers on which to operate.

This is done in the form:

```
chassis:slot1,slot2,...
```

Note: There must be no spaces within the chassis name and/or slot list.

- This format is used by `opacmdall` and chassis firmware update.
It may be used anywhere a chassis name or IP address is valid, such as the `-H` option, the `CHASSIS` environment variable, or chassis list files.
- The slot number specified is ignored on some operations (such as `opapingall`).

- Only slots containing management cards may be specified with this format.
- For all Intel® Omni-Path Chassis 100 Series chassis, slot 0 is always an alias for the presently active management card for the chassis.

For the remainder of slot usages in the chassis, the `chassisQuery` command can be executed against a given chassis to identify which slots have management cards.

Note: For any operation, care should be taken that a given chassis is listed only once with all relevant slots as part of that single specification. This is important so that parallel operations do not cause conflicting concurrent operations against a given chassis.

3.6.4 Externally-Managed Switch List Configuration File

The Externally-Managed Switch List configuration file allows you to specify the externally-managed Intel switches that FastFabric will operate against for many operations.

The file is located under `/etc/opa/switches`.

A sample file is provided, and matches the internal defaults of the FastFabric tools.

Alternate file names may be specified in `opafastfabric.conf`, using environment variables or on the command line. Refer to the [Descriptions of Command Line Tools](#) on page 128 for more information.

Modifying the Externally-Managed Switch List Configuration File

1. To modify the configuration file, refer to [Editing the Configuration Files for Externally-Managed Switch Setup](#) on page 85.
2. Adhere to the following requirements when editing the file:
 - Each line of the switch list file may specify a single switch, a comment, or another switch list file to include.
 - Switches are specified in the comma-separated form:
`guid,nodeDesc,distance` where
 - `guid` – Node GUID of the switch optionally followed by a colon and `hfi:port`
 - `nodeDesc` – Optional node description should be programmed into the switch by FastFabric.
 It is recommended to supply a unique `nodeDesc` for each switch to simplify management of the cluster.
 - `distance` – Optional relative distance of the switch from the FastFabric node.
 This is used by reboot operations to first operate on switches furthest from the FastFabric node. Nodes without a distance specified will be treated as furthest. Refer to [Defining the Distance Value](#) on page 61.
 - The GUID will be used to select the switch and on firmware update operations, the node description will be written to the switch such that other FastFabric tools (such as `opasaquery` and `opareport`) can provide a more easily readable name for the switch.



The node description can also be updated as part of switch basic configuration.

- The `hfi:port` may be used to specify which local port (subnet) to use to access the switch.

If this is omitted, all local ports specified will be checked for the switch and the first port found to be able to access the switch will be used to access it. Refer to the [Descriptions of Command Line Tools](#) on page 128 for more information about how to specify an `hfi:port` value.

- Files to be included may be specified using an `include` directive followed by a file name. File names specified should generally be absolute path names. If relative path names are used, they will be searched for within the current directory then `/etc/opa`.
- Comments may be placed on any line by using a `"#"` to precede the comment.

On lines with `chassis` or `include` directives, the `"#"` must be white-space separated from any preceding GUID, name, or included file name.

- Intel recommends that a unique node description be specified for each switch. This name should follow typical naming rules and use the characters a-z, A-Z, 0-9, and underscore. No spaces are allowed in the node description. Additionally, names should not start with a digit.

- For externally-managed switches, the node GUID can be found on a label on the bottom of the switch.
- Alternately the node GUIDs for switches in the fabric can be found using a command such as:

```
opasaquery -t sw -o nodeguid
```

Note: The preceding command will report all switch node GUIDs, including those of managed chassis such as the Intel® Omni-Path Switch 100 Series switches. GUIDs for managed chassis cannot be specified for use in the `switches` file.

Defining the Distance Value

The `opagenswitches` command can be used to help locate externally-managed switches in the fabric and generate a `switches` file. The `opagenswitches` tool will by default provide the proper distance value relative to the FastFabric node from which it was run. This capability requires use of IBTA standard TraceRecord queries that are not supported by openSM, but can be supplied by the Intel® Omni-Path Fabric Suite Fabric Manager (FM). Alternatively the `opagenswitches -R` option can suppress generation of this field. Refer to [opagenswitches](#) on page 224 for more information.

In a typical pure fat tree topology, with externally-managed switches as edge switches and managed switches as core switches, you can also manually specify proper distance by specifying 1 for the distance value of the switch next to the FastFabric node. Note that in such a topology, all other switches are an equal length from the FastFabric node, and a missing distance value will cause them to be treated as having a distance value that is larger than any other found in the file. Therefore, the other switches would be rebooted first and the FastFabric node's switch would be rebooted last.

FastFabric is topology-aware when updating externally-managed switch firmware or resetting the switches. Switches furthest from the FastFabric node are updated or reset first, and then each switch, working toward the FastFabric node. This way, switches that are rebooted are not in the path between the FastFabric node and others that are being rebooted.

The ordering is controlled by an optional `distance` field in the `switches` file or the `switches` provided on the command line. The `distance` field indicates the relative distance from the FastFabric node for each switch. Any `switches` file entries that do not specify a distance value are treated as having a value larger than any others in the file. The `switches` file contains any one of the following formats per line:

- `nodeguid`
- `nodeguid,,distance`
- `nodeguid:hfi:port`
- `nodeguid:hfi:port,,distance`
- `nodeguid,nodename`
- `nodeguid,nodename,distance`
- `nodeguid:hfi:port,nodename`
- `nodeguid:hfi:port,nodename,distance`

3.6.5 Hosts List Configuration Files

The Hosts List configuration files allow you to specify the hosts that FastFabric will operate against for many operations.

The files are located under `/etc/opa/hosts` and `/etc/opa/allhosts`.

A sample file is provided, and matches the internal defaults of the FastFabric tools.

Alternate filenames may be specified in `opafastfabric.conf`, using environment variables or on the command line. Refer to [Descriptions of Command Line Tools](#) on page 128 for more information.

Modifying the Hosts List Configuration Files

1. To modify the configuration file, refer to the following FastFabric TUI procedures:
 - [Editing the Configuration Files for Host Setup](#) on page 96
 - [Editing the Configuration Files for Host Verification](#) on page 110
2. Adhere to the following requirements when editing the file:
 - Each line of the host list file may specify a single host, a comment or another host list file to include.
 - Hosts are specified by IP address or by a resolvable TCP/IP hostname.

Typically, hostnames are used for readability.

Also, some FastFabric tools will translate the supplied host names to IPoIB hostnames, in which case names are generally easier to translate than numeric IP addresses. Typically, management network host names are specified. However, if desired, IPoIB hostnames or IP addresses may be used. This can accelerate large file transfers and other operations.



- If Ethernet is being used for the management network, specify the hostname corresponding to the ethernet IP address.
- Files to be included may be specified using an `include` directive followed by a file name.

In general, specified file names should be absolute path names. If relative path names are used, they will be searched for within the current directory, then `/etc/opa` directory.

- Comments may be placed on any line by using a `#` to precede the comment.
On lines with hosts or include directives, the `#` must be white-space separated from any preceding host name, IP address, or included file name.

3.6.6 Port Statistics Thresholds Configuration File

The `opamon.conf` configuration file defines the thresholds for each port statistic. Error Counters are specified in absolute number of errors since last cleared. If the threshold for a given statistic is not defined or is set to 0 (disabled), the given statistic will not be checked. This file is use by the following commands:

- `opareport`

Note: When used by `opareport` or fabric health tools, the counts are absolute values and are applied against the counters as found in the system.

- `opafabrianalysis`
- `opalinkanalysis`
- `opaextractbadlinks`
- `opaextractstat`
- `opaextractstat2`
- `opaallanalysis`

The file is located under `/etc/opa/opamon.conf`.

A sample file is provided, and matches the internal defaults of the FastFabric tools.

3.6.7 Signal Integrity Thresholds Configuration File

The `opamon.si.conf` configuration file defines thresholds for port counter signal integrity. This file allows analysis for any non-zero error counters related to signal integrity (bad cables, etc.) and can be enabled by adding the `-c` option to many FastFabric tools including:

- `opareport`
- `opaextractbadlinks`
- `opaextractstat`
- `opaextractstat2,`
- `opalinkanalysis`
- `opacabletest`
- `opafabrianalysis`



The file is located under `/etc/opa/opamon.si.conf`.

A sample file is provided, and matches the internal defaults of the FastFabric tools.

3.6.8 Fabric Topology Input File

The Fabric Topology input file (`topology.0:0.xml`) allows you to specify the expected fabric topology and augmented fabric information (such as cable labels, types, lengths, SM details, node details, link details, etc.). If present, this file will be used by assorted FastFabric commands such as `opareports`, `opafabricanalysis`, and `opaallanalysis`.

The file is located under `/etc/opa/topology.0:0.xml`.

A sample file is provided, and matches the internal defaults of the FastFabric tools.

Alternate filenames may be specified in `opafastfabric.conf`, using environment variables or on the command line. Refer to the [Descriptions of Command Line Tools](#) on page 128 for more information.

Modifying the Fabric Topology Input File

Refer to [topology.xlsx Overview](#) on page 53 for an overview on how the topology file describes the fabric.

An example of the topology input file (in XML format) is shown below:

```
<?xml version="1.0" encoding="utf-8" ?>
<Report date="day mmm dd hh:mm:ss yyyy" unixtime="1446650124" options="--o
topology" >
<Nodes>
  <FIs>
    <ConnectedFICount>2</ConnectedFICount>
    <Node id="0x00117501007067a2">
      <NodeGUID>0x00117501007067a2</NodeGUID>
      <NodeType>FI</NodeType>
      <NodeType_Int>1</NodeType_Int>
      <NodeDesc>mindy2 hfi-0</NodeDesc>
      <Port id="0x00117501007067a2:1">
        <PortNum>1</PortNum>
        <LID>0x0001</LID>
        <PortGUID>0x00117501007067a2</PortGUID>
        <LinkWidthActive>4</LinkWidthActive>
        <LinkWidthActive_Int>8</LinkWidthActive_Int>
        <LinkSpeedActive>25Gb</LinkSpeedActive>
        <LinkSpeedActive_Int>2</LinkSpeedActive_Int>
      </Port>
    </Node>
    <Node id="0x00117501007067e6">
      <NodeGUID>0x00117501007067e6</NodeGUID>
      <NodeType>FI</NodeType>
      <NodeType_Int>1</NodeType_Int>
      <NodeDesc>mindy2 hfi-0</NodeDesc>
      <Port id="0x00117501007067e6:1">
        <PortNum>1</PortNum>
        <LID>0x0002</LID>
        <PortGUID>0x00117501007067e6</PortGUID>
        <LinkWidthActive>4</LinkWidthActive>
        <LinkWidthActive_Int>8</LinkWidthActive_Int>
        <LinkSpeedActive>25Gb</LinkSpeedActive>
        <LinkSpeedActive_Int>2</LinkSpeedActive_Int>
      </Port>
    </Node>
  </FIs>
</Nodes>
```




```

</Fis>
<Switches>
  <ConnectedSwitchCount>0</ConnectedSwitchCount>
</Switches>
<SMs>
  <ConnectedSMCount>1</ConnectedSMCount>
  <SM id="0x00117501007067a2:1">
    <SMState>Master</SMState>
    <SMState_Int>3</SMState_Int>
    <NodeGUID>0x00117501007067a2</NodeGUID>
    <NodeDesc>mindy2 hfi-0</NodeDesc>
    <PortNum>1</PortNum>
    <PortGUID>0x00117501007067a2</PortGUID>
    <NodeType>FI</NodeType>
    <NodeType_Int>1</NodeType_Int>
  </SM>
</SMs>
</Nodes>
<LinkSummary>
  <LinkCount>1</LinkCount>
  <Link id="0x00117501007067a2:1">
    <Rate>100g</Rate>
    <Rate_Int>16</Rate_Int>
    <Internal>0</Internal>
    <Port id="0x00117501007067a2:1">
      <NodeGUID>0x00117501007067a2</NodeGUID>
      <PortGUID>0x00117501007067a2</PortGUID>
      <PortNum>1</PortNum>
      <NodeType>FI</NodeType>
      <NodeType_Int>1</NodeType_Int>
      <NodeDesc>mindy2 hfi-0</NodeDesc>
    </Port>
    <Port id="0x00117501007067e6:1">
      <NodeGUID>0x00117501007067e6</NodeGUID>
      <PortGUID>0x00117501007067e6</PortGUID>
      <PortNum>1</PortNum>
      <NodeType>FI</NodeType>
      <NodeType_Int>1</NodeType_Int>
      <NodeDesc>mindy3 hfi-0</NodeDesc>
    </Port>
  </Link>
</LinkSummary>
</Report>

```



4.0 FastFabric TUI Menus

This section describes the FastFabric TUI menus used to perform common fabric management tasks.

The menus guide you through the administration process for each of the following components:

- [Managing the Chassis Configuration](#) on page 66
- [Managing the Switch Configuration](#) on page 83
- [Managing the Host Configuration](#) on page 95
- [Verifying the Host](#) on page 108

4.1 Managing the Chassis Configuration

The FastFabric OPA Chassis Setup/Admin menu allows you to set up and manage the Intel® Omni-Path Architecture managed switches.

1. Log in to the server as root.
2. At the command prompt, enter **opafastfabric**.

The Intel FastFabric OPA Tools menu is displayed.

```
Intel FastFabric OPA Tools
Version: X.X.X.X.X

  1) Chassis Setup/Admin
  2) Externally Managed Switch Setup/Admin
  3) Host Setup
  4) Host Verification/Admin
  5) Fabric Monitoring

X) Exit
```

3. Type 1.

The **FastFabric OPA Chassis Setup/Admin Menu** is displayed.

```
FastFabric OPA Chassis Setup/Admin Menu
Chassis File: /etc/opa/chassis

Setup:
0) Edit Config and Select/Edit Chassis File [ Skip ]
1) Verify Chassis via Ethernet Ping [ Skip ]
2) Update Chassis Firmware [ Skip ]
3) Set Up Chassis Basic Configuration [ Skip ]
4) Set Up Password-less ssh/scp [ Skip ]
5) Reboot Chassis [ Skip ]
6) Get Basic Chassis Configuration [ Skip ]
7) Configure Chassis Fabric Manager (FM) [ Skip ]
8) Update Chassis FM Security Files [ Skip ]
9) Get Chassis FM Security Files [ Skip ]
Admin:
a) Check OPA Fabric Status [ Skip ]
```



```

b) Control Chassis Fabric Manager (FM)      [ Skip ]
c) Generate All Chassis Problem Report Info [ Skip ]
d) Run a Command on All Chassis             [ Skip ]
Review:
e) View opachassisadmin Result Files        [ Skip ]

P) Perform the Selected Actions              N) Select None
X) Return to Previous Menu (or ESC)

```

4. Select one or more items by typing the alphanumeric character associated with the item to toggle the selection from Skip to Perform.
5. Type **P** to perform the operations.

Note: Each menu item will present you with prompts to complete the operation.

Table 9. FastFabric OPA Chassis Setup/Admin Menu Descriptions

Menu Item	Description
0) Edit the Configuration and Select/ Edit Chassis File	(Switch) Allows you to edit the following configuration files: <ul style="list-style-type: none"> • /etc/opa/chassis The chassis file lists the managed Intel switching chassis. • /etc/opa/ports The ports file lists the local HFI ports (for example, subnets) to be used to access the fabric for analysis. • /etc/opa/opafastfabric.conf The opafastfabric.conf file lists the default settings for most of the FastFabric command line options
1) Verify Chassis via Ethernet Ping	(Switch) Allows you to verify the existence of each selected chassis listed in the chassis file using a ping over the management network.
2) Update Chassis Firmware	(Switch) Allows you to verify and update the chassis firmware version.
3) Set Up Chassis Basic Configuration	(Switch) Prompts you for chassis configuration settings and then configures all the selected chassis accordingly.
4) Set Up Password-less ssh/scp	(Switch) Allows you to set up secure password-less SSH so that the Fabric Management Node can securely log into all the other chassis as admin through the management network without requiring a password.
5) Reboot Chassis	(Switch) Allows you to reboot each chassis listed in the /etc/opa/chassis file, ensuring that each chassis reboot is successful (as verified using ping over the management network).
6) Get Basic Chassis Configuration	(Switch) Allows you to retrieve basic information from the chassis, such as <ul style="list-style-type: none"> • Syslog • NTP configuration • Time zone information • Link Width • Link CRC Mode • Node description
7) Configure Chassis Fabric Manager (FM)	(Switch) Assists you in configuring the Intel® Omni-Path Fabric Suite Fabric Manager (FM) for any member of the Intel® Omni-Path Chassis 100 Series.
8) Update Chassis FM Security Files	(Switch) Allows you to verify and update the chassis security files.
9) Get Chassis FM Security Files	(Switch) Allows you to retrieve the chassis FM security files from the chassis.
a) Check OPA Fabric Status	(Switch or All) Allows you to check the state and error counts of all ports.
<i>continued...</i>	



Menu Item	Description
b) Control Chassis Fabric Manager (FM)	(Switch) Assists you in controlling the FM for any Intel® Omni-Path Chassis 100 Series chassis. NOTE: This operation is skipped for other chassis models.
c) Generate All Chassis Problem Report Info	(Switch) Allows you to collect configuration and status information from all selected chassis and generates a single *.tgz file that can be sent to a support representative.
d) Run a Command on All Chassis	(Switch) Allows you to execute a CLI command against all selected chassis.
e) View opachassisadmin Result Files	(All) Allows you to view the test.log and test.res files, which reflect the results from opachassisadmin operations (such as for updating Chassis Firmware or rebooting all chassis).

4.1.1 Editing the Configuration Files for Chassis Setup

(Switch) The **Edit Config and Select/Edit Chassis File** selection allows you to select and edit the chassis, ports, and FastFabric configuration files.

1. From the FastFabric OPA Chassis Setup/Admin menu, type **0**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Chassis Admin: Edit Config and Select/Edit Chassis File
Using vi (to select a different editor, export EDITOR).
You will now have a chance to edit/review the FastFabric Config File:
/etc/opa/opafastfabric.conf
The values in this file will control the default operation of the
FastFabric Tools. With the exception of the host file to use,
the values you specify for defaults will be used for all FastFabric
Operations performed via this menu system
Beware existing environment variables will override the values in this file.

About to: vi /etc/opa/opafastfabric.conf
Hit any key to continue (or ESC to abort)...
```

3. Press any key to open the opafastfabric.conf file or **ESC** to abort the operation.

Note: To get to subsequent configuration files, you must access each file.

The configuration file opens.

4. Review the settings:

- a. Review the `FF_CHASSIS_LOGIN_METHOD` and `FF_CHASSIS_ADMIN_PASSWORD`.

- FastFabric provides the opportunity to enter the chassis password interactively when needed. It is not necessary to place it within opafastfabric.conf. If the Intel chassis admin password is placed in opafastfabric.conf, change the opafastfabric.conf permissions to be 0x600 (root-only access).



- All versions of Intel® Omni-Path Chassis 100 Series firmware permit SSH keys to be configured within the chassis for secure password-less login. There is no need to configure a `FF_CHASSIS_ADMIN_PASSWORD`, and `FF_CHASSIS_LOGIN_METHOD` can be set to SSH (the default).
- b. Select the location for the result files from FastFabric with the `FF_RESULT_DIR` parameter. The default is the directory from which a given session of FastFabric is invoked. Alternatively, it can be set to a directory relative to your home directory. For example:

```
export FF_RESULT_DIR=${FF_RESULT_DIR:-$HOME/
fastfabric_results}
```

Refer to [FastFabric Configuration File](#) on page 57 for more information.

5. After saving and closing the `opafastfabric.conf` file in the editor, you will be given the opportunity to edit the `ports` file.

```
You will now have a chance to edit/review the FastFabric PORTS_FILE:
/etc/opa/ports
Some of the FastFabric operations which follow will use this file to
specify the local HFI ports to use to access the fabric(s) to operate on
Beware existing environment variables will override the values in this file.

About to: vi /etc/opa/ports
Hit any key to continue (or ESC to abort)...
```

6. Press any key to open the `ports` file or **ESC** to abort the operation. The configuration file opens.
7. Review the file:
 - For typical single-subnet clusters, the default of "0:0" may be used. This uses the first active port on the Management Node to access the fabric.
 - For configuring a cluster with multiple subnets, refer to *Intel® Omni-Path Fabric Software Installation Guide*.

Refer to [Ports List Configuration File](#) on page 57 for more information.

For further details about the Port List File format, refer to [Port List Files](#) on page 48.

8. After saving and closing the `ports` file in the editor, you will be given the opportunity to select the `chassis` file.

```
The FastFabric operations which follow will require a file
listing the chassis to operate on
Select Chassis File to Use/Edit [/etc/opa/chassis]:
```

9. Press **Enter** to edit the file.

```
About to: vi /etc/opa/chassis
Hit any key to continue (or ESC to abort)...
```

10. Press any key to open the `chassis` file or **ESC** to abort the operation. The configuration file opens.
11. Create the file with a list of the chassis names (the TCP/IP Ethernet management port names assigned) or IP addresses.



Note: Intel recommends you use chassis names.

Enter one chassis name or IP address per line. For example:

```
Chassis1  
Chassis2
```

Note: Do not list externally-managed switches in this file.

Refer to [Chassis List Configuration Files](#) on page 58 for more information.

For further details about the Chassis List File format, refer to [Chassis List Files](#) on page 43.

12. After saving and closing the `chassis` file in the editor, you will be given the opportunity to review and change the configuration files again.

```
Selected Chassis File: /etc/opa/chassis  
Do you want to edit/review/change the files? [y]:
```

13. Press **Enter** to review and edit the files or type **n** and press **Enter** to end the operation.

4.1.2 Verifying Chassis via Ethernet Ping

(Switch) The **Verify Chassis via Ethernet Ping** selection allows you to ping each selected chassis over the management network.

Associated CLI command: `opapingall`

1. From the FastFabric OPA Chassis Setup/Admin menu, type **1**.

The menu item changes from `[Skip]` to `[Perform]`.

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

The status is displayed.

```
Performing Chassis Admin: Verify Chassis via Ethernet Ping  
Executing: /usr/sbin/opapingall -C -p -F /etc/opa/chassis  
10.228.208.245: is alive  
Hit any key to continue (or ESC to abort)...
```

3. Press any key to continue or **ESC** to abort the operation.
4. If some chassis were not found, use the following list to assist in troubleshooting:
 - Is chassis powered on and booted?
 - Is chassis connected to management network?
 - Are chassis IP address and network settings consistent with DNS or `/etc/hosts`?
 - Is Management node connected to the management network?
 - Are Management node IP address and network settings correct?
 - Is management network itself up (including switches, routers, and others)?



- Is correct set of chassis listed in the chassis file? You may need to repeat the previous step to review and edit the file.

4.1.3 Updating the Chassis Firmware

(Switch) The **Update Chassis Firmware** selection allows you to verify and update the chassis firmware version as needed.

Associated CLI command: `opachassisadmin update`

PREREQUISITE: Before updating the firmware, refer to the *Intel® Omni-Path Fabric Switches Release Notes* for any prerequisites.

1. From the FastFabric OPA Chassis Setup/Admin menu, type 2.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Chassis Admin: Update Chassis Firmware
Multiple Firmware files and/or Directories may be space separated
Shell wildcards may be used
For Directories all .dpkg or .spkg files in the directory tree will be used
Enter Files/Directories to use (or none):
```

3. Specify the directory where the relevant firmware files have been stored and press **Enter**.

This can be the mount point of the CD or the directory to which the files were copied in a previous step.

```
Would you like to run the firmware now? [n]:
```

4. Type **y** and press **Enter**.

FastFabric ensures that all chassis are running the firmware level provided, and installs and/or reboots each chassis as needed.

If any chassis fails to be updated, use the **View opachassisadmin Result Files** option to review the result files from the update. Refer to [Interpreting the opahostadmin, opachassisadmin, and opaswitchadmin log files](#) on page 247 for more details.

4.1.4 Setting Up Chassis Basic Configuration

(Switch) The **Setup Chassis Basic Configuration** allows you to perform the typical chassis setup operations for all chassis.

Associated CLI command: `opachassisadmin configure`

Important: First-time installation instructions are found the *Intel® Omni-Path Fabric Software Installation Guide*.

1. From the FastFabric OPA Chassis Setup/Admin menu, type 3.

The menu item changes from [Skip] to [Perform].



Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.
3. For each prompt, provide the required information and press **Enter**:

Prompt	Description
Would you like to be prompted for chassis' password? [n]:	Allows you to enter a password for the chassis.
Do you wish to adjust syslog configuration settings? [y]:	Allows you to modify the syslog configuration settings.
Enter IP address for syslog server (or none):	Allows you to enter the IP address to the syslog server.
Do you wish to configure the syslog TCP/UDP port number? [n]:	Allows you to enter the TCP/UDP port number.
Do you wish to configure the syslog facility? [n]:	Allows you to configure the syslog facility.
Do you wish to configure an NTP server? [y]:	Allows you to configure an NTP server.
Enter IP address for NTP server (or none):	Allows you to set up the NTP server IP address.
Do you wish to configure timezone and DST information? [y]:	Allows you to set timezone and DST information from local server or manually.
Do you want to use the local timezone information from the local server? [y]:	Allows you to synchronize local timezone information with a local server.
Do you wish to configure the chassis link width? [n]:	Allows you to configure chassis link width.
Do you wish to configure OPA Node Desc to match ethernet chassis name? [y]:	Allows you to set OPA Node Desc to match ethernet chassis name. If you select yes [y], a reboot of all chassis devices is required in order to activate changes to the chassis OPA Node Desc.
Do you wish to configure the Link CRC Mode? [n]:	Allows you to configure link CRC mode.

After executing the prompts, the following is displayed:

```
Executing configure Test Suite (configure) Tue Oct 04 15:43:00 EDT 2016 ...
Executing TEST SUITE configure CASE (configure.10.228.208.245.emb.configure)
Chassis 10.228.208.245 configure ...
TEST SUITE configure CASE (configure.10.228.208.245.emb.configure) Chassis
10.228.208.245 configure PASSED
TEST SUITE configure: 1 Cases; 1 PASSED; 0 FAILED
TEST SUITE configure PASSED
Done configure Test Suite Tue Oct 04 15:43:02 EDT 2016

Hit any key to continue (or ESC to abort)...
```

4. Press any key or **ESC** to end the operation

4.1.5 Setting Up Password-less ssh/scp

(Switch) The **Set up Password-Less SSH/SCP** selection sets up secure password-less SSH, such that the Management Node can securely log into all the chassis as admin through the management network, without requiring a password.



Associated CLI command: `opasetupssh`

1. From the FastFabric OPA Chassis Setup/Admin menu, type **4**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Would you like to override the default Chassis password? [n]:
```

3. Choose one of the following actions:

- Press **Enter** to accept the default Chassis password.

```
Default Chassis password will be used to perform the setup
Executing: /usr/sbin/opasetupssh -p -C -F /etc/opa/chassis
Configuring 10.228.208.245...
Successfully processed: 1
Hit any key to continue (or ESC to abort)...
```

- Type **y** and press **Enter** to configure a new password for all chassis.

```
Executing: /usr/sbin/opasetupssh -p -S -C -F /etc/opa/chassis
Password for admin on all chassis:
```

- Enter the new password and press **Enter**.

4.1.6 Rebooting the Chassis

(Switch) The **Reboot Chassis** selection allows you to reboot all the selected chassis and ensures that they reboot fully (as verified through ping over the management network). When the chassis come back up following the reboot, they are running with all the new configuration settings.

Associated CLI command: `opachassisadmin`

1. From the FastFabric OPA Chassis Setup/Admin menu, type **5**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Chassis Admin: Reboot Chassis
Would you like to be prompted for chassis' password? [n]:
```

3. Press **Enter** to accept the default.

The chassis reboots.

4.1.7 Getting Basic Chassis Configuration

(Switch) The **Get Basic Chassis Configuration** selection allows you to retrieve basic information from chassis such as syslog, NTP configuration, time zone, node description, and other information.



Associated CLI command: `opachassisadmin`

1. From the FastFabric OPA Chassis Setup/Admin menu, type **6**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

The status is displayed as shown in the example below.

```
Performing Chassis Admin: Get basic Chassis configuration
Executing: /usr/sbin/opachassisadmin -F /etc/opa/chassis getconfig
Executing getconfig Test Suite (getconfig) day mmm dd hh:mm:ss timezone
YYYY ...
Executing TEST SUITE getconfig CASE (getconfig.xx.xx.xx.xx.getconfig) get
xx.xx.xx.xx ...
TEST SUITE getconfig CASE (getconfig.xx.xx.xx.xx.getconfig) get xx.xx.xx.xx
xx.xx.xx.xx:
    Firmware Active           : xx.xx.xx.xx
    Firmware Primary          : xx.xx.xx.xx
    Syslog Configuration      : Syslog host set to: 0.0.0.0 port 514 facility 22
    NTP                       : Configured to use the local clock
    Time Zone                 : Time zone offset has not been configured
    LinkWidth Support         : 4X
    Node Description           : Node_Name
    Link CRC Mode              : 48b_or_14b_or_16b
PASSED
TEST SUITE getconfig: 1 Cases; 1 PASSED
TEST SUITE getconfig PASSED
Done getconfig Test Suite day mmm dd hh:mm:ss timezone yyyy

Hit any key to continue (or ESC to abort)...
```

3. Press any key or **ESC** to end the operation.

4.1.8 Configuring Chassis Fabric Manager

(Switch) The **Configure Chassis Fabric Manager (FM)** selection allows you to configure the Fabric Manager for any Intel® Omni-Path Chassis 100 Series.

Associated CLI command: `opachassisadmin fmconfig`

Note: Also refer to the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide* for information on the command `config_generate`.

1. From the FastFabric OPA Chassis Setup/Admin menu, type **7**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Chassis Admin: Configure Chassis Fabric Manager (FM)
Enter FM Config file to use (or none or generate):
```

3. Type **generate**.



This performs the `config_generate` operation to guide you through selecting FM configuration options. See the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide* for more information about `config_generate`.

4. For each prompt, provide the required information and press **Enter**:

Prompt	Description
Anticipated maximum fabric size [100]:	Allows you to set the size of the subnet. FM resources and buffering are scaled to match the anticipated maximum size of the fabric. The size is specified in terms of the number of HFIs in a single fabric. For Embedded Fabric Manager, its recommended to use a value of 100 or less.
LMC value to use (there will be 2^LMC LIDs per HFI) [0]:	Allows you to set LMC value. LMC is used to control the number of LIDs per HFI. Multiple LIDs can be used to permit multiple routes between endpoints. This permits selected applications (such as MPIs using Intel(R) PSM) to optimize performance and/or resiliency by using dispersive routing. Default 0 assigns 1 LID per HFI.
Should Adaptive Routing be enabled [n]:	Allows you to enable or disable adaptive routing. Adaptive routing permits Intel® Omni-Path Architecture switches to dynamically adjust routing based on traffic patterns and hence reduce congestion and improve overall cluster performance and efficiency.
Node Appearance Log Message Threshold [100]:	Allows you to set the number of node appearance messages per sweep. When nodes appear or disappear from the fabric, a message is logged. A Threshold can be configured to limit the number of such messages per sweep. This Threshold can help to avoid excessive messages when fabric changes occur.
Name for FM instance 0 (Switch Port 0) [fm0]:	Allows you to set a name for each FM.
IPoIB rate for this FM (25g recommended):	Allows you to set the IPoIB rate for the FM. The FM configures the rate and MTU used for IPoIB multicast. The rate selected must be no greater than the rate of the slowest link in the fabric(s). The MTU selected must be no greater than the MTU of the smallest MTU link in the fabric(s). When selecting the rate and MTU, HFIs which won't run IPoIB can be ignored. However all Switches must be operating with at least the rate and MTU selected. Values are: 1) 25g 2) 50g 3) 75g 4) 100g
IPoIB MTU for this FM (2048 recommended):	Allows you to set the IPoIB MTU for the FM. Values are: 1) 2048 2) 4096
Do you want to configure a preferred primary or secondary FM [n]:	Allows you to configure primary or secondary FM. The FM supports failover. The FM to be preferred as the primary can be selected per FM instance. If no preferred primary is selected, FMs will negotiate based on HFI GUIDs.
Will this FM be the preferred primary [y]:	Allows you to set the current FM to be the primary FM. Values are: y - Primary n - Secondary
Should Sticky Failover be enabled [n]:	Allows you to set up the FM to support sticky failover.
continued...	



Prompt	Description
	The FM supports sticky failover. When enabled sticky failover will prevent a master FM from relinquishing control even if the preferred primary FM comes online. This can prevent situations where a bouncing preferred primary repeatedly takes over then fails.
Subnet Prefix upper bits for cluster [0xfe80000000000000]:	Allows you to set to the subnet prefix upper bits for the cluster. Each fabric in a cluster must have a unique 64 bit subnet prefix. The subnet prefix must be consistently configured on all FMs which manage the given fabric (e.g., on the primary and secondaries). To simplify input, you will be prompted for the upper bits for the cluster, then you will be prompted for the lower bits for each instance. The two values will be OR'ed together to form the subnet prefix for each fabric.
Subnet Prefix lower bits for FM instance 0 (fm0) (Switch Port 0) [0x0]:	Allows you to set to the subnet prefix lower bits for the cluster.
PM Sweep Interval in seconds [10]:	Allows you to set the PM sweep interval. The Fabric Manager includes a Performance Manager (PM) which can monitor the data movement and error counters in all devices. The PM monitors the counters periodically and computes the delta for counters. If the PM Sweep Interval is set to 0, no automatic sweeps occur. The PM Sweep Interval must be > 0 when using tools such as Fabric Performance Monitoring (<code>opatop</code>).
PM Error Threshold Exceeded Log Message Limit [10]:	Allows you to limit the number of PM error messages per sweep. When a port exceeds the threshold for Integrity, Security, or Routing errors, a message is logged. A Threshold can be configured to limit the number of such messages per sweep. This Threshold can help to avoid excessive messages.
How many concurrent clients are expected? [3]:	Allows you to set the number of concurrent PM clients to expect. The PM can retain some recent history in memory. This history can then be viewed in tools such as Fabric Performance Monitoring (<code>opatop</code>). For each historical sweep, both the topology and performance data is retained. Each dataset is referred to as an "image". The values will be adjusted based on the number of concurrent PA clients expected.
How many images should be retained? [10]:	Allows you to set the number of images to be retained for history. Images include: Pm.TotalImages Pm.FreezeFrameImages

After executing the prompts, the following is displayed:

```
Generated ./opafm.xml
To activate this configuration, ./opafm.xml must be transfered to
the chassis and the FM must be restarted.
The fastfabric TUI provides an easy way to do this.
You have selected to use: ./opafm.xml
Syntax Checking ./opafm.xml...
Executing: /usr/lib/opa/fm_tools/config_check -s -c ./opafm.xml
Valid FM Config file: ./opafm.xml

After push, the FM may be started/restarted
Would you like to restart the FM? [n]:
```

5. Enter **y**.

This causes the FM to be started with the new configuration.

```
Would you like to run the FM on slave MMs? [n]:
```

6. Refer to the following If/Then table:

If	Then
Your fabric has a single chassis running the Fabric Manager. You can run the Fabric Manager on the slave management module (MM). This causes the Fabric Manager to be started in the applicable chassis.	Enter y
Your fabric has multiple chassis running the Fabric Manager. Intel recommends you run Fabric Manager on the master management module. This causes the Fabric Manager to be started only on the master management module in the applicable chassis.	Enter n

```
There will be a disruption as FMs are restarted
Doing the operation in parallel (on multiple chassis) will finish the fastest
Doing it serially may reduce disruption
Would you like to do the operation in parallel? [y]:
```

7. Press **Enter** to select the default option: **y**.

Intel recommends doing the operation in parallel.

```
You have selected to perform the push and FM restart in parallel
Would you like to enable FM start at boot? [n]:
```

8. Enter **y**.

This causes the Fabric Manager to be started on all applicable chassis each time those chassis boot.

```
Would you like to enable FM start on slave MMs at boot? [n]:
```

9. Refer to the following If/Then table:

If	Then
Your fabric has a single chassis running the Fabric Manager. You can run the Fabric Manager on the slave management module. This causes the Fabric Manager to be started in the applicable chassis.	Enter y
Your fabric has multiple chassis running the Fabric Manager. Intel recommends you run Fabric Manager on the master management module. This causes the Fabric Manager to only be started on the master management module in the applicable chassis.	Enter n

System prompts:

```
Would you like to be prompted for chassis' password? [n]:
```

10. Press **Enter** to select the default **n** option.

```
Are you sure you want to proceed? [n]:
```

11. Enter **y**.



This updates the Fabric Manager.

```
Hit any key to continue (or ESC to abort)...
```

12. Press any key or **ESC** to end the operation.

4.1.9 Updating the Chassis FM Security Files

(Switch) The **Update Chassis FM Security Files** selection allows you to verify and update the chassis security files.

Associated CLI command: `opachassisadmin fmsecurityfiles`

Note: The FM security files are the private key, public key, and certificate files required by the FM, in order to support secure socket connection via OpenSSL. Refer to the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide* for instructions on the administration tasks required to support these files.

1. From the FastFabric OPA Chassis Setup/Admin menu, type **8**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Chassis Admin: Update Chassis FM Security Files
Multiple FM Security files and/or Directories may be space separated
Shell wildcards may be used
For Directories all .pem files in the directory tree will be used
Enter Files/Directories to use (or none):
```

3. Enter the files/directories and press **Enter**.

4. For each subsequent prompt, provide the required information and press **Enter**:

Prompts guide you through the options:

- `push` - Ensures given security files are pushed to each chassis.
- `restart` - After push, restart FM on master, stop on slave.
- `restartall` - After push, restart FM on all MM.

Additional options prompted for:

- Selection of security files or directory containing pem files
- Parallel versus serial update
- Chassis password (default is to have password in `fastfabric.conf` or to use password-less SSH)

If any chassis fails to be updated, use the **View opachassisadmin Results Files** option to review the result files from the update. Refer to [Interpreting the opahostadmin, opachassisadmin, and opaswitchadmin log files](#) on page 247 for more information.



4.1.10 Getting Chassis FM Security Files

(Switch) The **Get Chassis FM Security Files** selection allows you to run the `opachassisadmin fmgetsecurityfiles` command to retrieve the chassis FM security files from the chassis.

Associated CLI command: `opachassisadmin fmgetsecurityfiles`

1. From the FastFabric OPA Chassis Setup/Admin menu, type **9**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

The status is displayed.

```
Performing Chassis Admin: Get Chassis FM Security Files
Executing: /usr/sbin/opachassisadmin -F /etc/opa/chassis fmgetsecurityfiles
Executing get FM security files Test Suite (fmgetsecurityfiles) Tue Oct 04
16:27:55 EDT 2016 ...
Executing TEST SUITE get FM security files CASE (fmgetsecurityfiles.
10.228.208.245.fm_get_security_files) get 10.228.208.245 *.pem ./uploads/
10.228.208.245/ ...
TEST SUITE get FM security files CASE (fmgetsecurityfiles.
10.228.208.245.fm_get_security_files) get 10.228.208.245 *.pem ./uploads/
10.228.208.245/ PASSED
TEST SUITE get FM security files: 1 Cases; 1 PASSED; 0 FAILED
TEST SUITE get FM security files PASSED
Done get FM security files Test Suite Tue Oct 04 16:27:57 EDT 2016

Hit any key to continue (or ESC to abort)...
```

3. Press any key to complete this procedure.

4.1.11 Checking the OPA Fabric Status

The **Check OPA Fabric Status** selection allows you to analyze the links in a fabric.

Associated CLI commands: `opalinkanalysis`, `opareport`, and `opashowallports`

1. From the FastFabric OPA Chassis Setup/Admin menu, type **a**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

3. For each prompt, provide the required information and press **Enter**:

Prompt	Description
Would you like to perform fabric error analysis? [y]:	Allows you to start the analysis.
Clear error counters after generating report? [n]:	Allows you to clear the error counters after generating the report.
<i>continued...</i>	



Prompt	Description
Would you like to perform fabric link speed error analysis? [y]:	Allows you to analyze fabric link speed errors.
Check for links configured to run slower than supported? [n]:	Allows you to check for Links running slower than expected.
Check for links connected with mismatched speed potential? [n]:	Allows you to check for links connected with mismatched speed.
Enter filename for results [/root/linkanalysis.res]:	Allows you to enter a filename for the results or use the default file.

After executing the prompts, the following is displayed:

```
Executing: /usr/sbin/opalinkanalysis  errors slowlinks > /root/
linkanalysis.res 2>&l
About to: vi /root/linkanalysis.res
Hit any key to continue (or ESC to abort)...
```

4. Press any key to view the results file in the editor.

An example output is shown below.

```
Links running slower than expected Summary

Links running slower than expected:
1 of 1 Links Checked, 0 Errors found
-----
Links with errors >= threshold Summary

Configured Thresholds:
LinkQualityIndicator          4
UncorrectableErrors           1
LinkDowned                    1
RcvErrors                     1
ExcessiveBufferOverruns       1
FMConfigErrors                1
1 of 1 Links Checked, 0 Errors found
-----
```

4.1.12 Controlling Chassis Fabric Manager

(Switch) The **Control Chassis Fabric Manager (FM)** selection allows you to control the FM.

Associated CLI command: `opachassisadmin fmcontrol`

1. From the FastFabric OPA Chassis Setup/Admin menu, type **b**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.
3. For each prompt, provide the required information and press **Enter**:



Prompt	Description
Would you like to restart the FM? [n]:	Allows you to restart the FM.
Would you like to make sure the FM is not running? [n]:	Allows you to ensure that the FM is not running.
Would you like to make sure the FM is running? [n]:	Allows you to ensure that the FM is running.
Would you like to run FM on slave MMs? [n]:	Allows you to run FM on slave management modules.
Would you like to do the operation in parallel? [y]:	Allows you to perform operations in parallel (on multiple chassis). Doing the operation in parallel will finish the fastest.
Would you like to change FM boot state to enable FM start at boot? [n]:	Allows you to enable FM start on slave management modules at boot.
Would you like to change FM boot state to disable FM start at boot? [n]:	Allows you to disable FM start on slave management modules at boot.
Would you like to be prompted for chassis' password? [n]:	Allows you to be prompted for the chassis password.

After executing the prompts, the following is displayed:

```
Are you sure you want to proceed? [n]:
```

4. Select **y** to complete the operation.
5. When complete, press any key or **ESC** to end the operation.

4.1.13 Generating All Chassis Problem Report Information

(Switch) The **Generate All Chassis Problem Report Info** selection allows you to collect configuration and status information from all chassis and generate a single *.tgz file that can be sent to an Intel support representative.

Associated CLI command: `opacaptureall`

1. From the FastFabric OPA Chassis Setup/Admin menu, type **c**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Chassis Admin: Generate All Chassis Problem Report Info
Would you like to be prompted for chassis' password? [n]:
```

3. Press **Enter** to select the default or **y** to prompt for password.

`opacaptureall` is initiated and results gathered in `chassiscapture.all.tgz`.

```
Executing: /usr/sbin/opacaptureall -C -p -F /etc/opa/chassis
Running capture on all chassis ...
admin@10.228.208.245: capture: Command execution PASSED (Login): ...
Combining captured files into ./uploads/chassiscapture.all.tgz ...
Done.
Hit any key to continue (or ESC to abort)...
```



- When complete, press any key or **ESC** to end the operation.

4.1.14 Running a Command on All Chassis

(Switch) The **Run a Command on All Chassis** selection allows you to perform other operations on all chassis. Each time this is executed, a single chassis CLI command may be specified to be executed against all selected chassis. When using these commands, additional setup or verification of the chassis may be performed.

Associated CLI command: `opacmdall`

- From the FastFabric OPA Chassis Setup/Admin menu, type **d**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

- Type **P** to begin the operation.

```
Performing Chassis Admin: Run a Command on All Chassis
Would you like to be prompted for chassis' password? [n]:
```

- Press **Enter** to select the default or **y** to prompt for password.

```
Enter Command to run on all chassis (or none):
```

- Enter the CLI command to run and press **Enter**.

```
Run in parallel on all chassis? [y]:
```

- Select **y** (yes) or **n** (no) and press **Enter**.

```
About to run: /usr/sbin/opacmdall -C -F /etc/opa/chassis 'opashowmc -v'
Are you sure you want to proceed? [n]:
```

- Select **y** (yes) or **n** (no) and press **Enter**.

```
Executing: /usr/sbin/opacmdall -C -F /etc/opa/chassis 'opashowmc -v'
[admin@10.228.208.245]# opashowmc -v
admin@10.228.208.245: opashowmc -v: Command execution ...
Hit any key to continue (or ESC to abort)...
```

- When complete, press any key or **ESC** to end the operation.

4.1.15 Viewing opachassisadmin Result Files

(Switch) The **View opachassisadmin Result Files** selection allows you to open and view the `punchlist.csv`, `test.res`, and `test.log` files.

Note: For more information on the log files, refer to [Interpreting the opahostadmin, opachassisadmin, and opaswitchadmin log files](#) and [opachassisadmin Logging](#).

- From the FastFabric OPA Chassis Setup/Admin menu, type **e**.

The menu item changes from [Skip] to [Perform].



Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Chassis Admin: View opachassisadmin Result Files
Using vi (to select a different editor, export EDITOR).
About to: vi /root/punchlist.csv /root/test.res /root/test.log
Hit any key to continue (or ESC to abort)...
```

3. Press any key to view the opashassisadmin results files.
4. After reviewing and closing the log, you are prompted to remove the following files.

```
3 files to edit
Would you like to remove test.res test.log test_tmp* and save_tmp
in /root ? [n]:
```

5. Select **y** (yes) or **n** (no) and press **Enter**.
6. If you chose **y** in the step above, press any key or **ESC** to end the operation.

4.2 Managing the Switch Configuration

The FastFabric OPA Switch Setup/Admin menu allows you to set up and manage the Intel® Omni-Path externally-managed Edge Switches.

1. Log in to the server as root.
2. At the command prompt, enter **opafastfabric**.

The Intel FastFabric OPA Tools menu is displayed.

```
Intel FastFabric OPA Tools
Version: X.X.X.X.X

1) Chassis Setup/Admin
2) Externally Managed Switch Setup/Admin
3) Host Setup
4) Host Verification/Admin
5) Fabric Monitoring

X) Exit
```

3. Type **2**.

The **FastFabric OPA Switch Setup/Admin Menu** is displayed.

```
FastFabric OPA Switch Setup/Admin Menu
Externally Managed Switch File: /etc/opa/switches
Setup:
0) Edit Config and Select/Edit Switch File      [ Skip ]
1) Generate or Update Switch File               [ Skip ]
2) Test for Switch Presence                     [ Skip ]
3) Verify Switch Firmware                      [ Skip ]
4) Update Switch Firmware                      [ Skip ]
5) Set Up Switch Basic Configuration            [ Skip ]
6) Reboot Switch                              [ Skip ]
7) Report Switch Firmware & Hardware Info      [ Skip ]
8) Get Basic Switch Configuration              [ Skip ]
Admin:
9) Report Switch VPD Information                [ Skip ]
```



```
Review:
a) View opaswitchadmin Result Files          [ Skip  ]
P) Perform the Selected Actions              N) Select None
X) Return to Previous Menu (or ESC)
```

4. Select one or more items by typing the alphanumeric character associated with the item to toggle the selection from `Skip` to `Perform`.
5. Type `P` to perform the operations.

Note: Each menu item will present you with prompts to complete the operation.

Table 10. FastFabric OPA Switch Setup/Admin Menu Descriptions

Menu Item	Description
0) Edit Config and Select/Edit Switch File	(Switch) Allow you to edit the following configuration files: <ul style="list-style-type: none"> <code>/etc/opa/ports</code> The <code>ports</code> file lists the local HFI ports (for example, subnets) to be used to access the fabric for analysis. <code>/etc/opa/switches</code> The <code>switches</code> file lists the externally-managed Intel® Omni-Path Switch 100 Series switches. <code>/etc/opa/opafastfabric.conf</code> The <code>opafastfabric.conf</code> file lists the default settings for most of the FastFabric command line options.
1) Generate or Update Switch File	(Switch) Allows you to generate or update the <code>/etc/opa/switches</code> file based on the <code>/etc/opa/topology.%P.xml</code> files.
2) Test for Switch Presence	(Switch) Allows you to test for the presence of the selected switches in the fabric.
3) Verify Switch Firmware	(Switch) Allows you to verify the integrity of the present firmware in the switch. If this operation fails prior to any switch reboots or power-offs of the switch, perform <code>Update Switch Firmware</code> to correct the firmware in the switch.
4) Update Switch Firmware	(Switch) Allow you to update the switch firmware version and set the switch node name.
5) Set Up Switch Basic Configuration	(Switch) Prompts you for switch configuration settings and then configures all the selected Intel® Omni-Path Edge Switch 100 Series externally managed switches accordingly.
6) Reboot Switch	(Switch) The <code>Reboot Switch</code> selection runs the <code>opaswitchadmin reboot</code> command to reboot all the switches listed in the <code>/etc/opa/switches</code> file that was created in a previous step.
7) Report Switch Firmware & Hardware Info	(Switch) Provides you with a summary of the present state for all the selected switches.
8) Get Basic Switch Configuration	(Switch) Allows you to retrieve basic information from an externally-managed switch, such as: <ul style="list-style-type: none"> MTU VL Cap Credit Distribution Link Width Link Speed

continued...



Menu Item	Description
	<ul style="list-style-type: none"> Node description
9) Report Switch VPD Information	(Switch) Provides you with the Virtual Product Data (VPD) for all the selected switches. This information is useful for inventory and asset control as well as to provide details about the product to customer support.
a) View opaswitchadmin Result Files	Allows you to view the test.log and test.res files that reflect the results from opaswitchadmin runs (such as those for updating switch firmware, or for rebooting all switches per menu items above).

4.2.1 Editing the Configuration Files for Externally-Managed Switch Setup

(Switch) The **Edit Config and Select/Edit Switch File** selection allows you to select and edit the switches, ports, and FastFabric configuration files.

Note: Intel® Omni-Path Fabric Suite FastFabric is topology-aware when updating externally-managed switch firmware or resetting the switches. The update or restart starts at the switches farthest from the FastFabric node and then works toward the FastFabric node. This way, switches that are rebooted are not in the path between the FastFabric node and others that are being updated or reset.

- From the FastFabric OPA Switch Setup/Admin menu, type **0**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

- Type **P** to begin the operation.

```
Performing Switch Admin: Edit Config and Select/Edit Switch File
Using vi (to select a different editor, export EDITOR).
You will now have a chance to edit/review the FastFabric Config File:
/etc/opa/opafastfabric.conf
The values in this file will control the default operation of the
FastFabric Tools. With the exception of the host file to use,
the values you specify for defaults will be used for all FastFabric
Operations performed via this menu system
Beware existing environment variables will override the values in this file.
```

```
About to: vi /etc/opa/opafastfabric.conf
Hit any key to continue (or ESC to abort)...
```

- Press any key to open the opafastfabric.conf file or ESC to abort the operation.

Note: To get to subsequent configuration files, you must access each file.

The configuration file opens.

- Review the settings.

Refer to [FastFabric Configuration File](#) on page 57 for more information.

- After saving and closing the opafastfabric.conf file in the editor, you will be given the opportunity to edit the ports file.

```
You will now have a chance to edit/review the FastFabric PORTS_FILE:
/etc/opa/ports
Some of the FastFabric operations which follow will use this file to
```



```
specify the local HFI ports to use to access the fabric(s) to operate on
Beware existing environment variables will override the values in this file.
```

```
About to: vi /etc/opa/ports
Hit any key to continue (or ESC to abort)...
```

6. Press any key to open the `ports` file or **ESC** to abort the operation.

The configuration file opens.

7. Review the file:

- For typical single-subnet clusters, the default of "0:0" may be used. This uses the first active port on the Management Node to access all externally managed switches. .
- For configuring a cluster with multiple subnets, refer to *Intel® Omni-Path Fabric Software Installation Guide*.

Refer to [Ports List Configuration File](#) on page 57 for more information.

For further details about the Port List File format, refer to [Port List Files](#) on page 48.

8. After saving and closing the `ports` file in the editor, you will be given the opportunity to select the `switches` file.

```
The FastFabric operations which follow will require a file
listing the externally managed switches to operate on
Select Switch File to Use/Edit [/etc/opa/switches]:
```

9. Press **Enter** to edit the file.

```
About to: vi /etc/opa/switches
Hit any key to continue (or ESC to abort)...
```

10. Press any key to open the `switches` file or **ESC** to abort the operation.

The configuration file opens.

11. Create the file with a list of the switch node GUID and required switch names.

Enter one switch node GUID and required switch name per line. Do not use any spaces before or after the comma separating the switch node GUID and the name, as shown in this example:

```
0x00117500d9000138,edge1
0x00117500d9000139,edge2
```

Note: Do not list managed chassis in this file.



- Tips:*
- The **Generate or Update Switch File** menu item or `opagenswitches` may be used to generate a list of the externally-managed switches presently in the fabric. For example, when using the vi editor, the command `:r ! opagenswitches` may be used to add the output from this command to the file.
 - If needed, an SA query can be used to get a list of all switches. This includes both managed and externally-managed switches. Consequently, the output must be edited to leave only the Intel externally managed switches. An example SA query is:

```
opasaquery -t sw -o nodeguid
```

Refer to [Externally-Managed Switch List Configuration File](#) on page 60 for more information.

For further details about the (externally-managed) Switch List File format, refer to [Switch List Files](#) on page 45.

12. After saving and closing the `switches` file in the editor, you will be given the opportunity to review and change the configuration files again.

```
Selected Externally Managed Switch File: /etc/opa/switches
Do you want to edit/review/change the files? [y]:
```

13. Press **Enter** to review and edit the files or type **n** and press **Enter** to end the operation.

4.2.2 Generating or Updating Switch File

(Switch) The **Generate or Update Switch File** allows you to generate or update the switches file. It can also update switch names in the switches file by comparing the actual fabric to topology xml data.

Associated CLI command: `opagenswitches`

1. From the FastFabric OPA Switch Setup/Admin menu, type **1**.

The menu item changes from `[Skip]` to `[Perform]`.

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Switch Admin: Generate or Update Switch File
/etc/opa/switches empty
This file will be regenerated based on present fabric contents
Do you want to update switch names based on
/etc/opa/topology.%P.xml file(s)? [y]:
```

3. Press **Enter** to update the switch names per `topology.%P.xml` or **n** to generate the switch file.

```
About to run: /usr/sbin/opagenswitches -s -o /etc/opa/switches
Are you sure you want to proceed? [n]:
```

4. Press **Enter** for no or **y** to continue.



4.2.3 Testing for Switch Presence

(Switch) The **Test for Switch Presence** selection allows you to verify that each externally-managed switch specified in the switches file can be accessed by the Management Node through the Fabric Network.

Associated CLI command: `opaswitchadmin ping`

1. From the FastFabric OPA Switch Setup/Admin menu, type 2.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

The Test Suite report for switch ping is performed.

```
Performing Switch Admin: Test for Switch Presence
Executing: /usr/sbin/opaswitchadmin -L /etc/opa/switches ping
Executing report switch ping Test Suite (switchping) Thu Oct 06 10:02:00 EDT
2016 ...
Executing TEST SUITE report switch ping CASE (switchping.
10.228.208.247.i2c.extmgd.switchping) ping switch 10.228.208.247 ...
TEST SUITE report switch ping CASE (switchping.
10.228.208.247.i2c.extmgd.switchping) ping switch 10.228.208.247
...
TEST SUITE report switch ping: 1 Cases; 1 PASSED; 0 FAILED
TEST SUITE report switch ping PASSED
Done report switch ping Test Suite Thu Oct 06 10:04:01 EDT 2016

Hit any key to continue (or ESC to abort)...
```

3. Press any key or **ESC** to end the operation.
4. If some switches were not found, use the following list to assist in troubleshooting:
 - Is switch powered on and booted?
 - Is switch connected to Intel® Omni-Path Fabric?
 - Is Subnet Manager running?
 - Is Management Node's Port active?
 - Is Management Node connected to the correct Intel® Omni-Path Fabric?
 - Is FM Switch LED activated on the switch port to which the Fabric Management node is connected?

For more information, refer to the "FM Switch" section in the *Intel® Omni-Path Fabric Switches Hardware Installation Guide*.

- Is the correct set of switches listed in the switches file?

You may need to perform the [Generating or Updating Switch File](#) on page 87 operation to review and edit the file.

4.2.4 Verifying Switch Firmware

(Switch) The **Verify Switch Firmware** selection allows you to check that each externally-managed switch is operational and that its firmware is valid and accessible.

Associated CLI command: `opaswitchadmin fwverify`



1. From the FastFabric OPA Switch Setup/Admin menu, type **3**.
The menu item changes from [Skip] to [Perform].
Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.
2. Type **P** to begin the operation.
The TEST SUITE report switch fwverify is performed.
Note: The operation may take several minutes to complete.

```
Performing Switch Admin: Verify Switch Firmware
Executing: /usr/sbin/opaswitchadmin -L /etc/opa/switches fwverify
Executing report switch fwverify Test Suite (switchfwverify) Thu Oct 06
10:22:50 EDT 2016 ...
Executing TEST SUITE report switch fwverify CASE (switchfwverify.
10.228.208.247.i2c.extmgd.switchfwverify) retrieve switch 10.228.208.247 ...
TEST SUITE report switch fwverify CASE (switchfwverify.
10.228.208.247.i2c.extmgd.switchfwverify) retrieve switch 10.228.208.247
...
TEST SUITE report switch fwverify: 1 Cases; 1 PASSED; 0 FAILED
TEST SUITE report switch fwverify PASSED
Done report switch fwverify Test Suite Thu Oct 06 10:26:55 EDT 2016

Hit any key to continue (or ESC to abort)...
```

3. Press any key or **ESC** to end the operation.

4.2.5 Updating Switch Firmware

(Switch) The **Update Switch Firmware** selection allows you to update the switch firmware version and set the switch node name.

Associated CLI command: `opaswitchadmin upgrade`

Note: Refer to the *Intel® Omni-Path Fabric Switches Release Notes* to ensure that any prerequisites for the upgrade to the new firmware level have been met prior to performing the upgrade through FastFabric.

1. From the FastFabric OPA Switch Setup/Admin menu, type **4**.
The menu item changes from [Skip] to [Perform].
Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.
2. Type **P** to begin the operation.

```
Performing Switch Admin: Update Switch Firmware
Multiple Firmware files and/or Directories may be space separated
Shell wildcards may be used
For Directories all .emfw files in the directory tree will be used
Enter Files/Directories to use (or none):
```

3. Specify the directory where the relevant firmware files are located.

```
After upgrade, the switch may be optionally rebooted.
Would you like to reboot the switch after the update? [n]:
```



4. Type **y**.

```
The firmware on the switch will be checked, and if the running version is the
same as the version being used for the update, the update operation will be
skipped.
Would you like to override this check, and force the update to occur? [n]:
```

5. Press **Enter** to select default (n).

Note: The fabric is not yet operational.

```
You have selected to update the switch firmware and reboot.
There will be a disruption as switch or switches are rebooted.
Doing the operation in parallel (on multiple switches) will finish the
fastest.
Doing it serially may reduce disruption.
Would you like to do the operation in parallel? [y]:
```

Note: Because the Intel® Omni-Path Fabric itself is used to update externally-managed switches, updating multiple switches with the reboot option may disrupt parallel update operations. If there are not any selected externally-managed switches in the path from the Management Node to any other externally-managed switch, parallel operations can be established. For example, if the Management Node is connected directly to a core switch and externally-managed switches are only at the edges.

To control the order of the rebooting of externally-managed switches by FastFabric, refer to the `distance` option for the `switches` file in [Externally-Managed Switch List Configuration File](#) on page 60.

6. Press **Enter** for yes, or type **n** for no and press **Enter**.

Note: Be aware that non-parallel operation for a fabric with many externally managed switches can take a significant amount of time.

FastFabric updates the firmware on all switches and sets the node names, as per the `switches` file. Each switch is then rebooted.

If any switch fails to be updated, use the **View opaswitchadmin result files** option to review the result files from the update. Refer to [Interpreting the opahostadmin, opachassisadmin, and opaswitchadmin log files](#) on page 247 for more details.

4.2.6 Setting Up Switch Basic Configuration

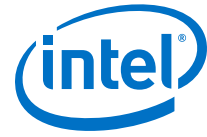
(Switch) The **Setup Switch Basic Configuration** selection allows you to perform typical switch setup operations using a wizard to configure all switches.

Associated CLI command: `opaswitchadmin configure`

1. From the FastFabric OPA Switch Setup/Admin menu, type 5.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.



2. Type **P** to begin the operation.

```
Performing Switch Admin: Set Up Switch Basic Configuration
Executing: /usr/sbin/opaswitchadmin -L /etc/opa/switches configure
Do you wish to configure the switch Link Width Options? [n]:
```

3. For each prompt, provide the required information and press **Enter**:

Prompt	Description
Do you wish to configure the switch Link Width Options? [n]:	<ul style="list-style-type: none"> Selecting n (no) causes the default switch Link Width Options to be used for all switches. If switches have previously been manually configured for different switch Link Width Options, this option keeps the previously configured switch Link Width Options. See the <i>Intel® Omni-Path Fabric Suite Fabric Manager GUI User Guide</i> for more information. Selecting y (yes) prompts for setting the switch link width supported setting for all ports on all switches. <p>Note: This operation is only applicable to Intel® Omni-Path Edge Switch 100 Series switches.</p>
Do you wish to configure the switch Node Description as it is set in the switches file? [n]:	<ul style="list-style-type: none"> Selecting n (no) causes the default switch Node Description on each switch to be used. If the switches have previously been manually configured for a customized switch Node Description, this option keeps the previously configured switch Node Descriptions. See the <i>Intel® Omni-Path Fabric Suite Fabric Manager GUI User Guide</i> for more information. Selecting y (yes) causes the Node Description on each switch to be updated as specified by the switches file. <p>Note: Only node descriptions on Intel® Omni-Path Edge Switch 100 Series switches can be changed in this step.</p>
Do you wish to configure the switch FM Enabled option? [n]:	<ul style="list-style-type: none"> Selecting n (no) causes all of the externally-managed switch ports to stay FM disabled. Selecting y (yes) prompts for setting the switch FM-enabled capability for all ports on all switches. <p>Setting it to enabled allows the FM to be connected to any port on any externally managed switch.</p> <p>If this is not desired, then select the default for the answer (disabled) and set the desired ports on the externally-managed switch to be FM-enabled using the FM switch.</p> <p>Refer to the "FM Switch" section in the <i>Intel® Omni-Path Fabric Switches Hardware Installation Guide</i> to set the port to FM enabled.</p> <p>Note: This operation is only applicable to Intel® Omni-Path Edge Switch 100 Series switches.</p>
Do you wish to configure the switch Link CRC Mode? [n]:	<ul style="list-style-type: none"> Selecting n (no) causes all of the externally managed switch ports Link CRC Mode to stay disabled. Selecting y (yes) prompts for setting the link CRC Mode for all ports on all switches. <p>Refer to the <i>Intel® Omni-Path Fabric Suite Fabric Manager GUI User Guide</i> for more information.</p>

After executing the prompts, you will be notified whether the operation passed or failed.

```
Executing configure Test Suite (configure) Thu Oct 06 16:21:04 EDT 2016 ...
Executing TEST SUITE configure CASE (configure.
10.228.208.247.i2c.extmgd.switchconfigure) configure switch 10.228.208.247 ...
TEST SUITE configure: 1 Cases; 1 PASSED; 0 FAILED
```



```
TEST SUITE configure PASSED
Done configure Test Suite Tue Oct 04 15:43:02 EDT 2016

Hit any key to continue (or ESC to abort)...
```

4. Press any key or **ESC** to end the operation

4.2.7 Rebooting the Switch

(Switch) The **Reboot Switch** selection allows you to reboot all switches, ensuring that all the configuration changes become effective and are discovered by the Intel® Omni-Path Fabric Suite Fabric Manager.

Associated CLI command: `opaswitchadmin reboot`

1. From the FastFabric OPA Switch Setup/Admin menu, type **6**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

Note: Reboot begins immediately.

```
Performing Switch Admin: Reboot Switch
Executing: /usr/sbin/opaswitchadmin -L /etc/opa/switches reboot
Executing reboot Test Suite (reboot) Thu Oct 06 16:37:30 EDT 2016 ...
Executing TEST SUITE reboot CASE (reboot.10.228.208.247.i2c.extmgd.reset)
reset switch 10.228.208.247 ...
```

4.2.8 Reporting Switch Firmware and Hardware Information

(Switch) The **Report Switch Firmware & Hardware Info** selection allows you to review reports on the firmware and hardware versions for each switch, along with other information for all of the externally-managed switches. Review the results against the expected models and firmware versions.

Associated CLI command: `opaswitchadmin info`

1. From the FastFabric OPA Switch Setup/Admin menu, type **7**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Switch Admin: Report Switch firmware & hardware info
Executing: /usr/sbin/opaswitchadmin -L /etc/opa/switches info
Executing report switch info Test Suite (switchinfo) day mmm dd hh:mm:ss
timezone yyyy ...
Executing TEST SUITE report switch info CASE (switchinfo).
0x00117500ff513121,Node_Name.i2c.extmgd.switchinfo)
  retrieve switch 0x00117500ff513121,Node_Name ...
TEST SUITE report switch info CASE (switchinfo).
0x00117500ff513121,Node_Name.i2c.extmgd.switchinfo)
  retrieve switch 0x00117500ff513121,Node_Name
0x00117500ff513121,hds1swb8171:
  F/W ver:xx.xx.xx.xx H/W ver:XXX H/W pt num:NNNNNN-NNN
  Fan status:Normal/Normal/Normal/Normal/Normal/Normal PS1 Status:N/A PS2
```



```
Status:ONLINE
PASSED
TEST SUITE report switch info: 1 Cases; 1 PASSED
TEST SUITE report switch info PASSED
Done report switch info Test Suite day mmm dd hh:mm:ss timezone yyyy
```

If any Intel® Omni-Path Switch 100 Series switches were purposely skipped, this operation should be repeated for those switches. In this case, Intel recommends that you create a separate file with a name other than switches.

4.2.9 Getting Basic Switch Configuration

(Switch) The **Get Basic Switch Configuration** selection allows you to view the switch configuration report for all of the ports.

Associated CLI command: `opaswitchadmin`

1. From the FastFabric OPA Switch Setup/Admin menu, type **8**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Switch Admin: Get basic Switch configuration
Executing: /usr/sbin/opaswitchadmin -L /etc/opa/switches getconfig
Executing report switch getconfig Test Suite (switchgetportconfig) day mmm dd
hh:mm:ss timezone yyyy ...
Executing TEST SUITE report switch getconfig CASE (switchgetportconfig.
0x00117500ff513121,hds1swb8171.i2c
.extmgd.switchgetportconfig) retrieve switch 0x00117500ff513121,Node_Name ...
TEST SUITE report switch getconfig CASE (switchgetportconfig.
0x00117500ff513121,Node_Name.i2c
.extmgd.switchgetportconfig) retrieve switch 0x00117500ff513121,Node_Name
Link Width           : 1,2,3,4
Link Speed           : 25Gb
FM Enabled           : Yes
Link CRC Mode        : 14-bit,16-bit,48-bit
vCU                  : 0
External Loopback Allowed : Yes
Node Description      : Node_Name

PASSED
TEST SUITE report switch getconfig: 1 Cases; 1 PASSED
TEST SUITE report switch getconfig PASSED
Done report switch getconfig Test Suite day mmm dd hh:mm:ss timezone yyyy
```

The results show the number of cases, how many of the cases passed, and how many of the cases failed. It also gives an overall summary of configuration and passed or failed.

4.2.10 Reporting Switch VPD Information

(Switch) The **Report Switch VPD Information** selection allows you to view the vital product data (VPD) for all of the nodes listed in `/etc/opa/switches`.

Associated CLI command: `opaswitchadmin hwvpd`

1. From the FastFabric OPA Switch Setup/Admin menu, type **9**.

The menu item changes from [Skip] to [Perform].



Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Switch Admin: Report Switch VPD information
Executing: /usr/sbin/opaswitchadmin -L /etc/opa/switches hwvpd
Executing report switch hwvpd Test Suite (switchhwvpd) day mmm dd hh:mm:ss
timezone yyyy ...
Executing TEST SUITE report switch hwvpd CASE (switchhwvpd.
0x00117500ff513121,Node_Name.i2c
.extmgd.switchhwvpd) retrieve switch 0x00117500ff513121,Node_Name ...
TEST SUITE report switch hwvpd CASE (switchhwvpd.
0x00117500ff513121,Node_Name.i2c
.extmgd.switchhwvpd) retrieve switch 0x00117500ff513121,Node_Name

0x00117500ff513121,hds1swb8171: H/W VPD serial number: USFU13150000D
0x00117500ff513121,hds1swb8171: H/W VPD part number : NNNNNN-NNN
0x00117500ff513121,hds1swb8171: H/W VPD model : 100SWE48QF2
0x00117500ff513121,hds1swb8171: H/W VPD h/w version : 004
0x00117500ff513121,hds1swb8171: H/W VPD manufacturer : Intel Corporation
0x00117500ff513121,hds1swb8171: H/W VPD prod desc : 100 OP Edge 48p Q7
forward 2PSU
0x00117500ff513121,hds1swb8171: H/W VPD mfg id : 001175
0x00117500ff513121,hds1swb8171: H/W VPD mfg date : m-dd-yyyy
0x00117500ff513121,hds1swb8171: H/W VPD mfg time : hh:mm
PASSED
TEST SUITE report switch hwvpd: 1 Cases; 1 PASSED
TEST SUITE report switch hwvpd PASSED
Done report switch hwvpd Test Suite day mmm dd hh:mm:ss timezone yyyy
```

4.2.11 Viewing opaswitchadmin Result Files

(Switch) The **View opaswitchadmin Result Files** selection allows you to open and view the punchlist.csv, test.res, and test.log files.

Note: For more information on the log files, refer to [Interpreting the opahostadmin, opachassisadmin, and opaswitchadmin log files](#) and [opaswitchadmin Details](#).

1. From the FastFabric OPA Switch Setup/Admin menu, type **a**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Switch Admin: View opaswitchadmin Result Files
Using vi (to select a different editor, export EDITOR).
About to: vi /root/punchlist.csv /root/test.res /root/test.log
Hit any key to continue (or ESC to abort)...
```

3. Press any key to view opaswitchadmin results files.
4. After reviewing and closing the log, you are prompted to remove the following files.

```
3 files to edit
Would you like to remove test.res test.log test_tmp* and save_tmp
in /root ? [n]:
```



5. Select **y** (yes) or **n** (no) and press **Enter**.
6. If you chose **y** in the step above, press any key or **ESC** to end the operation.

4.3 Managing the Host Configuration

The FastFabric OPA Host Setup menu allows you to set up and install the Fabric software on all the hosts.

To access up the FastFabric OPA Host Setup Menu, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opafastfabric**.

The Intel FastFabric OPA Tools menu is displayed.

```
Intel FastFabric OPA Tools
Version: X.X.X.X.X

1) Chassis Setup/Admin
2) Externally Managed Switch Setup/Admin
3) Host Setup
4) Host Verification/Admin
5) Fabric Monitoring

X) Exit
```

3. Type **3**.

The FastFabric OPA Host Setup menu is displayed.

```
FastFabric OPA Host Setup Menu
Host File: /etc/opa/hosts
Setup:
0) Edit Config and Select/Edit Host File      [ Skip ]
1) Verify Hosts Pingable                     [ Skip ]
2) Set Up Password-Less SSH/SCP               [ Skip ]
3) Copy /etc/hosts to All Hosts               [ Skip ]
4) Show uname -a for All Hosts                [ Skip ]
5) Install/Upgrade OPA Software               [ Skip ]
6) Configure IPoIB IP Address                 [ Skip ]
7) Build Test Apps and Copy to Hosts          [ Skip ]
8) Reboot Hosts                              [ Skip ]
Admin:
9) Refresh SSH Known Hosts                   [ Skip ]
a) Rebuild MPI Library and Tools              [ Skip ]
b) Run a Command on All Hosts                 [ Skip ]
c) Copy a File to All Hosts                   [ Skip ]
Review:
d) View opahostadmin Result Files             [ Skip ]

P) Perform the Selected Actions               N) Select None
X) Return to Previous Menu (or ESC)
```

4. Select one or more items by typing the alphanumeric character associated with the item to toggle the selection from **Skip** to **Perform**.
5. Type **P** to perform the operations.

Note: Each menu item will present you with prompts to complete the operation.

Table 11. FastFabric OPA Host Setup Menu Descriptions

Menu Item	Description
0) Edit Config and Select/Edit Host File	<p>Allows you to edit the following configuration files:</p> <ul style="list-style-type: none"> <code>/etc/opa/hosts</code> The <code>hosts</code> file lists the names of the hosts in a cluster except the FastFabric toolset node. <code>/etc/opa/opafastfabric.conf</code> The <code>opafastfabric.conf</code> file lists the default settings for most of the FastFabric command line options <p>NOTE: The <code>hosts</code> file selected and created using this menu should not list the FastFabric host itself.</p>
1) Verify Hosts Pingable	Allows you to ping all the hosts listed through the Management Network.
2) Set Up Password-Less SSH/SCP	(Linux) Allows you to set up secure password-less SSH such that the Fabric Management Node can securely log into all the other hosts as root through the management network without requiring a password.
3) Copy <code>/etc/hosts</code> to All Hosts	<p>(Linux) Allow you to copy the <code>/etc/hosts</code> file on this host to all the other selected hosts.</p> <p>NOTE: This is not necessary when using a DNS server to resolve host names for the cluster.</p>
4) Show <code>uname -a</code> for All Hosts	(Linux) Allows you to view the OS version on all the hosts. In typical clusters, all hosts are running the same OS and kernel version.
5) Install/Upgrade OPA Software	(Host) Allows you to install the Intel® OPA software on all the hosts.
6) Configure IPoIB IP Address	(Host) Allow you to create the <code>ifcfg-ib0</code> files on each host. The file will be created with a statically-assigned IPv4 address.
7) Build Test Apps and Copy to Hosts	(Host) Allows you to build the MPI sample benchmarks on the Fabric Management Node and copy the resulting object files to all the hosts.
8) Reboot Hosts	(Linux) Allows you to reboot all the selected hosts and to ensure they reboot fully (as verified using ping over the management network). When the hosts come back up, they will be running the software installed.
9) Refresh SSH Known Hosts	<p>(Linux) Allows you to refresh the ssh known hosts list on this server for the Management Network.</p> <p>This option needs to be executed after the configuration of IPoIB interfaces on any or all hosts.</p> <p>In addition, this option may be used to update security for this host to complete installation of the hosts or if hosts are installed, replaced, reinstalled, renamed, or repaired.</p>
a) Rebuild MPI Library and Tools	(Host) Allows you to rebuild the MPI Library and related tools (such as <code>mpirun</code>).
b) Run a Command on All Hosts	<p>(Linux) Allows you to run a command on all hosts.</p> <p>NOTE: A Linux shell command (or sequence of commands separated by semicolons) may be specified to be executed against all selected hosts.</p>
c) Copy a File to All Hosts	<p>(Linux) Allow you to copy a file to all hosts.</p> <p>NOTE: A file on the local host may be specified to be copied to all selected hosts.</p>
d) View <code>opahostadmin</code> Result Files	Allows you to view the <code>test.log</code> and <code>test.res</code> files that reflect the results from <code>opahostadmin</code> runs (such as for installing software or rebooting all hosts per menu items above).

4.3.1 Editing the Configuration Files for Host Setup

The **Edit Config and Select/Edit Host File** selection allows you to edit the hosts and FastFabric configuration files.



1. From the FastFabric OPA Host Setup menu, type 0.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Setup: Edit Config and Select/Edit Host File
Using vi (to select a different editor, export EDITOR).
You will now have a chance to edit/review the FastFabric Config File:
/etc/opa/opafastfabric.conf
The values in this file will control the default operation of the
FastFabric Tools. With the exception of the host file to use,
the values you specify for defaults will be used for all FastFabric
Operations performed via this menu system
Beware existing environment variables will override the values in this file.

About to: vi /etc/opa/opafastfabric.conf
Hit any key to continue (or ESC to abort)...
```

3. Press any key to open the opafastfabric.conf file or **ESC** to abort the operation.

Note: To get to subsequent configuration files, you must access each file.

The configuration file opens.

4. Review the settings.

Refer to [FastFabric Configuration File](#) on page 57 for more information.

5. After saving and closing the opafastfabric.conf file in the editor, you will be given the opportunity to edit the hosts file.

```
The FastFabric operations which follow will require a file
listing the hosts to operate on
You should select a file which OMITs this host
Select Host File to Use/Edit [/etc/opa/hosts]:
```

6. Press any key to open the hosts file or **ESC** to abort the operation.

The configuration file opens.

Refer to [Hosts List Configuration Files](#) on page 62 for more information.

For further details about the Host Lists file format, refer to [Host List Files](#) on page 42.

7. Create the file with a list of the hosts names (the TCP/IP management network names), except the Management Node from which FastFabric is presently being run.

Enter one host's name per line. For example:

```
host1
host2
```

Note: Do not list the Management Node itself (the node where FastFabric is currently running).

If additional Management Nodes are to be used, they may be listed at this time, and FastFabric can aid in their initial installation and verification.

8. After saving and closing the `hosts` file in the editor, you will be given the opportunity to review and change the configuration files again.

```
Selected Host File: /etc/opa/hosts
Do you want to edit/review/change the files? [y]:
```

9. Press **Enter** to review and edit the files or type **n** and press **Enter** to end the operation.

4.3.2 Verifying Hosts are Pingable

(All) The **Verify Hosts Pingable** selection pings each selected host over the management network.

Associated CLI command: `opapingall`

1. From the FastFabric OPA Host Setup menu, type **1**.

The menu item changes from `[Skip]` to `[Perform]`.

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Setup: Verify Hosts Pingable
Would you like to verify hosts are ssh-able? [n]:
```

3. Press **Enter** to select the default (n) or enter **y** and press **Enter**.

The status is displayed.

```
Executing: /usr/sbin/opafindgood -A -Q -R -f /etc/opa/hosts
1 hosts will be checked
1 hosts are pingable (alive)
1 hosts are alive (good)
0 hosts are bad (bad)
Bad hosts have been added to /root/punchlist.csv
Hit any key to continue (or ESC to abort)...
```

4. If some hosts were not found, press **ESC** and use the following list to assist in troubleshooting:
 - Host powered on and booted?
 - Host connected to management network?
 - Host management network IP address and network settings consistent with DNS or `/etc/hosts`?
 - Management node connected to the management network?
 - Management node IP address and network settings correct?
 - Management network itself up (including switches, routers, and others)?
 - Correct set of hosts listed in the hosts file? You may need to repeat the previous step to review and edit the file.

After fixing the issues, restart this task.



5. If all hosts were found, press any key to continue.

```
Would you like to now use /etc/opa/good as Host File? [y]:
```

6. Press **Enter** to select the default (y) or enter **n** and press **Enter** to end the operation.

4.3.3 Setting Up Password-Less SSH/SCP

(Linux) The **Setup Password-less ssh/scp** selection allows you to set up secure password-less SSH (root password) such that the Management Node can securely log in to all the other hosts as root through the management network without requiring a password.

Note: Password-less SSH is required by Intel® Omni-Path Fabric Suite FastFabric, MPI test applications, and most versions of MPI (including OpenMPI and MVAPICH2).

Associated CLI command: `opasetupssh`

1. From the FastFabric OPA Host Setup menu, type 2.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Setup: Set Up Password-Less SSH/SCP
Executing: /usr/sbin/opasetupssh -S -p -i '' -f /etc/opa/hosts
Password for root on all hosts:
```

3. Type the password for root on all hosts and press **Enter**.

4.3.4 Copying /etc/hosts to All Hosts

(Linux) The **Copy /etc/hosts to all hosts** selection allows you to copy the `/etc/hosts` file on this host to all the other selected hosts.

Typically, `/etc/resolv.conf` is set up as part of OS installation for each host. However, if `/etc/resolv.conf` was not set up on all the hosts during OS installation, the **Copy a File to All Hosts** operation could be used at this time to copy `/etc/resolv.conf` from the Management Node to all the other nodes.

Note: If DNS is being used, this task is not required.

Associated CLI command: `opascpall`

1. From the FastFabric OPA Host Setup menu, type 3.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.



2. Type **P** to begin the operation.

```
Performing Host Setup: Copy /etc/hosts to All Hosts
Executing: /usr/sbin/opascpall -p -f /etc/opa/hosts /etc/hosts /etc/hosts
scp -q /etc/hosts root@[phgppriv11]:/etc/hosts
Hit any key to continue (or ESC to abort)...
```

3. Press any key to continue or **ESC** and press **y** to cancel the operation.

4.3.5 Showing uname -a for All Hosts

(Linux) The **Show uname -a for All Hosts** selection allows you to show the OS version on all the hosts.

Associated CLI command: `opacmdall`

1. From the FastFabric OPA Host Setup menu, type **4**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Setup: Show uname -a for All Hosts
Executing: /usr/sbin/opacmdall -T 60 -f /etc/opa/hosts 'uname -a'
[root@phgppriv11]# uname -a
Linux phgppriv11.ph.intel.com 3.10.0-123.el7.x86_64 #1 SMP Mon May 5 11:16:57
EDT 2014 x86_64 x86_64 x86_64 GNU/Linux
Hit any key to continue (or ESC to abort)...
```

3. Press any key or **ESC** to end the operation.
4. Review the results to verify all the hosts have the expected OS version.
 - In typical clusters, all hosts are running the same OS and kernel version.
 - If any hosts are identified with an incorrect OS version, the OS on those hosts should be corrected at this time.

After the OS versions have been corrected, perform [Copying a File to All Hosts](#) on page 107.

4.3.6 Installing/Upgrading OPA Software

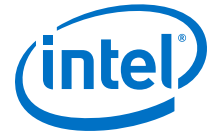
(Host) The **Install/Upgrade OPA Software** selection allows you to install or upgrade the Intel® Omni-Path Fabric Host Software on all the hosts. By default, it looks in the current directory for the `IntelOPA-[Basic|IFS].DISTRO.VERSION.tgz` file. If the file is not found in the current directory, the installer application prompts for a directory name where this file can be found.

Associated CLI command: `opahostadmin`, options: `load` and `update`

Note: Refer to the *Intel® Omni-Path Fabric Software Installation Guide* for performing first-time installations and upgrades.

1. From the FastFabric OPA Host Setup menu, type **5**.

The menu item changes from [Skip] to [Perform].



Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.
3. For each prompt, provide the required information and press **Enter**:

Prompt	Description
Enter Directory to get IntelOPA-[Basic IFS].DISTRO.VERSION.tgz from (or none):	Allows you to enter the directory to the software. If none , you will be prompted whether you want to proceed: <ul style="list-style-type: none"> • Select y to continue. • Select n to abort.
Do you want to use ./IntelOPA-[Basic IFS].DISTRO.VERSION.tgz? [y]:	Allows you to select the tgz that is required for the installation or upgrade.
Would you like to do a fresh [i]ninstall, an [u]pgrade or [s]kip this step? [u]:	<ul style="list-style-type: none"> • Select i to install software. • Select u to upgrade software. • Select s to skip this step.
Are you sure you want to proceed? [n]:	

After executing the prompts, the following is displayed:

```
/usr/sbin/opahostadmin -f /etc/opa/hosts -d . load
Executing load Test Suite (load) Day Mth DD HH:MM:SS timezone yyyy ...
.
.
.
Hit any key to continue (or ESC to abort)...
```

Note: If any hosts fail to be installed, you will see results as shown in the following example:

```
TEST SUITE load: 1 Cases; 0 PASSED; 1 FAILED
TEST SUITE load FAILED
```

Use the [Viewing opahostadmin Result Files](#) on page 126 option to review the result files from the update. Refer to [Interpreting the opahostadmin, opachassisadmin, and opaswitchadmin log files](#) on page 247 for more details.

4. Press any key or **ESC** to end the operation.

4.3.7 Configuring IPoIB IP Address

(Host) The **Configure IPoIB IP Address** selection allows you to create the `ifcfg-ib0` files on each host. The file is created with a statically-assigned IPv4 address. The IPoIB IP address for each host is determined by the resolver (Linux* host command). If not found through the resolver, the `/etc/hosts` file for the given host is checked.

Associated CLI command: `opahostadmin configipoib`

1. From the FastFabric OPA Host Setup menu, type **6**.

The menu item changes from `[Skip]` to `[Perform]`.

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.



2. Type **P** to begin the operation.

```
Performing Host Setup: Configure IPoIB IP Address
Executing: /usr/sbin/opahostadmin -f /etc/opa/hosts configipoib
Executing configure IPoIB Test Suite (configipoib) Fri Oct 07 11:31:49 EDT
2016 ...
Executing TEST SUITE configure IPoIB CASE (configipoib.phgppriv11.config)
config ipoib on phgppriv11 ...
...
Done configure IPoIB Test Suite Fri Oct 07 11:31:51 EDT 2016

Hit any key to continue (or ESC to abort)...
```

3. Press any key or **ESC** to end the operation.

4.3.8 Building Test Applications and Copying to Hosts

(Host) The **Build Test Apps and Copy to Hosts** selection allows you to build the MPI and/or SHMEM sample applications on the Management Node and copy the resulting object files to all the hosts. This is in preparation for execution of MPI and/or SHMEM performance tests and benchmarks.

Associated CLI commands: `opascpall`, `opauploadall`, and `opacmdall`

1. From the FastFabric OPA Host Setup menu, type 7.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Setup: Build Test Apps and Copy to Hosts
Do you want to build MPI Test Apps? [y]:
```

3. Press **Enter**.

The MPI Directory Selection TUI is displayed.

```
Host Setup: Build Test Apps and Copy to Hosts
MPI Directory Selection

Please Select MPI Directory:
0) /usr/mpi/gcc/mvapich2-x.x
1) /usr/mpi/gcc/mvapich2-x.x-hfi
2) /usr/mpi/gcc/openmpi-x.x.x
3) /usr/mpi/gcc/openmpi-x.x.x-hfi
4) /usr/mpi/intel/mvapich2-x.x-hfi
5) /usr/mpi/intel/openmpi-x.x.x-hfi
6) /usr/mpi/pgi/mvapich2-x.x-hfi
7) /usr/mpi/pgi/openmpi-x.x.x-hfi
8) Enter Other Directory

X) Return to Previous Menu (or ESC)
```

4. Select the target menu item or type **x** to return to the operation.

The next prompt is shown:

```
Do you want to build SHMEM Test Apps? [y]:
```



5. Press **Enter**.

The MPI Directory Selection for SHMEM Job Launch TUI is displayed.

```
Host Setup: Build Test Apps and Copy to Hosts
MPI Directory Selection for SHMEM Job Launch

Please Select MPI Directory:
0) /usr/mpi/gcc/openmpi-x.x.x
1) /usr/mpi/gcc/openmpi-x.x.x-hfi
2) /usr/mpi/intel/openmpi-x.x.x-hfi
3) /usr/mpi/pgi/openmpi-x.x.x-hfi
4) Enter Other Directory
5) Skip MPI Directory Selection for SHMEM Job Launch

X) Return to Previous Menu (or ESC)
```

6. Select the target menu item or type **x** to return to the operation.

7. Follow the prompts to complete the operation.

4.3.9 Rebooting Hosts

(Linux) The **Reboot Hosts** selection allows you to reboot all the selected hosts and ensure they fully reboot, as verified through ping over the management network.

Associated CLI command: `opahostadmin reboot`

1. From the FastFabric OPA Host Setup menu, type **8**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

Reboot begins immediately.

```
Performing Host Setup: Reboot Hosts
Executing: /usr/sbin/opahostadmin -f /etc/opa/hosts reboot
Executing reboot Test Suite (reboot) Fri Oct 07 11:58:48 EDT 2016 ...
Executing TEST SUITE reboot CASE (reboot.phgppriv11.reboot) phgppriv11
reboot ...
```

4.3.10 Refreshing SSH Known Hosts

(Linux) The **Refresh SSH Known Hosts** selection allows you to refresh the SSH known hosts list on this server for the Management Network. This may be used to update security for this host if hosts are replaced, reinstalled, renamed, or repaired.

Associated CLI command: `opasetupssh`

1. From the FastFabric OPA Host Setup menu, type **9**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.



2. Type **P** to begin the operation.

```
Performing Host Setup: Refresh SSH Known Hosts
Executing: /usr/sbin/opa-setupssh -p -U -f /etc/opa/hosts
Verifying localhost ssh...
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
localhost: Connected
Warning: Permanently added 'phgppriv10,10.228.209.74' (ECDSA) to the list of
known hosts.
phgppriv10: Connected
ssh: Could not resolve hostname phgppriv10-opa: Name or service not known
Connecting to phgppriv11...
Warning: Permanently added 'phgppriv11,10.228.209.75' (ECDSA) to the list of
known hosts.
phgppriv11: Connected
ssh: Could not resolve hostname phgppriv11-opa: Name or service not known
setup_self_ssh 100% 5599 5.5KB/s 00:00
phgppriv11: Verifying localhost ssh...
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
localhost: Connected
Warning: Permanently added 'phgppriv11,10.228.209.75' (ECDSA) to the list of
known hosts.
phgppriv11: Connected
ssh: Could not resolve hostname phgppriv11-opa: Name or service not known
phgppriv11: Configured localhost ssh
Successfully processed: 3
Hit any key to continue (or ESC to abort)...
```

3. Press any key or **ESC** to end the operation.

4.3.11 Rebuilding MPI Library and Tools

(Host) The **Rebuild MPI Library and Tools** allows you to rebuild the MPI Library and related tools (such as `mpirun`), and install the resulting rpms on all the hosts.

This operation is performed using the `do_build` tool supplied with the MPI Source. When rebuilding MPI, `do_build` prompts you for selection of which MPI to rebuild, and provides choices as to which available compiler to use. Refer to *Intel® Omni-Path Host Fabric Interface Installation Guide* and *Intel® Omni-Path Fabric Host Software User Guide* for more information.

Associated CLI commands: `opascpall`, `opauploadall`, and `opacmdall`

1. From the FastFabric OPA Host Setup menu, type **a**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Setup: Rebuild MPI Library and Tools
Executing: cd //usr/src/opa/MPI; ./do_build

OFED MVAPICH MPI Library/Tools rebuild
1) openmpi
2) mvapich2
Select MPI to Build:
```

3. Enter the menu item to rebuild and press **Enter**.
 - a. For openmpi, go to step 4.



- b. For mvapich2, go to step 5.
4. Rebuild openmpi.

```
OFA OpenMPI MPI Library/Tools rebuild
1) gcc
Select Compiler:
```

- a. Enter the menu item and press **Enter**.

```
Build for Omnipath HFI PSM [y]:
```

- b. Press **Enter** to continue or **n** and **Enter** to abort.

```
Executing: cd /usr/src/opa/MPI && /usr/sbin/opascpall -p -f /etc/opa/
hosts /var/tmp
Usage: opascpall [-p] [-r] [-f hostfile] source_file ... dest_file
       opascpall -t [-p] [-f hostfile] [source_dir [dest_dir]]
       or
       opascpall --help
--help - produce full help text
-p - perform copy in parallel on all hosts
-r - recursive copy of directories
-t - optimized recursive copy of directories using tar
     if dest_dir omitted, defaults to current directory name
     if source_dir and dest_dir omitted, both default to current
directory
-f hostfile - file with hosts in cluster, default is /etc/opa/hosts
source_file - list of source files to copy
source_dir - source directory to copy, if omitted . is used
dest_file - destination for copy.
             If more than 1 source file, this must be a directory
dest_dir - destination for copy. If omitted current directory name is
used
example:
opascpall MPI-PMB /root/MPI-PMB
opascpall -t -p /usr/src/opa/mpi_apps /usr/src/opa/mpi_apps
opascpall a b c /root/tools/
user@ syntax cannot be used in filenames specified
To copy from hosts in the cluster to this host, use opauploadall
Hit any key to continue (or ESC to abort)...
```

- c. Press any key to continue.

```
Executing: /usr/sbin/opacmdall -p -f /etc/opa/hosts 'cd /var/tmp; rpm -U
--force ; rm -f '
[root@phgppriv11]# cd /var/tmp; rpm -U --force ; rm -f
...
Hit any key to continue (or ESC to abort)...
```

- d. Press any key or **ESC** to end operation.
5. Rebuild mvapich2.

```
OFA MVAPICH2 MPI Library/Tools rebuild
1) gcc
Select Compiler:
```



- a. Enter the menu item and press **Enter**.

```
1) ofa
2) opa-psm
Select MVAICH2 Implementation (opa-psm recommended):
```

- b. Enter the menu item and press **Enter**.

```
Executing: cd /usr/src/opa/MPI && /usr/sbin/opascpall -p -f /etc/opa/
hosts /var/tmp
Usage: opascpall [-p] [-r] [-f hostfile] source_file ... dest_file
       opascpall -t [-p] [-f hostfile] [source_dir [dest_dir]]
       or
       opascpall --help
--help - produce full help text
-p - perform copy in parallel on all hosts
-r - recursive copy of directories
-t - optimized recursive copy of directories using tar
    if dest_dir omitted, defaults to current directory name
    if source_dir and dest_dir omitted, both default to current
    directory
-f hostfile - file with hosts in cluster, default is /etc/opa/hosts
source_file - list of source files to copy
source_dir - source directory to copy, if omitted . is used
dest_file - destination for copy.
            If more than 1 source file, this must be a directory
dest_dir - destination for copy. If omitted current directory name is
used
example:
  opascpall MPI-PMB /root/MPI-PMB
  opascpall -t -p /usr/src/opa/mpi_apps /usr/src/opa/mpi_apps
  opascpall a b c /root/tools/
user@ syntax cannot be used in filenames specified
To copy from hosts in the cluster to this host, use opauploadall
Hit any key to continue (or ESC to abort)...
```

- c. Press any key to continue.

```
Executing: /usr/sbin/opacmdall -p -f /etc/opa/hosts 'cd /var/tmp; rpm -U
--force ; rm -f '
[root@phgppriv11]# cd /var/tmp; rpm -U --force ; rm -f
...
Hit any key to continue (or ESC to abort)...
```

- d. Press any key or **ESC** to end operation.

4.3.12 Running a Command on All Hosts

(Linux) The **Run a Command on All Hosts** selection allows you to perform other operations on all hosts. Each time this is executed, a Linux* shell command may be specified to be executed against all selected hosts. You can also specify a sequence of commands separated by semicolons.

Associated CLI command: `opacmdall`

1. From the FastFabric OPA Host Setup menu, type **b**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.



2. Type **P** to begin the operation.

```
Performing Host Setup: Run a Command on All Hosts
Enter Command to run on all hosts (or none):
```

3. Enter a Linux command and press **Enter**.

```
Timelimit in minutes (0=unlimited): [1]:
```

4. Specify a time limit and press **Enter**.

```
Run in parallel on all hosts? [y]:
```

5. Select **y** (yes) or **n** (no) and press **Enter**.

```
About to run: /usr/sbin/opacmdall -T 60 -f /etc/opa/hosts 'xxxx'
Are you sure you want to proceed? [n]:
```

6. Type **y** and press **Enter** to proceed with the operation.

The operation is completed.

4.3.13 Copying a File to All Hosts

(Linux) The **Copy a File to All Hosts** selection allows you to run the `opascpall` command. A file on the local host may be specified to be copied to all selected hosts.

Associated CLI command: `opascpall`

1. From the FastFabric OPA Host Setup menu, type **c**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Setup: Copy a File to All Hosts
Enter File to copy to all hosts (or none):
```

3. Enter the name of the file to copy and press **Enter**.

```
Are you sure you want to proceed? [n]:
```

4. Type **y** and press **Enter** to continue.

```
Executing: /usr/sbin/opascpall -p -f /etc/opa/hosts /root/xxx /root/xxx
scp -q /root/xxx root@[phgppriv11]:/root/xxx
...
Hit any key to continue (or ESC to abort)...
```

5. Press any key or **ESC** to end the operation.



4.3.14 Viewing opahostadmin Result Files

(All) The **View opahostadmin Result File** selection allows you to display the `test.log` and `test.res` files that contain the results from prior `opahostadmin` runs, such as installing Fabric software or rebooting all hosts. You are also given the option to remove these files after viewing them.

If prior files are not removed, subsequent runs of `opachassisadmin`, `opahostadmin`, or `opaswitchadmin` from within the current directory continue to append to these files.

Note: For more information on the log files, refer to [Interpreting the opahostadmin, opachassisadmin, and opaswitchadmin log files](#) and [opahostadmin Details](#).

1. From the FastFabric OPA Host Setup menu, type **d**.

The menu item changes from `[Skip]` to `[Perform]`.

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Setup: View opahostadmin Result Files
Using vi (to select a different editor, export EDITOR).
About to: vi /root/test.res /root/test.log
Hit any key to continue (or ESC to abort)...
```

3. Press any key to view the `opahostadmin` results files.
4. After reviewing and closing the log, you are prompted to remove the following files.

```
Would you like to remove test.res test.log test_tmp* and save_tmp
in /root ? [n]:
```

5. Select **y** (yes) or **n** (no) and press **Enter**.
6. If you chose **y** in the step above, press any key or **ESC** to end the operation.

4.4 Verifying the Host

The FastFabric OPA Host Verification/Admin Menu allows you to verify hosts and the fabric, as well as manage of all the hosts.

To access up the FastFabric OPA Host Setup Menu, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opafastfabric**.

The Intel FastFabric OPA Tools menu is displayed.

```
Intel FastFabric OPA Tools
Version: X.X.X.X.X

1) Chassis Setup/Admin
2) Externally Managed Switch Setup/Admin
3) Host Setup
4) Host Verification/Admin
```



```

5) Fabric Monitoring
X) Exit

```

3. Type 4.

The FastFabric OPA Host Verification/Admin Menu is displayed.

```

FastFabric OPA Host Verification/Admin Menu
Host File: /etc/opa/allhosts
Validation:
0) Edit Config and Select/Edit Host File           [ Skip ]
1) Summary of Fabric Components                   [ Skip ]
2) Verify Hosts Are Pingable, SSHable, and Active [ Skip ]
3) Perform Single Host Verification               [ Skip ]
4) Verify OPA Fabric Status and Topology          [ Skip ]
5) Verify Hosts See Each Other                   [ Skip ]
6) Verify Hosts Ping via IPoIB                   [ Skip ]
7) Refresh SSH Known Hosts                       [ Skip ]
8) Check MPI Performance                         [ Skip ]
9) Check Overall Fabric Health                   [ Skip ]
a) Start or Stop Bit Error Rate Cable Test       [ Skip ]
Admin:
b) Generate All Hosts Problem Report Info        [ Skip ]
c) Run a Command on All Hosts                   [ Skip ]
Review:
d) View opahostadmin Result Files                [ Skip ]

P) Perform the Selected Actions                  N) Select None
X) Return to Previous Menu (or ESC)

```

4. Select one or more items by typing the alphanumeric character associated with the item to toggle the selection from Skip to Perform.

5. Type P to perform the operations.

Note: Each menu item will present you with prompts to complete the operation.

Table 12. FastFabric OPA Host Verification/Admin Menu Descriptions

Menu Item	Description
0) Edit Config and Select/Edit Host File	<p>Allows you to edit the following configuration files:</p> <ul style="list-style-type: none"> • /etc/opa/allhosts The allhosts file lists of all hosts including the FastFabric toolset node. • /etc/opa/ports The ports file lists the local HFI ports (for example, subnets) to be used to access the fabric for analysis. • /etc/opa/opafastfabric.conf The opafastfabric.conf file lists the default settings for most of the FastFabric command line options.
1) Summary of Fabric Components	Allows you to view a brief summary of the components in the fabric including the number of components, how many switch chips, HFIs, and links. It also indicates whether any degraded or omitted (quarantined or out of policy) links were found.
2) Verify Hosts Are Pingable, SSHable, and Active	Allows you to ping all the hosts listed through the Management Network.
3) Perform Single Host Verification	Allows you to perform verification on all nodes in the selected host file including configuration, performance, and stability using a variety of tools and checks including single node HPL .

continued...



Menu Item	Description
	For additional information on the verification that is performed, refer to the <code>/usr/share/opa/samples/hostverify.sh</code> file.
4) Verify OPA Fabric Status and Topology	(Host or All) Allows you to review the fabric state and error counts of all ports.
5) Verify Hosts See Each Other	(Host) Allows you to verify that each host can see all the others through queries to the Subnet Administrator.
6) Verify Hosts Ping via IPoIB	(Host) Allows you to verify that IPoIB is properly configured and running on all the hosts. This is accomplished through the Fabric Management node pinging each host using IPoIB.
7) Refresh SSH Known Hosts	(Linux) Allows you to refresh the ssh known hosts list on this server for the IPoIB and Management Networks. This option may be used to update security for this host to complete installation of the hosts or if hosts are replaced, reinstalled, renamed, or repaired.
8) Check MPI Performance	(Host) Allows you to perform a quick check of PCI and MPI performance using end-to-end latency and bandwidth tests.
9) Check Overall Fabric Health	(Host) Allows you to check the overall fabric health.
a) Start or Stop Bit Error Rate Cable Test	(Host) Allows you to start or stop the Cable Bit Error Rate stress tests for HFI-to-switch links and/or ISLs.
b) Generate All Hosts Problem Report Info	(Host) Allows you to collect configuration and status information from all hosts and generates a single <code>*.tgz</code> file, which can be sent to a support representative.
c) Run a Command on All Hosts	(Linux) Allows you to execute a command on all hosts.
d) View opahostadmin Result Files	Allows you to view the <code>test.log</code> and <code>test.res</code> files that reflect the results from opahostadmin runs (such as those for installing software or rebooting all hosts per menu items above).

4.4.1 Editing the Configuration Files for Host Verification

The **Edit Config and Select/Edit Host File** section allows you to select and edit the hosts, ports, and FastFabric configuration files.

1. From the FastFabric OPA Host Verification/Admin menu, type **0**.

The menu item changes from `[Skip]` to `[Perform]`.

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Admin: Edit Config and Select/Edit Host File
Using vi (to select a different editor, export EDITOR).
You will now have a chance to edit/review the FastFabric Config File:
/etc/opa/opafastfabric.conf
The values in this file will control the default operation of the
FastFabric Tools. With the exception of the host file to use,
the values you specify for defaults will be used for all FastFabric
Operations performed via this menu system
Beware existing environment variables will override the values in this file.

About to: vi /etc/opa/opafastfabric.conf
Hit any key to continue (or ESC to abort)...
```

3. Press any key to open the `opafastfabric.conf` file or **ESC** to abort the operation.



Note: To get to subsequent configuration files, you must access each file.
The configuration file opens.

4. Review the settings.

Especially review the following:

- FF_TOPOLOGY_FILE
- FF_IPOIB_SUFFIX
- FF_DEVIATION_ARGS
- ff_host_basename_to_ipoib
- ff_host_basename

Refer to [FastFabric Configuration File](#) on page 57 for more information.

Note: Intel recommends that a FastFabric topology file is created as `/etc/opa/topology.0:0.xml` to describe the intended topology of the fabric. The file can also augment assorted fabric reports with customer-specific information, such as cable labels and additional details about nodes, SMS, links, ports, and cables. Refer to [Fabric Topology Input File](#) on page 64, [Topology Verification](#) on page 329, and [opareport Detailed Information](#) on page 183 for more information about topology verification files.

5. After saving and closing the `opafastfabric.conf` file in the editor, you will be given the opportunity to edit the `ports` file.

```
You will now have a chance to edit/review the FastFabric PORTS_FILE:
/etc/opa/ports
Some of the FastFabric operations which follow will use this file to
specify the local HFI ports to use to access the fabric(s) to operate on
Beware existing environment variables will override the values in this file.

About to: vi /etc/opa/ports
Hit any key to continue (or ESC to abort)...
```

6. Press any key to open the `ports` file or **ESC** to abort the operation.

The configuration file opens.

a. Review the file:

- For typical single-subnet clusters, the default of "0:0" may be used. This uses the first active port on the Management Node to access all externally managed switches.
- For configuring a cluster with multiple subnets, refer to *Intel® Omni-Path Fabric Software Installation Guide*.

Refer to [Ports List Configuration File](#) on page 57 for more information.

For further details about the Port List File format, refer to [Port List Files](#) on page 48.

7. After saving and closing the `ports` file in the editor, you will be given the opportunity to edit the `allhosts` file.

```
The FastFabric operations which follow will require a file
listing the hosts to operate on
You should select a file which INCLUDES this host
Select Host File to Use/Edit [/etc/opa/allhosts]:
```



8. Select the host file to edit or leave blank for the default and press **Enter**.

```
About to: vi /etc/opa/allhosts
Hit any key to continue (or ESC to abort)...
```

9. Press any key to open the `allhosts` file or **ESC** to abort the operation.

The configuration file opens.

Refer to [Hosts List Configuration Files](#) on page 62 for more information.

For further details about the Host Lists file format, refer to [Host List Files](#) on page 42.

10. Create the file with the Management Node's host name (the TCP/IP management network name, for example `mgmthost`) and include the hosts file previously created.

Enter one host's name per line. For example:

```
mgmthost
include /etc/opa/hosts
```

11. After saving and closing the `hosts` file in the editor, you will be given the opportunity to review and change the configuration files again.

```
Selected Host File: /etc/opa/allhosts
Do you want to edit/review/change the files? [y]:
```

12. Press **Enter** to review and edit the files again or type **n** and press **Enter** to end the operation.

4.4.2 Viewing a Summary of Fabric Components

The **Summary of Fabric Components** selection allows you to generate a brief summary of the counts of components in the fabric, including how many switch chips, hosts, and links are in the fabric. The summary also indicates whether any degraded or omitted links were found which can indicate a poorly seated or bad cable, incorrect fabric configuration, or security issues.

Associated CLI command: `opafabricinfo` described in *Intel® Omni-Path Fabric Host Software User Guide*

1. From the FastFabric OPA Host Verification/Admin menu, type **1**.

The menu item changes from `[Skip]` to `[Perform]`.

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

The summary is generated.

```
Performing Host Admin: Summary of Fabric Components
Executing: /usr/sbin/opafabricinfo
Fabric 0:0 Information:
SM: phcpriv10 hfil_0 Guid: 0x0011750101575300 State: Master
Number of HFI's: 2
Number of Switches: 0
Number of Links: 1
```




```

Number of HFI Links: 1          (Internal: 0   External: 1)
Number of ISLs: 0              (Internal: 0   External: 0)
Number of Degraded Links: 0    (HFI Links: 0   ISLs: 0)
Number of Omitted Links: 0     (HFI Links: 0   ISLs: 0)
-----

```

```

Hit any key to continue (or ESC to abort)...

```

3. Press any key or **ESC** to end the operation.

4.4.3 Verifying Hosts Pingable, SSHable, and Active

The **Verify Hosts Pingable, SSHable, and Active** selection allows you to verify each host and provides a concise summary of the bad hosts found.

Interactive prompts allow you to select ping, SSH, and port active verification. After completion of this test, you have the option of using the resulting good hosts file for the remainder of the operations within this TUI session.

Associated CLI command: `opapingall`

1. From the FastFabric OPA Host Verification/Admin menu, type 2.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.
3. For each prompt, provide the required information and press **Enter**:

Prompt	Description
Would you like to verify hosts are ssh-able? [y]:	Allows you to see which hosts are ssh-able.
Would you like to verify host OPA ports are active? [y]:	Allows you to view which ports are active.
Would you like to verify host OPA ports are not quarantined? [y]:	Allows you to view which ports are not quarantined.

After executing the prompts, the results are displayed.

```

Executing: /usr/sbin/opafindgood -f /etc/opa/allhosts
2 hosts will be checked
2 hosts are pingable (alive)
2 hosts are ssh'able (running)
opasaquery: Failed to open port hfi 0:0: Resource temporarily unavailable
0 total hosts have FIs active on one or more fabrics (active)
opareport: No Active ports found in System
Parse error at line 1: no element found
Parse error at line 1: Fatal error parsing file 'stdin'
opaxmlextract: XML Parse error
0 hosts are alive, running, active (good)
2 hosts are bad (bad)
Bad hosts have been added to /root/punchlist.csv
Hit any key to continue (or ESC to abort)...

```

The following files are created in `opasorthosts` with all duplicates removed in the `OPA_CONFIG_DIR/` directory:

- `good`



- alive
- running
- active
- bad
- quarantined

The resulting `good` file can then be used in as input for subsequent verification commands and to create `mpi_hosts` files for running `mpi_apps` and the HFI-SW cable test.

4. If some hosts were not found, press **ESC** and use the following list to assist in troubleshooting:
 - Host powered on and booted?
 - Host connected to management network?
 - Host management network IP address and network settings consistent with DNS or `/etc/hosts`?
 - Management node connected to the management network?
 - Management node IP address and network settings correct?
 - Management network itself up (including switches, routers, and others)?
 - Correct set of hosts listed in the hosts file? You may need to repeat the previous step to review and edit the file.

After fixing the issues, restart this task.

5. If all hosts were found, press any key to continue.

```
Would you like to now use /etc/opa/good as Host File? [y]:
```

6. Press **Enter** to use the host file or type **n** and press **Enter** to end the operation

4.4.4 Performing Single Host Verification

The **Perform Single Host Verification** selection allows you to perform a single host test on all hosts.

Associated CLI commands: `opacheckload` and `opaverifyhosts`

Notes:

- Prior to using this selection, you must have a copy of the `hostverify.sh` in the directory pointed to by `FF_HOSTVERIFY_DIR`.
- If the file does not exist in that directory, copy the sample file `/usr/share/opa/samples/hostverify.sh` to the directory pointed to by `FF_HOSTVERIFY_DIR`. When placed in the editor to review `hostverify.sh`, review the settings near the top and the list of TESTS selected, edit and save as needed.
- This test can be run on a subset of hosts placed in a file created under `/etc/opa`. The test then allows tailoring `hostverify.sh` for that subset. The tailored `hostverify.sh` can be saved with a unique suffix using the subset filename.

1. From the FastFabric OPA Host Verification/Admin menu, type 3.

The menu item changes from `[Skip]` to `[Perform]`.



Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

When operating on different subsets, select menu items 0 and 3 each time. Then select the desired host file while following the flow for menu item 0.

2. Type **x** to begin the operation.
3. For each prompt, provide the required information and press **Enter**:

Prompt	Description
Would you like to copy /root/hostverify.sh to hosts? [y]:	Allows you to copy the local hostverify.sh to the destination host.
Would you like to copy /usr/share/opa/samples/hostverify.sh to /root/hostverify_allhosts.sh? [n]:	Allows you to copy the hostverify.sh file from the sample directory in order to edit for use. NOTE: The copy location is dependent on the file name under /etc/opa used for listing the hosts to operate on using FastFabric OPA Host Verification/Admin menu Step 0. (Used /etc/hosts/allhosts in this example.)
Would you like to edit /root/hostverify_allhosts.sh? [y]:	Allows you to edit the hostverify_*.sh file. The next prompt will appear after you close the file.
Would you like to copy /root/hostverify_allhosts.sh to hosts? [y]:	Allows you to copy the local hostverify.sh to the destination host. Choose n only if /root/hostverify_allhosts.sh on hosts has not changed.
Would you like to specify tests to run? [n]:	Allows you to run specific tests.
Enter filename for upload destination file [hostverify.res]:	Allows you to enter a file name for the results file or use the default hostverify.res.
Timelimit in minutes: [1]:	Allows you to set the time limit for the tests.
View Load on hosts prior to verification? [y]:	Allows you to view the load on the hosts before verification begins.

After executing the prompts, the average loads per host are displayed.

```
Executing: /usr/sbin/opacheckload -f /etc/opa/allhosts
loadavg          host
0.00 0.01 0.05 2/1161 3044   phkpstl085
0.00 0.01 0.05 2/1161 3044   phkpstl085
0.00 0.01 0.05 2/1117 25477   phkpstl087
0.00 0.01 0.05 1/1118 25164   phkpstl086
Hit any key to continue (or ESC to abort)...
```

4. Press any key to start the tests.

```
Executing: /usr/sbin/opaverifyhosts -k -c -u hostverify.res -T 60 -f /etc/opa/allhosts
-F /root/hostverify_allhosts.sh
Killing hostverify and xhpl on hosts...
[root@phkpstl085]# kill -9 -f -x 'host[v]erify.*.sh'; kill -9 '[x]hpl'; echo -n
[root@phkpstl086]# kill -9 -f -x 'host[v]erify.*.sh'; kill -9 '[x]hpl'; echo -n
[root@phkpstl087]# kill -9 -f -x 'host[v]erify.*.sh'; kill -9 '[x]hpl'; echo -n
[root@phkpstl085]# kill -9 -f -x 'host[v]erify.*.sh'; kill -9 '[x]hpl'; echo -n
3 hosts will be verified
SCPing /root/hostverify_allhosts.sh to /root/hostverify.sh ...
scp -q /root/hostverify_allhosts.sh root@[phkpstl086]:/root/hostverify.sh
scp -q /root/hostverify_allhosts.sh root@[phkpstl087]:/root/hostverify.sh
scp -q /root/hostverify_allhosts.sh root@[phkpstl085]:/root/hostverify.sh
Running /root/hostverify.sh -d /root ...
phkpstl085: FAIL initscripts: acpi_pad kernel module loaded is loaded - unload or
blacklist.
phkpstl087: FAIL initscripts: acpi_pad kernel module loaded is loaded - unload or
```



```

blacklist.
phpkstl087: FAIL ipoib: ib0 is in 'datagram' mode - should be in 'connected' mode
phpkstl086: FAIL initscripts: acpi_pad kernel module loaded is loaded - unload or
blacklist.
phpkstl086: FAIL ipoib: ib0 is in 'datagram' mode - should be in 'connected' mode
phpkstl085: FAIL initscripts: acpi_pad kernel module loaded is loaded - unload or
blacklist.
Killing hostverify and xhpl on hosts...
[root@phpkstl087]# pkill -9 -f -x 'host[v]erify.*.sh'; pkill -9 '[x]hpl'; echo -n
[root@phpkstl086]# pkill -9 -f -x 'host[v]erify.*.sh'; pkill -9 '[x]hpl'; echo -n
[root@phpkstl085]# pkill -9 -f -x 'host[v]erify.*.sh'; pkill -9 '[x]hpl'; echo -n
[root@phpkstl085]# pkill -9 -f -x 'host[v]erify.*.sh'; pkill -9 '[x]hpl'; echo -n
Uploading /root/hostverify.res to ./uploads/hostverify.res ...
scp -q root@[phpkstl086]:/root/hostverify.res ./uploads/phpkstl086/hostverify.res
scp -q root@[phpkstl087]:/root/hostverify.res ./uploads/phpkstl087/hostverify.res
scp -q root@[phpkstl085]:/root/hostverify.res ./uploads/phpkstl085/hostverify.res
scp -q root@[phpkstl085]:/root/hostverify.res ./uploads/phpkstl085/hostverify.res
About to: vi /root/verifyhosts.res
Hit any key to continue (or ESC to abort)...

```

5. Press any key to view the results file.
The results of the test are shown in the editor.
6. Close the results file to end the operation.

4.4.5 Verifying OPA Fabric Status and Topology

The **Verify OPA Fabric Status and Topology** selection allows you to run various checks on the fabric and topology.

Associated CLI commands: [opashowallports](#) and [opareport](#)

1. From the FastFabric OPA Host Verification/Admin menu, type **4**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.
3. For each prompt, provide the required information and press **Enter**:

Prompt	Description
Would you like to perform fabric error analysis? [y]:	Allows you to perform fabric error analysis.
Clear error counters after generating report? [n]:	Allows you to clear the error counters after the report is generated.
Would you like to perform fabric link speed error analysis? [y]:	Allows you to perform link speed error analysis.
Check for links configured to run slower than supported? [n]:	Allow you to look for links that are configured to run slower than supported.
Check for links connected with mismatched speed potential? [n]:	Allows you to look for connected links with mismatch speed potential.
Would you like to verify fabric topology? [y]:	Allows you to verify the fabric topology. NOTE: The fabric deployment can be verified against the planned topology. Typically the planned topology will have been converted to an XML topology file using <code>opaxlattopology</code> , <code>opaxlattopology_cust</code> (deprecated) or a customized variation. If this step has been done and a topology file has been placed in the location specified by the

continued...



Prompt	Description
	FF_TOPOLOGY_FILE in opafastfabric.conf file, then a topology verification can be performed. Refer to Topology Verification on page 329 and opareport Detailed Information on page 183 for more information.
Verify all aspects of topology (links, nodes, SMs)? [y]:	Allows you to verify all links, nodes and SMs in the topology.
Include unexpected devices in punchlist? [y]:	Allows you to include unexpected devices in the punchlist.
Enter filename for results [/root/linkanalysis.res]:	Allows you to enter a file name for the result file or accept the default linkanalysis.res.

After executing the prompts, the average loads per host are displayed.

```
Executing: /usr/sbin/opacheckload -f /etc/opa/allhosts
loadavg          host
0.66 0.41 0.20 2/880 193597      phgppriv10
0.00 0.01 0.05 1/813 4001       phgppriv11
Hit any key to continue (or ESC to abort)...
```

The following items are verified:

- Perform a fabric error analysis.
- Perform a fabric link speed error analysis.
- Check for links that are configured to run slower than supported.
- Check links that are connected with mismatched speed potential.
- Verify the fabric topology.
- Verify all aspects of the topology including links, nodes, and SMs.
- Include unexpected devices in the punchlist.

The results can be seen in the \$FF_RESULT_DIR/linkanalysis.res file. A punch list of issues is appended to the \$FF_RESULT_DIR/punchlist.csv file.

4. Press any key or **ESC** to end the operation.

Note: You can use CLI commands to clear error counters after generating the report, add `clearerrors` and optionally `clearhwerrors` options to the `opalinkanalysis` run. Be aware that clear hardware counters (`-A` option) is optional and may affect the PM and other tools. See "PM Running Counters to Support opareport" section in the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide* for more information.

4.4.6 Verifying Hosts See Each Other

(Host) The **Verify Hosts See Each Other** selection allows you to confirm that each host can see all the others through queries to the Subnet Administrator. This ensures all nodes are connected to the same fabric and can properly access the Subnet Administrator.

Associated CLI command: `opahostadmin sacache`

1. From the FastFabric OPA Host Verification/Admin menu, type 5.

The menu item changes from [Skip] to [Perform].



Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Admin: Verify Hosts See Each Other
Executing: /usr/sbin/opahostadmin -f /etc/opa/allhosts sacache
Executing sacache Test Suite (sacache) Fri Oct 07 16:06:08 EDT 2016 ...
Executing TEST SUITE sacache CASE (sacache.phgppriv10.dsc) phgppriv10 can see
phgppriv10 phgppriv11 ...
Executing TEST SUITE sacache CASE (sacache.phgppriv11.dsc) phgppriv11 can see
phgppriv10 phgppriv11 ...
...
Done sacache Test Suite Fri Oct 07 16:08:10 EDT 2016

Hit any key to continue (or ESC to abort)...
```

3. Press any key or **ESC** to end the operation.

4.4.7 Verifying Hosts Ping via IPoIB

(Host) The **Verify Hosts Ping via IPoIB** selection allows you to confirm that IPoIB is properly configured and running on all the hosts. This is accomplished through the Management Node pinging each host through IPoIB.

Note: This operation requires that IPoIB be enabled on the Management Node as well as on each host selected for verification. Also, the management host must have IPoIB configured.

Associated CLI command: `opahostadmin ipoibping`

1. From the FastFabric OPA Host Verification/Admin menu, type **6**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

The status is displayed.

```
Performing Host Admin: Verify Hosts Ping via IPoIB
Executing: /usr/sbin/opahostadmin -f /etc/opa/allhosts ipoibping
Executing ipoib ping Test Suite (ipoibping) Fri Oct 07 16:10:53 EDT 2016 ...
Executing TEST SUITE ipoib ping CASE (ipoibping.localhost_ping1) simple ping
from localhost ...
TEST SUITE ipoib ping CASE (ipoibping.localhost_ping1) simple ping from
localhost ...
...
TEST CASE simple ping from localhost: 2 Items; 2 PASSED; 0 FAILED
TEST SUITE ipoib ping CASE (ipoibping.localhost_ping1) simple ping from
localhost PASSED
TEST SUITE ipoib ping: 1 Cases; 1 PASSED; 0 FAILED
TEST SUITE ipoib ping PASSED
Done ipoib ping Test Suite Fri Oct 07 16:10:54 EDT 2016

Hit any key to continue (or ESC to abort)...
```

3. Press any key or **ESC** to end the operation.



4.4.8 Refreshing SSH Known Hosts

(Linux) The **Refresh SSH Known Hosts** selection allows you to refresh the SSH known_hosts file on the Management Node to include the IPoIB hostnames of all the hosts.

Note: This operation requires that IPoIB be enabled on the Management Node as well as on each host selected for verification.

Associated CLI command: `opasetupssh`

1. From the FastFabric OPA Host Verification/Admin menu, type 7.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

The status is displayed.

```
Performing Host Admin: Refresh SSH Known Hosts
Executing: /usr/sbin/opasetupssh -p -U -f /etc/opa/allhosts
Verifying localhost ssh...
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
localhost: Connected
Warning: Permanently added 'phgppriv10,10.228.209.74' (ECDSA) to the list of
known hosts.
phgppriv10: Connected
ssh: Could not resolve hostname phgppriv10-opa: Name or service not known
Connecting to phgppriv10...
Connecting to phgppriv11...
Warning: Permanently added 'phgppriv10,10.228.209.74' (ECDSA) to the list of
known hosts.
Warning: Permanently added 'phgppriv11,10.228.209.75' (ECDSA) to the list of
known hosts.
phgppriv11: Connected
phgppriv10: Connected
ssh: Could not resolve hostname phgppriv11-opa: Name or service not known
ssh: Could not resolve hostname phgppriv10-opa: Name or service not known
setup_self_ssh          100% 5599      5.5KB/s   00:00
setup_self_ssh          100% 5599      5.5KB/s   00:00
phgppriv11: Verifying localhost ssh...
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
phgppriv10: Verifying localhost ssh...
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
localhost: Connected
localhost: Connected
Warning: Permanently added 'phgppriv11,10.228.209.75' (ECDSA) to the list of
known hosts.
Warning: Permanently added 'phgppriv10,10.228.209.74' (ECDSA) to the list of
known hosts.
phgppriv11: Connected
phgppriv10: Connected
ssh: Could not resolve hostname phgppriv11-opa: Name or service not known
phgppriv11: Configured localhost ssh
ssh: Could not resolve hostname phgppriv10-opa: Name or service not known
phgppriv10: Configured localhost ssh
Successfully processed: 2
Hit any key to continue (or ESC to abort)...
```

3. Press any key or **ESC** to end the operation.

4.4.9 Checking MPI Performance

(Host) The **MPI Performance** selection allows you to perform a quick check of PCIe and MPI performance through end-to-end latency and bandwidth tests.

Note: This test identifies nodes whose performance is not consistent with others in the fabric. It is not intended as a benchmark of fabric latency and bandwidth. This test purposely uses techniques to reduce test runtime.

Associated CLI command: `opacheckload` and `opahostadmin`

1. From the FastFabric OPA Host Verification/Admin menu, type **8**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

The status is displayed.

```
Performing Host Admin: Check MPI Performance
Test Latency and Bandwidth deviation between all hosts? [y]:
```

3. Press **Enter** to select default (y).

```
View Load on hosts prior to test? [y]:
```

4. Press **Enter** to select default (y).

```
Performing Host Admin: Check MPI Performance
Test Latency and Bandwidth deviation between all hosts? [y]:
View Load on hosts prior to test? [y]:
Executing: /usr/sbin/opacheckload -f /etc/opa/allhosts
loadavg          host
0.00 0.01 0.05 1/857 234917      phgpprivl0
0.00 0.01 0.05 1/814 4642       phgpprivl1
Hit any key to continue (or ESC to abort)...
```

5. Press any key to continue.

```
Executing: /usr/sbin/opahostadmin -f /etc/opa/allhosts mpiperfdeviation
Executing mpi lat/bw deviation Test Suite (mpiperfdeviation) Fri Oct 07
16:40:18 EDT 2016 ...

TEST SUITE FAILURE: FAILURE during test suite: mpi_apps not built
TEST SUITE mpi lat/bw deviation TESTS ABORTED
TEST SUITE mpi lat/bw deviation: 0 Cases; 0 PASSED
TEST SUITE mpi lat/bw deviation FAILED
Done mpi lat/bw deviation Test Suite Fri Oct 07 16:40:18 EDT 2016

Hit any key to continue (or ESC to abort)...
```

The results display the pair-wise analysis of latency and bandwidth for the selected hosts and report pairs outside an acceptable tolerance range. By default, performance is compared relative to other hosts in the fabric. It is assumed that all hosts selected for a given run have comparable fabric performance. Failing hosts are clearly indicated.



Intel recommends that you review the `FF_DEVIATION_ARGS` parameter in `opafastfabric.conf` and adjust it as appropriate for the cluster. The default can accommodate a wide range of cluster designs.

The results are also written to the `test.res` file, which may be viewed through the **View opahostadmin result files** option. Refer to [Interpreting the opahostadmin, opachassisadmin, and opaswitchadmin log files](#) on page 247 for more details.

6. Press any key or **ESC** to end the operation.

Additional Details

If any hosts fail, carefully examine the failing hosts to verify the HFI models, PCIe slot used, BIOS settings, and any motherboard or BIOS settings related to devices on PCIe buses or slot speeds. Also verify the HFI and any riser cards are properly seated.

The bandwidth that is reported should also be checked against the PCIe speeds in the Performance Impact table below. If all pairs are not in the expected performance range, carefully examine all hosts to verify the HFI models, PCIe slot used, BIOS settings and any motherboard or BIOS settings related to devices on PCIe buses or slot speeds. Also verify the HFI and any riser cards are properly seated.

Table 13. Performance Impact

PCIe Speed	Fabric Speed	Typical Bandwidth
PCIe 8GT/s x16 (Gen3)	100 Gbps	12.0 - 12.4 GBps
PCIe 8GT/s x8 (Gen3)	100 Gbps	6.4 - 6.8 GBps
PCIe 5GT/s x16 (Gen2)	100 Gbps	6.4 - 6.8 GBps
PCIe 5GT/s x8 (Gen2)	100 Gbps	3.2 - 3.4 GBps
Note: 1 GBps = 1,000,000,000 bytes/second		

4.4.10 Checking Overall Fabric Health

The **Check Overall Fabric Health** selection allows you to baseline the present fabric configuration for use in future fabric health checks. Perform this check after configuring any additional Management Nodes and establishing a healthy fabric via successful execution of all the other tests. If desired, a baseline of an incomplete or unhealthy fabric may be taken for future comparison after making additions or corrections to the fabric.

Associated CLI command: `opaallanalysis`

Refer to Configure and Initialize Health Check Tools in the *Intel® Omni-Path Fabric Software Installation Guide* for more information.

1. From the FastFabric OPA Host Verification/Admin menu, type **9**.

The menu item changes from `[Skip]` to `[Perform]`.

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.



The status is displayed.

```
Performing Host Admin: Check Overall Fabric Health
Baseline present configuration? [n]:
```

3. Press **Enter** (n) to analyze the configuration without baselining it.

```
Executing: /usr/sbin/opaallanalysis
opafabricsanalysis: Port 0:0 Error: Previous baseline run required
opafabricsanalysis: Possible fabric errors or changes found
opachassisanalysis: Warning: showAllConfig command failed for 1 or more
chassis. See /var/usr/lib/opa/analysis/latest/chassis.showAllConfig
opachassisanalysis: Error: Chassis error. See /var/usr/lib/opa/analysis/
latest/chassis.hwCheck
opachassisanalysis: Possible Chassis errors or changes found
opaallanalysis: Possible errors or changes found
Hit any key to continue (or ESC to abort)...
```

4. Type **y** and press **Enter** to baseline the configuration.

The configuration is baselined.

```
Executing: /usr/sbin/opaallanalysis -b
opafabricsanalysis: Port 0:0 Error: Unable to access fabric.
See /var/usr/lib/opa/analysis/latest/fabric.0:0.snapshot.stderr
opafabricsanalysis: Possible fabric errors or changes found
opachassisanalysis: Warning: showAllConfig command failed for 1 or more
chassis. See /var/usr/lib/opa/analysis/latest/chassis.showAllConfig
opachassisanalysis: Baselined
opaallanalysis: Possible errors or changes found
Hit any key to continue (or ESC to abort)...
```

5. Press any key or **ESC** to end the operation.

4.4.11 Starting or Stopping Bit Error Rate Cable Test

The **Start or Stop Bit Error Rate Cable Test** selection allows you to perform host and/or ISL cable testing. The test allows for starting and stopping an extended Bit Error Rate test. The system prompts to clear hardware counters.

Note: Clearing of hardware counters (-A option) is optional and may affect the PM and other tools. See "PM Running Counters to Support opareport" section in the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide* for more information.

Intel recommends that you run this test for 20-60 minutes for a thorough test. While the test is running, monitor the fabric for signal integrity or stability errors using `opatop`, `opareport`, and/or the Fabric Manager GUI. Once the desired test time has elapsed, return to this item in the menu and stop the test.

Associated CLI command: `opacabletest`

1. From the FastFabric OPA Host Verification/Admin menu, type **a**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.
3. For each prompt, provide the required information and press **Enter**:



Prompt	Description
Stop or cleanup any already running Cable Test? [y]:	Allows you to stop and clean up any cable tests in process.
Stop HFI-Switch Cable Test? [y]:	Allows you to stop HFI-Switch cable test.
Stop ISL Cable Test? [y]:	Allows you to stop ISL cable test.
Start Cable Test? [y]:	Allows you to start a new cable test.
Clear error counters? [y]:	Allows you to clear the error counters.
Force Clear of hardware error counters too? [y]:	Allows you to clear hardware counters.
Start HFI-Switch Cable Test? [y]:	Allows you to start a new HFI-Switch cable test.
Start ISL Cable Test? [y]:	Allows you to start a new ISL cable test.

After executing the prompts, the following is displayed.

```
About to run: /usr/sbin/opacabletest -A
Hit any key to continue (or ESC to abort)...
```

- Press any key to execute the cabletest or **ESC** to end the operation.

4.4.12 Generating All Hosts Problem Report Information

(Host) The **Generate all Hosts Problem Report Info** selection allows you to collect configuration and status information from all hosts and generate a single *.tgz file that can be sent to an Intel support representative.

Associated CLI command: `opacaptureall`

- From the FastFabric OPA Host Verification/Admin menu, type **b**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

- Type **P** to begin the operation.

The status is displayed.

```
Performing Host Admin: Generate All Hosts Problem Report Info
Capture detail level (1-Normal 2-Fabric 3-Fabric+FDB 4-Analysis): [4]:
```

The detail levels are cumulative and shown below:

Detail Level	Description
1-Normal	Obtains local information from each host.
2-Fabric	In addition to "Normal", obtains basic fabric information by queries to the SM and fabric error analysis using iba_report.
continued...	



Detail Level	Description
3-Fabric+FDB	In addition to "Fabric", obtains all the switch forwarding tables and OPA multicast membership lists from the SM.
4-Analysis	In addition to "Fabric+FDB", obtains all <code>all_analysis</code> results. If <code>all_analysis</code> has not yet been run, it is run as part of the capture.
Notes: <ul style="list-style-type: none">• Detail levels 2-4 can be used when fabric operational problems occur. If the problem appears to be node-specific, detail level 1 should be sufficient.• Detail levels 2-4 require an operational Fabric Manager. Typically, your support representative requests a given detail level. If a given detail level takes excessively long or fails to be gathered, try a lower detail level.• For detail levels 2-4, the additional information is only gathered on the node running the <code>captureall</code> command. The information is gathered for every fabric specified in the <code>/etc/opa/ports</code> file.	

3. Type the menu item for the level of details required for the report and press **Enter**.

`opacaptureall` is initiated and results gathered in a `hostcapture.all.tgz`.

A sample of a "Normal" analysis is shown below.

```
Executing: /usr/sbin/opacaptureall -p -D 1 -f /etc/opa/allhosts
Running capture on all non-local hosts ...
[root@phkpstl042]# rm -f ~root/hostcapture.tgz; opacapture ~root/
hostcapture.tgz
Getting software and firmware version information ...
Capturing FM binaries and debuginfo if available
Getting TMM information...
Obtaining OS configuration ...
Obtaining dmesg logs ...
Obtaining present process and module list ...
Obtaining PCI device list ...
ls: cannot access /dev/ipath*: No such file or directory
Obtaining environment variables ...
Obtaining network interfaces ...
Obtaining DMI information ...
Obtaining Shared Memory information ...
Obtaining OmniPath information ...
Obtaining MPI configuration ...
Copying configuration and statistics for OPA drivers from /proc ...
Obtaining additional CPU info...
Obtaining HFI statistics ...
Copying kernel debug information from /sys/kernel/debug/hfil...
Copying configuration and statistics for ib_ drivers from /sys ...
  Getting statedump for hfil_0 ...
Copying configuration and statistics for OPA from /sys/module ...
Gathering Host FM Information ...
Creating dump directory...
Getting systemd information...
Getting FM rpm version...
Copying FM configuration...
Copying FM core dumps...
Skipping SM 0: Not Running
Skipping SM 1: Not Running
Skipping SM 2: Not Running
Skipping SM 3: Not Running
Skipping SM 4: Not Running
Skipping SM 5: Not Running
Skipping SM 6: Not Running
Skipping SM 7: Not Running
Packaging capture file...
Saved FM capture as smdump-10Oct16095350.tgz
Gathering Distributed SA data...
Creating tar file /root/hostcapture.tgz ...
Done.
```



```

Please include /root/hostcapture.tgz with any problem reports to Customer
Support
Uploading capture from each host ...
Running capture on local host ...
scp root@[phpkstl042]:hostcapture.tgz ./uploads/phpkstl042/.
Getting software and firmware version information ...
hostcapture.tgz 100% 17MB 17.2MB/s 00:00
Capturing FM binaries and debuginfo if available
Getting TMM information...
Obtaining OS configuration ...
Obtaining dmesg logs ...
Obtaining present process and module list ...
Obtaining PCI device list ...
ls: cannot access /dev/opath*: No such file or directory
Obtaining environment variables ...
Obtaining network interfaces ...
Obtaining DMI information ...
Obtaining Shared Memory information ...
Obtaining OmniPath information ...
Obtaining MPI configuration ...
Copying configuration and statistics for OPA drivers from /proc ...
Obtaining additional CPU info...
Obtaining HFI statistics ...
Copying kernel debug information from /sys/kernel/debug/hfil...
Copying configuration and statistics for ib_ drivers from /sys ...
Getting statedump for hfil_0 ...
Copying configuration and statistics for OPA from /sys/module ...
Gathering Host FM Information ...
Creating dump directory...
Getting systemd information...
Getting FM rpm version...
Copying FM configuration...
Copying FM core dumps...
Getting SM 0 counters...
Getting PM 0 counters...
Getting SM 0 run-time core file
Skipping SM 1: Not Running
Skipping SM 2: Not Running
Skipping SM 3: Not Running
Skipping SM 4: Not Running
Skipping SM 5: Not Running
Skipping SM 6: Not Running
Skipping SM 7: Not Running
Packaging capture file...
Saved FM capture as smdump-10Oct16095357.tgz
Gathering Distributed SA data...
Creating tar file /root/./uploads/phpkstl041/hostcapture.tgz ...
Done.

Please include /root/./uploads/phpkstl041/hostcapture.tgz with any problem
reports to Customer Support
Combining captured files into ./uploads/hostcapture.all.tgz ...
Done.
Hit any key to continue (or ESC to abort)...

```

4. Press any key or **ESC** to end the operation.

4.4.13 Running a Command on All Hosts

(Linux) The **Run a command on all hosts** selection allows you to perform other operations on all hosts. Each time this is executed, a Linux* shell command may be specified to be executed against all selected hosts. You can also specify a sequence of commands separated by semicolons.

Associated CLI command: [opahostadmin](#)



1. From the FastFabric OPA Host Verification/Admin menu, type **c**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Admin: Run a Command on All Hosts
Enter Command to run on all hosts (or none):
```

3. Enter a Linux command and press **Enter**.

```
Timelimit in minutes (0=unlimited): [1]:
```

4. Specify a time limit and press **Enter**.

```
Run in parallel on all hosts? [y]:
```

5. Select **y** (yes) or **n** (no) and press **Enter**.

```
About to run: /usr/sbin/opacmdall -T 60 -f /etc/opa/hosts 'xxxx'
Are you sure you want to proceed? [n]:
```

6. Type **y** and press **Enter** to proceed with the operation.

The operation is completed.

4.4.14 Viewing opahostadmin Result Files

The **View opahostadmin result files** allows you to display the `test.log` and `test.res` files that contain the results from prior `opahostadmin` runs, such as installing fabric software or rebooting all hosts. You are also given the option to remove these files after viewing them.

If prior files are not removed, subsequent runs of `opachassisadmin`, `opahostadmin`, or `opaswitchadmin` from within the current directory continue to append to these files.

Note: For more information on the log files, refer to [Interpreting the opahostadmin, opachassisadmin, and opaswitchadmin log files](#) and [opahostadmin Details](#).

1. From the FastFabric OPA Host Verification/Admin menu, type **d**.

The menu item changes from [Skip] to [Perform].

Note: More than one menu item may be selected. The operations will perform individually and in sequence with the menu.

2. Type **P** to begin the operation.

```
Performing Host Admin: View opahostadmin Result Files
Using vi (to select a different editor, export EDITOR).
About to: vi /root/punchlist.csv /root/verifyhosts.res /root/test.res /root/
test.log
Hit any key to continue (or ESC to abort)...
```



3. Press any key to view the `opahostadmin` results files.
4. After reviewing and closing the log, you are prompted to remove the following files.

```
4 files to edit
Would you like to remove verifyhosts.res test.res test.log test_tmp* and
save_tmp
in /root ? [n]:
```

5. Select **y** (yes) or **n** (no) and press **Enter**.
6. If you chose **y** in the step above, press any key or **ESC** to end the operation.



5.0 Descriptions of Command Line Tools

This section provides a complete description of each Intel® Omni-Path Fabric Suite FastFabric Toolset command line tool and its parameters.

Whereas the TUI menus in the previous section are presented sequentially showing you how to perform common fabric management tasks, the CLI tools provide more functional granularity and are grouped, in this section, according to the following categories:

- [High-Level TUIs](#)
- [Health Check and Baselining Tools](#)
- [Verification, Analysis, and Control CLIs](#)
- [Detailed Fabric Data Gathering](#)
- [Configuration and Control for Chassis, Switch, and Host](#)
- [Basic Setup and Administration Tools](#)
- [File Management Tools](#)
- [Fabric Link and Port Control](#)
- [Fabric Debug](#)
- [FastFabric Utilities](#)

Note: Basic CLI tools are described in the *Intel® Omni-Path Fabric Host Software User Guide*.

5.1 High-Level TUIs

The tools described in this section are used for fabric monitoring, deployment verification, and analysis.

5.1.1 opafastfabric

(Switch and Host) Starts the top-level Intel® Omni-Path Fabric Suite FastFabric Text-based User Interface (TUI) menu to enable setup and configuration.

Syntax

```
opafastfabric
```

Options

None.



Example

```
#opafastfabric
Intel FastFabric OPA Tools
Version: X.X.X.X.X

    1) Chassis Setup/Admin
    2) Externally Managed Switch Setup/Admin
    3) Host Setup
    4) Host Verification/Admin
    5) Fabric Monitoring

    X) Exit
```

5.1.2 opatop

Starts the Fabric Performance Monitor (`opatop`) Text-based User Interface (TUI) menu to display performance, congestion, and error information about a fabric.

Syntax

```
opatop [-v] [-q] [-h hfi] [-p port] [-i seconds]
```

Options

<code>--help</code>	Produces full help text.
<code>-v/--verbose level</code>	Specifies the verbose output level. Value is additive and includes: <ul style="list-style-type: none"> 1 Screen 4 STDERR opatop 16 STDERR PaClient
<code>-q/--quiet</code>	Disables progress reports.
<code>-h/--hfi <i>hfi</i></code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p <i>port</i></code> port is a system-wide port number. (Default is 0.)
<code>-p/--port <i>port</i></code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-i/--interval <i>seconds</i></code>	Interval in <i>seconds</i> at which PA queries are performed to refresh to the latest PA image. Default = 10 seconds.

-h and -p options permit a variety of selections:

`-h 0` First active port in system (default).



- h 0 -p 0 First active port in system.
- h x First active port on HFI x.
- h x -p 0 First active port on HFI x.
- h 0 -p y Port y within system (no matter which ports are active).
- h x -p y HFI x, port y.

5.2 Health Check and Baselining Tools

(All) The software includes tools to rapidly identify if the fabric has a problem or if its configuration has changed since the last baseline. Analysis includes hardware, software, fabric topology, and SM configuration. The tools are designed to permit easy manual execution or automated execution using `cron` or other mechanisms. The health check tools include:

- `opafabricanalysis` – Performs fabric topology and PMA error counters analysis.
- `opachassisanalysis` – Performs chassis configuration and health analysis for selected chassis.
- `opaesmanalysis` – Performs embedded SM configuration and health analysis for selected chassis.
- `opahostsmanalysis` – Performs host SM configuration and health analysis for the local host.
- `opaallanalysis` – Performs analysis on all components or a subset of components. Intel recommends this as the primary tool for general analysis.

5.2.1 Usage Model

The health check tools support three modes of operation: health check only mode, baseline mode, and check mode. The typical usage model for the tools is:

- Perform initial fabric install and verification:
 - Optionally run tools in *health check only* mode
 - Performs quick health check
 - Duplicates some of steps already done during verification
- Run tools in *baseline* mode:
 - Takes a baseline of present hardware and software configuration
- Periodically run tools in *check* mode:
 - Performs quick health check
 - Compares present hardware and software configuration to baseline
 - Can be scheduled in hourly `cron` jobs
- As needed, rerun *baseline* when expected changes occur, including:
 - Fabric upgrades



- Hardware replacements and changes
- Software configuration changes

5.2.2 Common Operations and Options

The Health Check and Baselining tool supports the following options:

- `-b` Performs a baseline snapshot of the configuration.
- `-e` Performs an error check/health analysis only.

If no option is specified, the tool performs a snapshot of the present configuration, compares it to the baseline, and performs an error check/health analysis.

Using both `-b` and `-e` on a given run is not permitted.

A typical use case is:

- Perform an initial error check by running the `-e` option.
- Review and correct the errors reported in the files indicated by the tools.
- Once all the errors are corrected, perform a baseline of the configuration using the `-b` option. The baseline configuration is saved to files in `FF_ANALYSIS_DIR/baseline`. The default `/var/usr/lib/opa/analysis/baseline` is set through `/etc/opa/opafastfabric.conf`. This baseline configuration should be carefully reviewed to make sure it matches the intended configuration. If it does not, correct the configuration and run a new baseline.

Example

```
opafabricanalysis -e
```

Errors reported could include links with high error rates, unexpected low speeds, etc. Correct any errors, then rerun `opafabricanalysis -e` to make sure there is a good fabric.

```
opafabricanalysis -b
```

The baseline configuration is saved to `FF_ANALYSIS_DIR/baseline`. This includes files starting with `links` and `comps`, which are the results of `opareport -o links` and `opareport -o comps` reports respectively. Review these files and make sure all the expected links and components are present. For example, make sure all the switches and servers in the cluster are present. Also, verify the appropriate links between servers and switches are present. If the fabric is not correctly configured, correct the configuration and rerun the baseline.

Note: Alternatively, the advanced topology verification capabilities of `opareport` can be used to verify the fabric deployment against the intended design.

Once a good baseline has been established, use the tools to compare the present fabric against the baseline and check its health.

```
opafabricanalysis
```



Checks the present fabric links and components against the previous baseline. If there have been changes, it reports a failure and indicate which files hold the resulting snapshot and differences. It also checks the PMA error counters and link speeds for the fabric, similar to `opafabricanalysis -e`. If either of these checks fail, it returns a non-zero exit status, permitting higher level scripts to detect a failed condition.

The differences files are generated using the Linux* command specified by `FF_DIFF_CMD` in `opafastfabric.conf`. By default, this is the `diff -C 1` command. It is run against the baseline and new snapshot. Therefore, lines after each `*** #, # ****` heading in the `diff` are from the baseline and lines after each `--- #, # ----` heading are from the new snapshot. If `FF_DIFF_CMD` is simply set to `diff`, lines indicated by "<" in the `diff` are from the baseline and lines indicated by ">" in the `diff` are from the new snapshot.

Another useful command is the Linux* `sdiff` command. For more information about the `diff` output format, consult the Linux* man page for `diff`.

If the configuration is intentionally changed, Intel recommends that you obtain a new error analysis and baseline using the same sequence as the initial installation to establish a new baseline for future comparisons.

In addition, all of the tools support the following two options:

- `-s`

Saves history of failures.

When the `-s` option is used, each failed run also creates a directory whose name is the date and time the analysis tool was started. The directory contains the failing snapshot information and `diffs`, allowing you to track a history of failures. Note that every run of the tools also creates a `latest` directory with the latest snapshot. The `latest` files are overwritten by each subsequent run of the tool, which means the most recent run results are always available.

Beware, frequent use of the health check tools in conjunction with `-s` can consume a large amount of disk space. The space requirements depend greatly on the size of the cluster. For example, it could be > 10 megabytes per run on a 1000 node cluster.

- `-d dir`

Specifies the top-level directory for saving baseline, snapshots, and history.

Runs using `-d` must use the same directory as any previous baseline to be compared to (except when the `-e` option is used). Default is `FF_ANALYSIS_DIR` which is set in `opafastfabric.conf`.

The `FF_ANALYSIS_DIR` option can be changed to provide a customer-specific alternate directory to be used whenever the `-d` option is not specified.

Subdirectories under `FF_ANALYSIS_DIR` are created as follows:

- `baseline` Baseline snapshot from each analysis tool.
- `latest` Latest snapshot from each analysis tool.
- `YYYY-MM-DD-HH:MM:SS` Failed analysis from analysis run with `-s`.



5.2.3 opafabricanalysis

(All) Performs analysis of the fabric.

Syntax

```
opafabricanalysis [-b|-e] [-s] [-d dir] [-c file] [-t portsfile]
[-p ports] [-T topology_input]
```

Options

<code>-- help</code>	Produces full help text.
<code>-b</code>	Specifies the baseline mode, default is compare/check mode.
<code>-e</code>	Evaluates health only, default is compare/check mode.
<code>-s</code>	Saves history of failures (errors/differences).
<code>-d dir</code>	Specifies the top-level directory for saving baseline and history of failed checks. Default = <code>/var/usr/lib/opa/analysis</code>
<code>-c file</code>	Specifies the error thresholds config file. Default = <code>/etc/opa/opamon.conf</code>
<code>-t portsfile</code>	Specifies the file with list of local HFI ports used to access fabric(s) for analysis. Default = <code>/etc/opa/ports</code>
<code>-p ports</code>	Specifies the list of local HFI ports used to access fabrics for analysis. Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format <code>hfi:port</code> , for example: <code>0:0</code> First active port in system. <code>0:y</code> Port <i>y</i> within system. <code>x:0</code> First active port on HFI <i>x</i> . <code>x:y</code> HFI <i>x</i> , port <i>y</i> .
<code>-T topology_input</code>	Specifies the name of topology input file to use. Any <code>%P</code> markers in this filename are replaced with the <code>HFI:port</code> being operated on (such as <code>0:0</code> or <code>1:2</code>). Default = <code>/etc/opa/topology.%P.xml</code> . If <code>-T NONE</code> is specified, no topology input file is used. See Details and opareport on page 171 for more information.



Example

```
opafabricanalysis  
opafabricanalysis -p '1:1 1:2 2:1 2:2'
```

The fabric analysis tool checks the following:

- Fabric links (both internal to switch chassis and external cables)
- Fabric components (nodes, links, SMs, systems, and their SMA configuration)
- Fabric PMA error counters and link speed mismatches

Note: The comparison includes components on the fabric. Therefore, operations such as shutting down a server cause the server to no longer appear on the fabric and are flagged as a fabric change or failure by `opafabricanalysis`.

Environment Variables

The following environment variables are also used by this command:

PORTS	List of ports, used in absence of <code>-t</code> and <code>-p</code> .
PORTS_FILE	File containing list of ports, used in absence of <code>-t</code> and <code>-p</code> .
FF_TOPOLOGY_FILE	File containing <code>topology_input</code> (may have <code>%P</code> marker in filename), used in absence of <code>-T</code> .
FF_ANALYSIS_DIR	Top-level directory for baselines and failed health checks.

Details

For simple fabrics, the Intel® Omni-Path Fabric Suite FastFabric Toolset host is connected to a single fabric. By default, the first active port on the FastFabric Toolset host is used to analyze the fabric. However, in more complex fabrics, the FastFabric Toolset host may be connected to more than one fabric or subnet. In this case, you can specify the ports or HFIs to use with one of the following methods:

- On the command line using the `-p` option.
- In a file specified using the `-t` option.
- Through the environment variables `PORTS` or `PORTS_FILE`.
- Using the `PORTS_FILE` configuration option in `opafastfabric.conf`.

If the specified port does not exist or is empty, the first active port on the local system is used. In more complex configurations, you must specify the exact ports to use for all fabrics to be analyzed. For more information, refer to [Selection of Devices](#) on page 41.

You can specify the `topology_input` file to be used with one of the following methods:

- On the command line using the `-T` option.
- In a file specified through the environment variable `FF_TOPOLOGY_FILE`.
- Using the `ff_topology_file` configuration option in `opafastfabric.conf`.



If the specified file does not exist, no `topology_input` file is used. Alternately the filename can be specified as `NONE` to prevent use of an input file.

For more information, refer to [opareport](#) on page 171.

By default, the error analysis includes PMA counters and slow links (that is, links running below enabled speeds). You can change this using the `FF_FABRIC_HEALTH` configuration parameter in `opafastfabric.conf`. This parameter specifies the `opareport` options and reports to be used for the health analysis. It also can specify the PMA counter clearing behavior (`-I seconds`, `-C`, or none at all). See the [FastFabric Configuration File](#) on page 57 for more information.

When a `topology_input` file is used, it can also be useful to extend `FF_FABRIC_HEALTH` to include fabric topology verification options such as `-o verifylinks`.

The thresholds for PMA counter analysis default to `/etc/opa/opamon.conf`. However, you can specify an alternate configuration file for thresholds using the `-c` option. The `opamon.si.conf` file can also be used to check for any non-zero values for signal integrity (SI) counters.

All files generated by `opafabricanalysis` start with `fabric` in their file name. This is followed by the port selection option identifying the port used for the analysis. Default is `0:0`.

The `opafabricanalysis` tool generates files such as the following within `FF_ANALYSIS_DIR`:

Health Check

- `latest/fabric.0:0.errors`
stdout of `opareport` for errors encountered during fabric error analysis.
- `latest/fabric.0:0.errors.stderr`
stderr of `opareport` during fabric error analysis.

Baseline

During a baseline run, the following files are also created in `FF_ANALYSIS_DIR/latest`.

- `baseline/fabric.0:0.snapshot.xml`
opareport snapshot of complete fabric components and SMA configuration.
- `baseline/fabric.0:0.comps`
opareport summary of fabric components and basic SMA configuration.
- `baseline/fabric.0:0.links`
opareport summary of internal and external links.

Full Analysis

- `latest/fabric.0:0.snapshot.xml`
opareport snapshot of complete fabric components and SMA configuration.



- `latest/fabric.0:0.snapshot.stderr`
stderr of opareport during snapshot.
- `latest/fabric.0:0.errors`
stdout of opareport for errors encountered during fabric error analysis.
- `latest/fabric.0:0.errors.stderr`
stderr of opareport during fabric error analysis.
- `latest/fabric.0:0.comps`
stdout of opareport for fabric components and SMA configuration.
- `latest/fabric.0:0.comps.stderr`
stderr of opareport for fabric components.
- `latest/fabric.0:0.comps.diff`
diff of baseline and latest fabric components.
- `latest/fabric.0:0.links`
stdout of opareport summary of internal and external links.
- `latest/fabric.0:0.links.stderr`
stderr of opareport summary of internal and external links.
- `latest/fabric.0:0.links.diff`
diff of baseline and latest fabric internal and external links.
- `latest/fabric.0:0.links.changes.stderr`
stderr of opareport comparison of links.
- `latest/fabric.0:0.links.changes`
opareport comparison of links against baseline. This is typically easier to read than the `links.diff` file and contains the same information.
- `latest/fabric.0:0.comps.changes.stderr`
stderr of opareport comparison of components.
- `latest/fabric.0:0.comps.changes`
opareport comparison of components against baseline. This is typically easier to read than the `comps.diff` file and contains the same information.

The `.diff` and `.changes` files are only created if differences are detected.

If the `-s` option is used and failures are detected, files related to the checks that failed are also copied to the time-stamped directory name under `FF_ANALYSIS_DIR`.

Fabric Items Checked Against the Baseline

Based on `opareport -o links`:

- Unconnected/down/missing cables
- Added/moved cables
- Changes in link width and speed



- Changes to Node GUIDs in fabric (replacement of HFI or Switch hardware)
- Adding/Removing Nodes [FI, Virtual FIs, Virtual Switches, Physical Switches, Physical Switch internal switching cards (leaf/spine)]
- Changes to server or switch names

Based on `opareport -o comps:`

- Overlap with items from links report
- Changes in port MTU, LMC, number of VLS
- Changes in port speed/width enabled or supported
- Changes in HFI or switch device IDs/revisions/VendorID (for example, ASIC hardware changes)
- Changes in port Capability mask (which features/agents run on port/server)
- Changes to ErrorLimits and PKey enforcement per port
- Changes to IOUs/IOCs/IOC Services provided

Note: Only applicable if IOUs in fabric (such as Virtual IO cards, native storage, and others).

Location (port, node) and number of SMs in fabric. Includes:

- Primary and backups
- Configured priority for SM

Fabric Items Also Checked During Health Check

Based on `opareport -s -C -o errors -o slowlinks:`

- PMA error counters on all Intel® Omni-Path Fabric ports (HFI, switch external and switch internal) checked against configurable thresholds.
 - Counters are cleared each time a health check is run. Each health check reflects a counter delta since last health check.
 - Typically identifies potential fabric errors, such as symbol errors.
 - May also identify transient congestion, depending on the counters that are monitored.
- Link active speed/width as compared to Enabled speed.
 - Identifies links whose active speed/width is < min (enabled speed/width on each side of link).
 - This typically reflects bad cables or bad ports or poor connections.
- Side effect is the verification of SA health.

5.2.4 opachassisanalysis

(Switch) Performs analysis of the chassis.

The `opachassisanalysis` tool checks the following for the Intel® Omni-Path Fabric Chassis:

- Chassis configuration (as reported by the chassis commands specified in `FF_CHASSIS_CMDS` in `opafastfabric.conf`).



- Chassis health (as reported by the chassis command specified in FF_CHASSIS_HEALTH in opafastfabric.conf).

Syntax

```
opachassisanalysis [-b|-e] [-s] [-d dir] [-F chassisfile]
[-H 'chassis']
```

Options

<code>--help</code>	Produces full help text.
<code>-b</code>	Specifies the baseline mode. Default is the compare/check mode.
<code>-e</code>	Evaluates health only. Default is the compare/check mode.
<code>-s</code>	Saves history of failures (errors/differences).
<code>-d dir</code>	Specifies the top-level directory for saving baseline and history of failed checks. Default = /var/usr/lib/opa/analysis
<code>-F chassisfile</code>	Specifies the file with the chassis in the cluster. Default = /etc/opa/chassis
<code>-H 'chassis'</code>	Specifies the list of chassis on which to execute the command.

Example

```
opachassisanalysis
```

Environment Variables

The following environment variables are also used by this command:

CHASSIS	List of chassis, used if <code>-F</code> and <code>-H</code> options are not supplied.
CHASSIS_FILE	File containing list of chassis, used if <code>-F</code> and <code>-H</code> options are not supplied.
FF_ANALYSIS_DIR	Top-level directory for baselines and failed health checks.
FF_CHASSIS_CMDS	List of commands to issue during analysis, unused if <code>-e</code> option supplied.
FF_CHASSIS_HEALTH	Single command to issue to check overall health during analysis, unused if <code>-b</code> option supplied.



Details

Intel recommends that you set up SSH keys for chassis (see [opasetupssh](#) on page 249). If SSH keys are not set up, all chassis must be configured with the same admin password and the password must be kept in the `/etc/opa/opafastfabric.conf` configuration file.

The default set of `FF_CHASSIS_CMDS` is:

```
showInventory fwVersion showNodeDesc timeZoneConf timeDSTConf
snmpCommunityConf snmpTargetAddr showChassisIpAddr showDefaultRoute
```

The commands specified in `FF_CHASSIS_CMDS` must be simple commands with no arguments. The output of these commands are compared to the baseline using `FF_DIFF_CMD`. Therefore, commands that include dynamically changing values, such as port packet counters, should not be included in this list.

`FF_CHASSIS_HEALTH` can specify one command (with arguments) to be used to check the chassis health. For chassis with newer firmware, the `hwCheck` command is recommended. For chassis with older firmware, a benign command, such as `fruInfo`, should be used. The default is `hwCheck`. Note that only the exit status of the `FF_CHASSIS_HEALTH` command is checked. The output is not captured and compared in a snapshot. However, on failure its output is saved to aid diagnosis.

The `opachassisanalysis` tool performs its analysis against one or more chassis in the fabric. As such, it permits the chassis to be specified using the `-H`, `-F`, `CHASSIS`, `chassis_file` or `opafastfabric.conf`. The handling of these options and settings is comparable to `opacmdall -C` and similar FastFabric Toolset commands against a chassis.

All files generated by `opafabricanalysis` start with `chassis.` in the file name.

The `opachassisanalysis` tool generates files such as the following within `FF_ANALYSIS_DIR`. The actual file names reflect the individual chassis commands that have been configured through the `FF_CHASSIS_HEALTH` and `FF_CHASSIS_CMDS` parameters:

Health Check

- `latest/chassis.hwCheck`
Output of `hwCheck` command for all selected chassis

Baseline: During a baseline run, the following files are also created in `FF_ANALYSIS_DIR/latest`.

- `baseline/chassis.fwVersion`
Output of `fwVersion` command for all selected chassis.
- `baseline/chassis.showChassisIpAddr`
Output of the `showChassisIpAddr` command for all selected chassis.
- `baseline/chassis.showDefaultRoute`
Output of the `showDefaultRoute` command for all selected chassis.
- `baseline/chassis.showNodeDesc`



Output of the `showNodeDesc` command for all selected chassis.

- `baseline/chassis.showInventory`

Output of the `showInventory` command for all selected chassis.

- `baseline/chassis.snmpCommunityConf`

Output of the `snmpCommunityConf` command for all selected chassis.

- `baseline/chassis.snmpTargetAddr`

Output of the `snmpTargetAddr` command for all selected chassis.

- `baseline/chassis.timeDSTConf`

Output of the `timeDSTConf` command for all selected chassis.

- `baseline/chassis.timeZoneConf`

Output of the `timeZoneConf` command for all selected chassis.

Full Analysis: The following `.diff` files are only created if differences are detected.

- `latest/chassis.hwCheck`

Output of the `hwCheck` command for all selected chassis.

- `latest/chassis.fwVersion`

Output of the `fwVersion` command for all selected chassis.

- `latest/chassis.fwVersion.diff`

diff of the baseline and latest `fwVersion`.

- `latest/chassis.showChassisIpAddr`

Output of the `showChassisIpAddr` command for all selected chassis.

- `latest/chassis.showChassisIpAddr.diff`

diff of baseline and latest `showChassisIpAddr`.

- `latest/chassis.showDefaultRoute`

Output of the `showDefaultRoute` command for all selected chassis.

- `latest/chassis.showDefaultRoute.diff`

diff of the baseline and the latest `showDefaultRoute`.

- `latest/chassis.showNodeDesc`

Output of the `showNodeDesc` command for all selected chassis.

- `latest/chassis.showNodeDesc.diff`

diff of the baseline and latest `showNodeDesc`.

- `latest/chassis.showInventory`

Output of the `showInventory` command for all selected chassis.

- `latest/chassis.showInventory.diff`

diff of the baseline and latest `showInventory`.

- `latest/chassis.snmpCommunityConf`

Output of the `snmpCommunityConf` command for all selected chassis.



- `latest/chassis.snmpCommunityConf.diff`
diff of the baseline and latest `snmpCommunityConf`.
- `latest/chassis.snmpTargetAddr`
Output of the `snmpTargetAddr` command for all selected chassis.
- `latest/chassis.snmpTargetAddr.diff`
diff of the baseline and latest `snmpTargetAddr`.
- `latest/chassis.timeDSTConf`
Output of the `timeDSTConf` command for all selected chassis.
- `latest/chassis.timeDSTConf.diff`
diff of the baseline and latest `timeDSTConf`.
- `latest/chassis.timeZoneConf`
Output of the `timeZoneConf` command for all selected chassis.
- `latest/chassis.timeZoneConf.diff`
diff of the baseline and latest `timeZoneConf`.

If the `-s` option is used and failures are detected, files related to the checks that failed are also copied to a time-stamped directory name under `FF_ANALYSIS_DIR`.

Chassis Items Checked Against the Baseline

Based upon `showInventory`:

- Addition/removal of Chassis FRUs
Replacement is only checked for FRUs that `showInventory` displays the serial number.
- Removal of redundant FRUs (spines, power supply, fan)

Based upon `fwVersion`:

- Changes to primary or alternate FW versions installed in cards in chassis.

Based upon `showNodeDesc`:

- Changes to configured node description for chassis. Note changes detected here would also be detected in fabric level analysis.

Based upon `timeZoneConf` and `timeDSTConf`:

- Changes to the chassis time zone and daylight savings time configuration.

Based upon `snmpCommunityConf` and `snmpTargetAddr`:

- Changes to SNMP persistent configuration within the chassis.

The following Chassis items are not checked against baseline:

- Changes to the chassis configuration on the management LAN (for example, `showChassisIpAddr`, `showDefaultRoute`). Such changes typically result in the chassis not responding on the LAN at the expected address that is detected by failures that perform other chassis checks.



Chassis Items Also Checked During Health Check

Based upon `hwCheck`:

- Overall health of FRUs in chassis:
 - Status of Fans in chassis
 - Status of Power Supplies in chassis
 - Temp/Voltage for each card
- Presence of adequate power/cooling of FRUs
- Presence of N+1 power/cooling of FRUs
- Presence of Redundant AC input

5.2.5 opahostsmanalysis

(All) Performs analysis against the local server only. It is assumed that both the host SM and the FastFabric are installed on the same system.

The host SM analysis tool checks the following:

- Host SM software version
- Host SM configuration file (simple text compare using `FF_DIFF_CMD`)
- Host SM health (for example, is it running?)

Syntax

```
opahostsmanalysis [-b|-e] [-s] [-d dir]
```

Options

- `--help` Produces full help text.
- `-b` Specifies the baseline mode. Default is the compare/check mode.
- `-e` Evaluates health only. Default is the compare/check mode.
- `-s` Saves history of failures (errors/differences).
- `-d dir` Specifies the top-level directory for saving baseline and history of failed checks. Default = `/var/usr/lib/opa/analysis`

Example

```
opahostsmanalysis
```

Environment Variables

The following environment variables are also used by this command:

`FF_ANALYSIS_DIR` Top-level directory for baselines and failed health checks.



FF_CURTIME	Timestamp to use on the directory created in FF_DIFF_CMD.
FF_DIFF_CMD	Linux* command to use to compare baseline to latest snapshot.

Details

All files generated by `opahostsmanalysis` start with `hostsm` in the file name.

The `opahostsmanalysis` tool generates files such as the following within `FF_ANALYSIS_DIR`. The actual file names reflect the individual chassis commands that have been configured using the `FF_CHASSIS_HEALTH` and `FF_CHASSIS_CMDS` parameters:

Health Check

- `latest/hostsm.smstatus` – Output of the `sm_query smShowStatus` command.

Baseline

- `baseline/hostsm.smver` – Host SM version.
- `baseline/hostsm.smconfig` – Copy of `opafm.xml`.

During a baseline run, the files are also created in `FF_ANALYSIS_DIR/latest`.

Full Analysis

- `latest/hostsm.smstatus` – Output of the `sm_query smShowStatus` command.
- `latest/hostsm.smver` – Host SM version.
`latest/hostsm.smver.diff` – diff of the baseline and latest host SM version.
- `latest/hostsm.smconfig` – Copy of `opafm.xml`.
- `latest/hostsm.smconfig.diff` – diff of the baseline and the latest `opafm.xml`.

The `.diff` files are only created if differences are detected.

If the `-s` option is used and failures are detected, files related to the checks that failed are also copied to a time-stamped directory name under `FF_ANALYSIS_DIR`.

Host SM Items Checked Against the Baseline

- SM configuration file.
- Version of the SM rpm installed on the system.

Host SM Items Also Checked During Health Check

- The SM is in the running state.

5.2.6 opaesmanalysis

(Switch) Performs analysis of the embedded Subnet Manager (SM) for configuration and health. The `opaesmanalysis` tool checks the `opafm.xml` file for the chassis.



All files generated by `opaesmanalysis` start with `esm` in the file name.

Intel recommends that you set up SSH keys for chassis (see [opasetupssh](#) on page 249). If SSH keys are not set up, all chassis must be configured with the same admin password and the password must be kept in the `opafastfabric.conf` configuration file.

Syntax

```
opaesmanalysis [-b|-e] [-s] [-d dir] [-G esmchassisfile]
[-E 'esmchassis']
```

Options

<code>--help</code>	Produces full help text.
<code>-b</code>	Specifies the baseline mode. Default is the compare/check mode.
<code>-e</code>	Evaluates health only. Default is the compare/check mode.
<code>-s</code>	Saves history of failures (errors/differences).
<code>-d dir</code>	Specifies the top-level directory for saving baseline and history of failed checks. Default = <code>/var/usr/lib/opa/analysis</code>
<code>-G esmchassisfile</code>	Specifies the file with SM chassis in the cluster. Default = <code>/etc/opa/esm_chassis</code>
<code>-E 'esmchassis'</code>	Specifies the list of SM chassis on which to execute the command.

Example

```
opaesmanalysis
```

Environment Variables

The following environment variables are also used by this command:

<code>ESM_CHASSIS</code>	List of SM chassis, used if <code>-G</code> and <code>-E</code> options are not supplied.
<code>ESM_CHASSIS_FILE</code>	File containing list of SM chassis, used if <code>-G</code> and <code>-E</code> options are not supplied.
<code>FF_ANALYSIS_DIR</code>	Top-level directory for baselines and failed health checks.



5.2.7 opaallanalysis

(All) opaallanalysis command performs the set of analysis specified in FF_ALL_ANALYSIS and can be specified for fabric, chassis, esm, or hostsm.

Syntax

```
opaallanalysis [-b|-e] [-s] [-d dir] [-c file]
[-t portsfile] [-p ports]
[-F chassisfile] [-H 'chassis']
[-G esmchassisfile] [-E esmchassis]
[-T topology_input]
```

Options

--help	Produces full help text.
-b	Sets the baseline mode. Default is compare/check mode.
-e	Evaluates health only. Default is compare/check mode.
-s	Saves history of failures (errors/differences).
-d dir	Identifies the top-level directory for saving baseline and history of failed checks. Default = /var/usr/lib/opa/analysis
-c file	Specifies the error thresholds configuration file. Default = /etc/opa/opamon.conf
-t portsfile	Specifies the file with list of local HFI ports used to access fabric(s) for analysis. Default = /etc/opa/ports
-p ports	Specifies the list of local HFI ports used to access fabric(s) for analysis. Default is the first active port. Specified as HFI:port as follows: 0:0 First active port in system. 0:y Port y within system. x:0 First active port on HFI x. x:y HFI x, port y.
-F chassisfile	Specifies the file with a chassis in a cluster. Default = /etc/opa/chassis
-H 'chassis'	Specifies the list of chassis on which to execute the command.



- `-G esmchassisfile` Specifies the file with embedded SM chassis in the cluster.
Default = `/etc/opa/esm_chassis`
- `-E esmchassis` Specifies the list of embedded SM chassis to analyze.
- `-T topology_input` Specifies the name of topology input file to use. Any `%P` markers in this filename are replaced with the `HFI:port` being operated on, such as `0:0` or `1:2`. Default = `/etc/opa/topology.%P.xml`. If `-T NONE` is specified, no topology input file is used. See [opareport](#) on page 171 for more information.

Example

```
opaallanalysis  
opaallanalysis -p '1:1 2:1'
```

Environment Variables

The following environment variables are also used by this command:

PORTS	List of ports, used in absence of <code>-t</code> and <code>-p</code> .
PORTS_FILE	File containing list of ports, used in absence of <code>-t</code> and <code>-p</code> .
FF_TOPOLOGY_FILE	File containing <code>topology_input</code> (may have <code>%P</code> marker in filename), used in absence of <code>-T</code> .
CHASSIS	List of chassis, used if <code>-F</code> and <code>-H</code> options are not supplied.
CHASSIS_FILE	File containing list of chassis, used if <code>-F</code> and <code>-H</code> options are not supplied.
ESM_CHASSIS	List of SM chassis, used if <code>-G</code> and <code>-E</code> options are not supplied.
ESM_CHASSIS_FILE	File containing list of SM chassis, used if <code>-G</code> and <code>-E</code> options are not supplied.
FF_ANALYSIS_DIR	Top level directory for baselines and failed health checks.
FF_CHASSIS_CMDS	List of commands to issue during analysis, unused if <code>-e</code> option supplied.
FF_CHASSIS_HEALTH	Single command to issue to check overall health during analysis, unused if <code>-b</code> option supplied

Details

The `opaallanalysis` command performs the set of analysis specified in `FF_ALL_ANALYSIS`, which must be a space-separated list. This can be provided by the environment or using `/etc/opa/opafastfabric.conf`. The analysis set



includes the options: `fabric`, `chassis`, `esm`, or `hostsm`. For a fabric with only externally managed switches, `FF_ALL_ANALYSIS` should be set to `-fabric` in `opafastfabric.conf`.

Note that the `opaallanalysis` command has options which are a super-set of the options for all other analysis commands. The options are passed along to the respective tools as needed. For example, the `-c` file option is passed on to `opafabricanalysis` if it is specified in `FF_ALL_ANALYSIS`.

The output files are all the output files for the `FF_ALL_ANALYSIS` selected set of analysis. See the previous sections for the specific output files.

5.2.8 Manual and Automated Usage

There are two basic ways to use the tools:

- **Manual**
Run the tools manually when trying to diagnose problems, or when you want to validate the fabric configuration and health.
- **Automated**
Run `opaallanalysis` or a specific tool in an automated script (such as a `cron` job). When run in this mode, the `-s` option may prove useful, but care must be taken to avoid excessive saved failures. When run in automated mode, Intel recommends you use a frequency of no faster than hourly. For many fabrics, a daily run or perhaps every few hours is sufficient. Because the exit code from each of the tools indicates the overall success/failure, an automated script can easily check the exit status. If failure occurs, an e-mail of the output can be sent from the analysis tool to the appropriate administrators for further analysis and corrective action.

- Notes:** Running these tools too often can have negative impacts. Among the potential risks:
- Each run adds a potential burden to the SM, fabric, and switches. For infrequent runs (hourly or daily), this impact is negligible. However, if this were to be run very frequently, the impacts to fabric and SM performance can be noticeable.
 - Runs with the `-s` option consume additional disk space for each run that identifies an error. The amount of disk space varies depending on fabric size. For a larger fabric, this can be on the order of 1-40 MB. Therefore, care must be taken not to run the tools too often and to visit and clean out the `FF_ANALYSIS_DIR` periodically. If the `-s` option is used during automated execution of the health check tools, it may be helpful to also schedule automated disk space checks, for example, as a `cron` job.
 - Runs coinciding with down time for selected components, (such as servers that are offline or rebooting, are considered failures and generate the resulting failure information. If the runs are not carefully scheduled, this data could be misleading and also waste disk space.

5.2.9 Re-Establishing Health Check Baseline

Intel recommends you establish a baseline after you change the fabric configuration. The following activities are examples of ways in which the fabric configuration may be changed:



- Repair a faulty board, which leads to a new serial number for that component.
- Update switch firmware or Fabric Manager.
- Change time zones in a switch.
- Add or delete a new device or link to a fabric.
- Remove a failed link and its devices from the Fabric Manager database.

Perform the following procedure to re-establish the health check baseline:

1. Make sure that you have fixed all problems with the fabric, including inadvertent configuration changes, before proceeding.
2. Verify that the fabric configured is as expected. The simplest way to do this is to run `opafabricinfo` which returns information for each subnet to which the fabric management server is connected. The following is an example output for a single subnet.

```
# opafabricinfo
Fabric 0:0 Information:
SM: hds1fnb6241 hfil_0 Guid: 0x0011750101575ffe State: Master
Number of HFIs: 8
Number of Switches: 1
Number of Links: 8
Number of HFI Links: 8 (Internal: 0 External: 8)
Number of ISLs: 0 (Internal: 0 External: 0)
Number of Degraded Links: 0 (HFI Links: 0 ISLs: 0)
Number of Omitted Links: 0 (HFI Links: 0 ISLs: 0)
```

3. Save the old baseline because it may be required for future debug. The old baseline is a group of files in `/var/usr/lib/opa/analysis/baseline`.
4. Run `opaallanalysis -b`
5. Check the new output files in `/var/usr/lib/analysis/baseline` to verify that the configuration is as you expect it.

5.2.10 Interpreting the Health Check Results

When any of the health check tools are run, the overall success or failure is indicated in the output of the tool and its exit status. The tool also indicates which areas had problems and which files should be reviewed. The results from the latest run can be found in `FF_ANALYSIS_DIR/latest/`. This directory includes the latest configuration of the fabric and any errors/differences found during the health check.

If the `-s` option was used when running the health check, a directory whose name is the date and time of the failing run is created under `FF_ANALYSIS_DIR`. In this case, refer to that directory instead of the `latest` directory shown in the following examples.

Intel recommends that you review the results for any ESM or SM health check failures. If the SM is misconfigured or not running, it can cause other health checks to fail. In this case, correct the SM problems first, then rerun the health check.

For a host SM analysis, review the files in the following order:

1. `latest/hostsm.smstatus`



Make sure this file indicates the SM is running. If no SMs are running on the fabric, correct that problem before proceeding further. Once corrected, rerun the health checks to look for further errors.

2. `latest/hostsm.smver.diff`

Indicates the SM version has changed. If this was not an expected change, correct the SM before proceeding further. Once corrected, rerun the health checks to look for further errors. If the change was expected and permanent, rerun a baseline once all other health check errors have been corrected.

3. `latest/hostsm.smconfig.diff`

Indicates that the SM configuration has changed. Review this file and compare the `latest/hostsm.smconfig` file to `baseline/hostsm.smconfig`. Correct the SM configuration, if necessary. Once corrected, rerun the health checks to look for further errors. If the change was expected and permanent, rerun a baseline once all other health check errors have been corrected.

For an ESM analysis, the `FF_ESM_CMDS` configuration setting selects which ESM commands are used for the analysis. When using the default setting for this parameter, review the files in the following order:

1. `latest/esm.smstatus`

Make sure this indicates the SM is running. If no SMs are running on the fabric, correct the problem before proceeding further. Once corrected, rerun the health checks to look for further errors.

2. `latest/esm.CHASSIS.opafm.xml`

The `opafm.xml` file for the given chassis.

3. `latest/esm.CHASSIS.opafm.xml.diff`

Indicates that the SM configuration has changed. Review this file and compare the `latest/esm.CHASSIS.opafm.xml` file to `baseline/esm.CHASSIS.opafm.xml`. Correct the SM configuration, if necessary. Once corrected, rerun the health checks to look for further errors. If the change was expected and permanent, rerun a baseline once all other health check errors have been corrected.

4. `latest/esm.smShowSMParms.diff`

Indicates that the SM configuration has changed. Review this file and compare the `latest/esm.smShowSMParms` file to `baseline/esm.smShowSMParms`. Correct the SM configuration, if necessary. Once corrected, rerun the health checks to look for further errors. If the change was expected and permanent, rerun a baseline once all other health check errors have been corrected.

5. `latest/esm.smShowDefBcGroup.diff`

Indicates that the SM broadcast group for IPoIB configuration has changed. Review this file and compare the `latest/esm.smShowDefBcGroup` file to `baseline/esm.smShowDefBcGroup`. Correct the SM configuration, if necessary. Once corrected, rerun the health checks to look for further errors. If the change was expected and permanent, rerun a baseline once all other health check errors have been corrected.

6. `latest/esm/*.diff`



If `FF_ESM_CMDS` has been modified, review the changes in results for those additional commands. Correct the SM configuration, if necessary. Once corrected, rerun the health checks to look for further errors. If the change was expected and permanent, rerun a baseline once all other health check errors have been corrected.

Next, review the results of the fabric analysis for each configured fabric. If nodes or links are missing, the fabric analysis detects them. Missing links or nodes can cause other health checks to fail. If such failures are expected (for example, a node or switch is offline), you can perform further review of result files, however, be aware that the loss of the node or link can cause other analysis to also fail.

The following discussion presents the analysis order for `fabric.0.0`. If other or additional fabrics are configured for analysis, review the files in the order shown for each fabric. There is no specific order recommended for which fabric to review first.

1. `latest/fabric.0.0.errors.stderr`

If this file is not empty, it can indicate problems with `opareport`, such as inability to access an SM. This may result in unexpected problems or inaccuracies in the related errors file. Correct problems reported in this file first. Once corrected, rerun the health checks to look for further errors.

2. `latest/fabric.0:0.errors`

If any links with excessive error rates or incorrect link speeds are reported, correct them. If there are links with errors, beware the same links may also be detected in other reports such as the links and comps files.

3. `latest/fabric.0.0.snapshot.stderr`

If this file is not empty, it can indicate problems with `opareport`, such as inability to access an SM. This may result in unexpected problems or inaccuracies in the related links and comps files. Correct problems reported in this file first. Once corrected, rerun the health checks to look for further errors.

4. `latest/fabric.0:0.links.stderr` and `latest/fabric.0:0.links.changes.stderr`

If these files are not empty, it can indicate problems with `opareport` which can result in unexpected problems or inaccuracies in the related links files. Correct problems reported in this file first. Once corrected, rerun the health checks to look for further errors. For more information on `.changes` files, refer to [Interpreting Health Check .changes Files](#) on page 151.

5. `latest/fabric.0:0.links.diff` and `latest/fabric.0:0.links.changes`

These indicate that the links between components in the fabric have changed, been removed/added, or that components in the fabric have disappeared. If both files are available, use the `fabric.0:0.links.changes` file since it has a more concise and precise description of the fabric link changes. Compare the `latest/fabric.0:0.links` file to `baseline/fabric.0:0.links`. If components have disappeared, review the `latest/fabric.0:0.comps.diff` and `latest/fabric.0:0.comps.changes` files. Correct missing nodes and links, if necessary. Once corrected, rerun the health checks to look for further errors. If the change was expected and is permanent, rerun a baseline once all other health check errors have been corrected. For more information on `.changes` files, refer to [Interpreting Health Check .changes Files](#) on page 151.



6. `latest/fabric.0:0.comps.stderr` and `latest/fabric.0:0.comps.changes.stderr`

If these files are not empty, it can indicate problems with `opareport` which can result in unexpected problems or inaccuracies in the related `comps` file. Correct problems reported in these files first. Once corrected, rerun the health checks to look for further errors. For more information on `.changes` files, refer to [Interpreting Health Check .changes Files](#) on page 151.

7. `latest/fabric.0:0.comps.diff` and `latest/fabric.0:0.comps.changes`

These indicate that the components in the fabric or their SMA configuration have changed. If both files are available, use the `fabric.0:0.comps.changes` file since it has a more concise and precise description of the fabric component changes. Compare the `latest/fabric.0:0.comps` file to `baseline/fabric.0:0.comps`. Correct missing nodes, missing SMs, ports that are down, and port misconfigurations, if necessary. Once corrected, rerun the health checks to look for further errors. If the change was expected and permanent, rerun a baseline once all other health check errors have been corrected. For more information on `.changes` files, refer to [Interpreting Health Check .changes Files](#) on page 151.

Review the results of the `opachassisanalysis`. If chassis configuration has changed, the `opachassisanalysis` report detects it. Previous checks should have already detected missing chassis, missing or added links and many aspects of chassis configuration. For `opachassisanalysis`, the `FF_CHASSIS_CMDS` and `FF_CHASSIS_HEALTH` configuration settings select which chassis commands are used for the analysis. When using the default setting for this parameter, review the files in the following order:

1. `latest/chassis.hwCheck`

Make sure this indicates all chassis are operating properly with the desired power and cooling redundancy. If there are problems, correct them, but other analysis files can be analyzed first. Once any problems are corrected, rerun the health checks to verify the correction.

2. `latest/chassis.fwVersion.diff`

Indicates the chassis firmware version has changed. If this was not an expected change, correct the chassis firmware before proceeding further. Once corrected, rerun the health checks to look for further errors. If the change was expected and permanent, rerun a baseline once all other health check errors have been corrected.

3. `latest/chassis.*.diff`

These files reflect other changes to chassis configuration based on checks selected by `FF_CHASSIS_CMDS`. Review the changes in results for these remaining commands. Correct the chassis, if necessary. Once corrected, rerun the health checks to look for further errors. If the change was expected and permanent, rerun a baseline once all other health check errors have been corrected.

5.2.11 Interpreting Health Check .changes Files

Files with the extension `.changes` summarize what has changed in a configuration based on the queries done by the health check.



This type of file uses the following format:

- [What is being verified]
- [Indication that something is not correct]
- [Items that are not correct and what is incorrect about them]
- [How many items were checked]
- [Total number of incorrect items]
- [Summary of how many items had particular issues]

The following example of `fabric.*:*.links.changes` only shows links that were “Unexpected”. That means that the link was not found in the previous baseline.

```
# cat latest/fabric.0:0.links.changes
Links Topology Verification

Links Found with incorrect configuration:
Rate NodeGUID          Port Type Name
100g 0x0011750101603593  1 FI  phs1fnivd13u07n3 hfi1_0
<-> 0x00117501026a5619 11 SW  phs1swivd13u21
Unexpected Link

4 of 4 Fabric Links Checked

Links Expected but Missing, Duplicate in input or Incorrect:
3 of 3 Input Links Checked

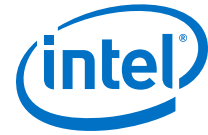
Total of 1 Incorrect Links found
0 Missing, 1 Unexpected, 0 Misconnected, 0 Duplicate, 0 Different
-----
```

The following table summarizes possible issues found in `.changes` files.

Table 14. Possible issues found in health check `.changes` files

Issue	Description and possible actions
Missing	<p>This indicates an item that is in the baseline, is not in this instance of health check output. This may indicate a broken item or a configuration change that has removed the item from the configuration.</p> <p>If you have intentionally removed this item from the configuration, save the original baseline and rerun the baseline. For example, if you've removed an HFI connection, the HFI and the link to it are shown as Missing in <code>fabric.*:*.links.changes</code> and <code>fabric.*:*.comps.changes</code> files.</p> <p>If the item is still part of the configuration, check for faulty connections or unintended changes to configuration files on the fabric management server.</p> <p>You should also look for any “Unexpected” or “Different” items that may correspond to this item. In some cases, the configuration of an item has changed in a way that makes it difficult to determine precisely how it has changed.</p>
Unexpected	<p>This indicates that an item is in this instance of health check output, but it is not in the baseline. This may indicate that an item was broken when the baseline was taken or a configuration change has added the item to the configuration.</p> <p>If you have added this item to the configuration, save the original baseline and rerun the baseline. For example, if you've added an HFI connection, it is shown as Unexpected in <code>fabric.*:*.links.changes</code> and <code>fabric.*:*.comps.changes</code> files.</p> <p>You should also look for any “Missing” or “Different” items that may correspond to this item. In some cases, the configuration of an item has changed in a way that makes it difficult to determine precisely how it has changed.</p>

continued...



Issue	Description and possible actions
Misconnected	<p>This only applies to links and indicates that a link is not connected properly. This should be fixed.</p> <p>It is possible to find miswires by examining all of the Misconnected links in the fabric. However, you must look at all of the <code>fabric.*:*.links.changes</code> files to find miswires between subnets.</p> <p>You should also look for any “Missing” or “Different” items that may correspond to this item. In some cases, the configuration of an item has changed in a way that makes it difficult to determine precisely how it has changed.</p> <p>Individual links that are Misconnected are reported as “Incorrect Link” and are added into the Misconnected summary count.</p>
Duplicate	<p>This indicates that an item has a duplicate in the fabric. This situation should be resolved so there is only one instance of any particular item being discovered in the fabric.</p> <p>This error can occur if there are changes in the fabric such as addition of parallel links. It can also be reported when there enough changes to the fabric that it is difficult to properly resolve and report all the changes. It can also occur when <code>opareport</code> is run with manually generated topology input files that may have duplicate items or incomplete specifications.</p>
Different	<p>This indicates that an item still exists in the current health check, but it is different from the baseline configuration.</p> <p>If the configuration has changed purposely since the most recent baseline, and the expected difference is reflected here, save the original baseline and rerun the baseline.</p> <p>If this difference was not intended, you must rectify the difference to prevent future health checks from reporting the same difference from the baseline.</p> <p>You should also look for any “Missing” or “Unexpected” items that may correspond to this item. In some cases, the configuration of an item has changed in a way that makes it difficult to determine precisely how it has changed.</p> <p>Individual items that are Different are reported as “Mismatched” or “Inconsistent” and are added into the Different summary count.</p>
Port Attributes Inconsistent	<p>This indicates that the attributes of a port on one side of a link have changed, such as PortGuid, Port Number, Device Type, or others. The inconsistency is caused by connecting a different type of device or a different instance of the same device type. This may also occur after replacing a faulty device.</p> <p>If the configuration has changed purposely since the most recent baseline, and this difference is reflected here, save the original baseline and rerun the baseline. If a faulty device was replaced, it is important to re-establish the baseline.</p> <p>If this difference was not intended, you must rectify the difference to prevent future health checks from reporting the same difference from the baseline.</p> <p>This is a specific case of “Different”.</p>
Node Attributes Inconsistent	<p>This indicates that the attributes of a node in the fabric have changed, such as NodeGuid, Node Description, Device Type, or others. The inconsistency is caused by connecting a different type of device or a different instance of the same device type. This may also occur after replacing a faulty device.</p> <p>If the configuration has changed purposely since the most recent baseline, and this difference is reflected here, save the original baseline and rerun the baseline. If a faulty device was replaced, it is important to re-establish the baseline.</p> <p>If this difference was not intended, you must rectify the difference to prevent future health checks from reporting the same difference from the baseline.</p> <p>This is a specific case of “Different”.</p>
SM Attributes Inconsistent	<p>This indicates that the attributes of the node or port running an SM in the fabric have changed, such as NodeGuid, Node Description, Port Number, Device Type, or others. The inconsistency is caused by moving a cable, changing from host-based subnet management to embedded subnet management (or vice-versa), or by replacing the HFI in the fabric management server.</p> <p>If the configuration has changed purposely since the most recent baseline, and this difference is reflected here, save the original baseline and rerun the baseline. If the HFI in the fabric management server was replaced, it is important to re-establish the baseline.</p>

continued...



Issue	Description and possible actions
	If this difference was not intended, you must rectify the difference to prevent future health checks from reporting the same difference from the baseline. This is a specific case of "Different".
X mismatch: expected ... found:	This indicates an aspect of an item has changed as compared to the baseline configuration. The aspect that changed and the expected and found values are shown. This typically indicates configuration differences such as MTU, Speed, and Node Description. It can also indicate that GUIDs have changed, such as replacing a faulty device with a comparable device. If the configuration has changed purposely since the most recent baseline, and this difference is reflected here, save the original baseline and rerun the baseline. If a faulty device was replaced, it is important to re-establish the baseline. If this difference was not intended, you must rectify the difference to prevent future health checks from reporting the same difference from the baseline. This is a specific case of "Different".
Incorrect Link	This only applies to links and indicates that a link is not connected properly. This should be fixed. It is possible to find miswires by examining all of the Misconnected links in the fabric. However, you must look at all of the <code>fabric.*:*.links.changes</code> files to find miswires between subnets. You should also look for any "Missing" or "Different" items that may correspond to this item. In some cases, the configuration of an item has changed in a way that makes it difficult to determine precisely how it has changed. This is a specific case of "Misconnected".

5.3 Verification, Analysis, and Control CLIs

The CLIs described in this section are used for fabric deployment verification, analysis, and control.

5.3.1 opacabletest

(Switch) Initiates or stops Cable Bit Error Rate stress tests for Intel® Omni-Path Host Fabric Interface (HFI)-to-switch links and/or ISLs.

Syntax

```
opacabletest [-C|-A] [-c file] [-f hostfile]
[-h 'hosts'] [-n numprocs] [-t portsfile]
[-p ports] [start|start_fi|start_isl|stop|stop_fi| stop_isl] ...
```

Options

<code>--help</code>	Produces full help text.
<code>-C</code>	Clears error counters.
<code>-A</code>	Forces the system to clear hardware error counters. Implies <code>-C</code> .
<code>-c file</code>	Specifies the error thresholds configuration file. Default is <code>/etc/opa/opamon.si.conf</code> file. Only used if <code>-C</code> or <code>-A</code> specified.
<code>-f hostfile</code>	Specifies the file with hosts to include in HFI-to-SW test. Default is <code>/etc/opa/hosts</code> file.



<code>-h hosts</code>	Specifies the list of hosts to include in HFI-SW test.
<code>-n numprocs</code>	Specifies the number of processes per host for HFI-SW test.
<code>-t portsfile</code>	Specifies the file with list of local HFI ports used to access fabrics when clearing counters. Default is <code>/etc/opa/ports</code> file.
<code>-p ports</code>	Specifies the list of local HFI ports used to access fabrics for counter clear. Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format <code>hfi:port</code> , for example: <code>0:0</code> First active port in system. <code>0:y</code> Port <i>y</i> within system. <code>x:0</code> First active port on HFI <i>x</i> . <code>x:y</code> HFI <i>x</i> , port <i>y</i> .
<code>start</code>	Starts the HFI-SW and ISL tests.
<code>start_fi</code>	Starts the HFI-SW test.
<code>start_isl</code>	Starts the ISL test.
<code>stop</code>	Stops the HFI-SW and ISL tests.
<code>stop_fi</code>	Stops the HFI-SW test.
<code>stop_isl</code>	Stops the ISL test.

The HFI-SW cable test requires that the `FF_MPI_APPS_DIR` is set, and it contains a pre-built copy of the `mpi_apps` for an appropriate message passing interface (MPI).

The ISL cable test started by this tool assumes that the master Host Subnet Manager (HSM) is running on this host. If using the Embedded Subnet Manager (ESM), or if a different host is the master HSM, the ISL cable test must be controlled by the switch CLI, or by Intel® Omni-Path Fabric Suite FastFabric on the master HSM respectively.

Examples

```
opacabletest -A start
opacabletest -f good -A start
opacabletest -h 'arwen elrond' start_fi
HOSTS='arwen elrond' opacabletest stop
opacabletest -A
```

Environment Variables

The following environment variables are also used by this command:



HOSTS	List of hosts, used if <code>-h</code> option not supplied.
HOSTS_FILE	File containing list of hosts, used in absence of <code>-f</code> and <code>-h</code> .
PORTS	List of ports, used in absence of <code>-t</code> and <code>-p</code> .
PORTS_FILE	File containing list of ports, used in absence of <code>-t</code> and <code>-p</code> .
FF_MAX_PARALLEL	Maximum concurrent operations.

5.3.2 opaextractbadlinks

Produces a CSV file listing all or some of the links that exceed `opareport -o` error thresholds. `opaextractbadlinks` is a front end to the `opareport` tool. The output from this tool can be imported into a spreadsheet or parsed by other scripts.

Syntax

```
opaextractbadlinks [opareport options]
```

Options

<code>--help</code>	Produces full help text.
<code>opareport options</code>	The following options are passed to <code>opareport</code> . This subset is considered typical and useful for this command.
<code>-h/--hfi hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p/--port port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-X/--infile snapshot_input</code>	Generates a report using the data in the <code>snapshot_input</code> file. <code>snapshot_input</code> must have been generated during a previous <code>-o snapshot</code> run. When used, the <code>-s</code> , <code>-i</code> , <code>-C</code> , and <code>-a</code> options are ignored. <code>'-'</code> may be used as the <code>snapshot_input</code> to specify <code>stdin</code> .
<code>-T/--topology topology_input</code>	Uses <code>topology_input</code> file to augment and verify fabric information. When used, various reports can be augmented with information not available electronically (such as cable labels). <code>'-'</code> may be used to specify <code>stdin</code> .



<code>-i/--interval seconds</code>	Obtains performance statistics over interval <i>seconds</i> . Clears all statistics, waits interval <i>seconds</i> , then generates report. Implies <code>-s</code> option.
<code>-b/--begin date_time</code>	Obtains past performance stats over an interval beginning at <i>date_time</i> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago.
<code>-e/--end date_time</code>	Obtains past performance stats over an interval ending at <i>date_time</i> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago.
<code>-C/--clear</code>	Clears performance statistics for all ports. Only statistics with error thresholds are cleared. A clear occurs after generating the report.
<code>-a/--clearall</code>	Clears all performance statistics for all ports.
<code>-M/--pmadirect</code>	Accesses performance statistics using direct PMA.
<code>-A/--allports</code>	Gets PortInfo for down switch ports. Uses direct SMA to get this data. If used with <code>-M</code> , also gets PMA stats for down switch ports.
<code>-c/--config file</code>	Specifies the error thresholds configuration file. Default is <code>/etc/opa/opamon.conf</code> file.
<code>-L/--limit</code>	Limits operation to exact specified focus with <code>-F</code> for port error counters check (<code>-o errors</code>) and port counters clear (<code>-C</code> or <code>-i</code>). Normally, the neighbor of each selected port is also checked/cleared. Does not affect other reports.



`-F/--focus point` Specifies the focus area for report. Used for all reports except `route` to limit scope of report. Refer to [Point Syntax](#) on page 177 for details.

-h and -p options permit a variety of selections:

`-h 0` First active port in system (default).

`-h 0 -p 0` First active port in system.

`-h x` First active port on HFI x.

`-h x -p 0` First active port on HFI x.

`-h 0 -p y` Port y within system (no matter which ports are active).

`-h x -p y` HFI x, port y.

Examples

```
# List all the bad links in the fabric:
opaextractbadlinks

# List all the bad links to a switch named "coresw1":
opaextractbadlinks -F "node:coresw1"

# List all the bad links to end-nodes:
opaextractbadlinks -F "nodetype:FI"

# List all the bad links on the 2nd HFI's fabric of a multi-plane fabric:
opaextractbadlinks -h 2
```

5.3.3 opaextractlink

Produces a CSV file listing all or some of the links in the fabric. `opaextractlink` is a front end to the `opareport` tool. The output from this tool can be imported into a spreadsheet or parsed by other scripts.

Syntax

```
opaextractlink [opareport options]
```

Options

`--help` Produces full help text.

`opareport options` The following options are passed to `opareport`. This subset is considered typical and useful for this command.



<code>-h/--hfi hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p/--port port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-X/--infile snapshot_input</code>	Generates a report using the data in the <code>snapshot_input</code> file. <code>snapshot_input</code> must have been generated during a previous <code>-o snapshot</code> run. When used, the <code>-s</code> , <code>-i</code> , <code>-C</code> , and <code>-a</code> options are ignored. '-' may be used as the <code>snapshot_input</code> to specify stdin.
<code>-T/--topology topology_input</code>	Uses <code>topology_input</code> file to augment and verify fabric information. When used, various reports can be augmented with information not available electronically (such as cable labels). '-' may be used to specify stdin.

-h and -p options permit a variety of selections:

- `-h 0` First active port in system (default).
- `-h 0 -p 0` First active port in system.
- `-h x` First active port on HFI x.
- `-h x -p 0` First active port on HFI x.
- `-h 0 -p y` Port y within system (no matter which ports are active).
- `-h x -p y` HFI x, port y.

Examples

```
# List all the links in the fabric:
opaextractlink

# List all the links to a switch named "coresw1":
opaextractlink -F "node:coresw1"

# List all the links to end-nodes:
opaextractlink -F "nodetype:FI"

# List all the links on the 2nd HFI's fabric of a multi-plane fabric:
opaextractlink -h 2
```



5.3.4 opaextractmissinglinks

Produces a CSV file listing all or some of the links in the fabric.

`opaextractmissinglinks` is a front end to the `opareport` tool that generates a report listing all or some of the links that are present in the supplied topology file, but are missing in the fabric. The output from this tool can be imported into a spreadsheet or parsed by other scripts.

Syntax

```
opaextractmissinglinks [-T topology_input] [-o report]
[opareport options]
```

Options

<code>--help</code>	Produces full help text.
<code>-T/--topology topology_input</code>	Uses <code>topology_input</code> file to augment and verify fabric information. When used, various reports can be augmented with information not available electronically (such as cable labels). '-' may be used to specify <code>stdin</code> .
<code>-o/--output report</code>	Specifies the report type for output. Refer to Report Types for details.
<code>opareport options</code>	The following options are passed to <code>opareport</code> . This subset is considered typical and useful for this command.
<code>-h/--hfi hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p/--port port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-X/--infile snapshot_input</code>	Generates a report using the data in the <code>snapshot_input</code> file. <code>snapshot_input</code> must have been generated during a previous <code>-o snapshot</code> run. When used, the <code>-s</code> , <code>-i</code> , <code>-C</code> , and <code>-a</code> options are ignored. '-' may be used as the <code>snapshot_input</code> to specify <code>stdin</code> .
<code>-F/--focus point</code>	Specifies the focus area for report. Used for all reports except <code>route</code> to limit scope of report. Refer to Point Syntax on page 177 for details.



-h and -p options permit a variety of selections:

- h 0 First active port in system (default).
- h 0 -p 0 First active port in system.
- h x First active port on HFI x.
- h x -p 0 First active port on HFI x.
- h 0 -p y Port y within system (no matter which ports are active).
- h x -p y HFI x, port y.

Report Types

verifylinks	Compares fabric (or snapshot) links to supplied topology and identifies differences and omissions.
verifyextlinks	Compares fabric (or snapshot) links to supplied topology and identifies differences and omissions. Limits analysis to links external to systems.
verifyfilinks	Compares fabric (or snapshot) links to supplied topology and identify differences and omissions. Limits analysis to links to FIs.
verifyislinks	Compares fabric (or snapshot) links to supplied topology and identify differences and omissions. Limits analysis to inter-switch links.
verifyextislinks	Compares fabric (or snapshot) links to supplied topology and identify differences and omissions. Limits analysis to inter-switch links external to systems.
verifyall	Returns verifyfis, verifyfsws, verifyfsm, and verifylinks reports.

Examples

```
List all the missing links in the fabric:
opaextractmissinglinks

List all the missing links to a switch named "coresw1":
opaextractmissinglinks -T topology.0:0.xml -F "node:coresw1"

List all the missing connections to end-nodes:
opaextractmissinglinks -o verifyfilinks

List all the missing links on the 2nd HFI's fabric of a multi-plane fabric:
opaextractmissinglinks -h 2 -T /etc/opa/topology.2:1.xml

List all the missing links between two switches:
opaextractmissinglinks -o verifyislinks -T topology.0:0.xml
```



5.3.5 opaextractsellinks

Produces a CSV file listing all or some of the links in the fabric. `opaextractsellinks` is a front end to the `opareport` tool. The output from this tool can be imported into a spreadsheet or parsed by other scripts.

Syntax

```
opaextractsellinks [opareport options]
```

Options

<code>--help</code>	Produces full help text.
<code>opareport options</code>	The following options are passed to <code>opareport</code> . This subset is considered typical and useful for this command.
<code>-h/--hfi hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p/--port port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-o/--output report</code>	Specifies the report type for output. Refer to Report Types for details.
<code>-X/--infile snapshot_input</code>	Generates a report using the data in the <code>snapshot_input</code> file. <code>snapshot_input</code> must have been generated during a previous <code>-o snapshot</code> run. When used, the <code>-s</code> , <code>-i</code> , <code>-C</code> , and <code>-a</code> options are ignored. '-' may be used as the <code>snapshot_input</code> to specify stdin.
<code>-T/--topology topology_input</code>	Uses <code>topology_input</code> file to augment and verify fabric information. When used, various reports can be augmented with information not available electronically (such as cable labels). '-' may be used to specify stdin.
<code>-F/--focus point</code>	Specifies the focus area for report. Used for all reports except <code>route</code> to limit scope of report. Refer to Point Syntax on page 177 for details.

-h and -p options permit a variety of selections:

<code>-h 0</code>	First active port in system (default).
-------------------	--



- h 0 -p 0 First active port in system.
- h x First active port on HFI x.
- h x -p 0 First active port on HFI x.
- h 0 -p y Port y within system (no matter which ports are active).
- h x -p y HFI x, port y.

Report Types

verifylinks	Compares fabric (or snapshot) links to supplied topology and identifies differences and omissions.
verifyextlinks	Compares fabric (or snapshot) links to supplied topology and identifies differences and omissions. Limits analysis to links external to systems.
verifyfilinks	Compares fabric (or snapshot) links to supplied topology and identify differences and omissions. Limits analysis to links to FIs.
verifyislinks	Compares fabric (or snapshot) links to supplied topology and identify differences and omissions. Limits analysis to inter-switch links.
verifyextislinks	Compares fabric (or snapshot) links to supplied topology and identify differences and omissions. Limits analysis to inter-switch links external to systems.
verifyall	Returns verifyfis, verifysws, verifysms, and verifylinks reports.

Examples

```
# List all the links in the fabric:
opaextractsellinks

# List all the links to a switch named "coresw1":
opaextractsellinks -F "node:coresw1"

# List all the connections to end-nodes:
opaextractsellinks -F "nodetype:FI"

# List all the links on the 2nd HFI's fabric of a multi-plane fabric:
opaextractsellinks -h 2
```



5.3.6 opaextractstat2

Performs an error analysis of a fabric and provides augmented information from a `topology_file` including all error counters. The output is in a CSV format suitable for importing into a spreadsheet or parsed by other scripts. `opaextractstat2` is a front end to the `opareport` and `opaxmlextract` tools.

Syntax

```
opaextractstat2 topology_file [opareport options]
```

Options

<code>--help</code>	Produces full help text.
<code>topology_file</code>	Specifies <code>topology_file</code> to use.
<code>opareport options</code>	The following options are passed to <code>opareport</code> . This subset is considered typical and useful for this command.
<code>-h/--hfi hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p/--port port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-X/--infile snapshot_input</code>	Generates a report using the data in the <code>snapshot_input</code> file. <code>snapshot_input</code> must have been generated during a previous <code>-o snapshot</code> run. When used, the <code>-s</code> , <code>-i</code> , <code>-C</code> , and <code>-a</code> options are ignored. '-' may be used as the <code>snapshot_input</code> to specify <code>stdin</code> .
<code>-i/--interval seconds</code>	Obtains performance statistics over interval <code>seconds</code> . Clears all statistics, waits interval <code>seconds</code> , then generates report. Implies <code>-s</code> option.
<code>-b/--begin date_time</code>	Obtains past performance stats over an interval beginning at <code>date_time</code> . Implies <code>-s</code> option. <code>date_time</code> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example,



	"2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago.
<code>-e/--end <i>date_time</i></code>	Obtains past performance stats over an interval ending at <i>date_time</i> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago.
<code>-C/--clear</code>	Clears performance statistics for all ports. Only statistics with error thresholds are cleared. A clear occurs after generating the report.
<code>-a/--clearall</code>	Clears all performance statistics for all ports.
<code>-M/--pmadirect</code>	Accesses performance statistics using direct PMA.
<code>-A/--allports</code>	Gets PortInfo for down switch ports. Uses direct SMA to get this data. If used with <code>-M</code> , also gets PMA stats for down switch ports.
<code>-c/--config <i>file</i></code>	Specifies the error thresholds configuration file. Default is <code>/etc/opa/opamon.conf</code> file.
<code>-L/--limit</code>	Limits operation to exact specified focus with <code>-F</code> for port error counters check (<code>-o errors</code>) and port counters clear (<code>-C</code> or <code>-i</code>). Normally, the neighbor of each selected port is also checked/cleared. Does not affect other reports.
<code>-F/--focus <i>point</i></code>	Specifies the focus area for report. Used for all reports except <code>route</code> to limit scope of report. Refer to Point Syntax on page 177 for details.

-h and -p options permit a variety of selections:

`-h 0` First active port in system (default).



- h 0 -p 0 First active port in system.
- h x First active port on HFI x.
- h x -p 0 First active port on HFI x.
- h 0 -p y Port y within system (no matter which ports are active).
- h x -p y HFI x, port y.

The portion of the script that calls `opareport` and `opaxmlextract` follows:

```
opareport -x -d 10 -s -o errors -T $@ | opaxmlextract -d \;  
-e Rate -e MTU -e Internal -e LinkDetails -e CableLength -e CableLabel  
-e CableDetails -e Port.NodeGUID -e Port.PortGUID -e Port.PortNum  
-e Port.PortType -e Port.NodeDesc -e Port.PortDetails  
-e PortXmitData.Value -e PortXmitPkts.Value -e PortRcvData.Value  
-e PortRcvPkts.Value -e SymbolErrors.Value -e LinkErrorRecovery.Value  
-e LinkDowned.Value -e PortRcvErrors.Value  
-e PortRcvRemotePhysicalErrors.Value -e PortRcvSwitchRelayErrors.Value  
-e PortXmitConstraintErrors.Value -e PortRcvConstraintErrors.Value  
-e LocalLinkIntegrityErrors.Value -e ExcessiveBufferOverrunErrors.Value
```

Examples

```
opaextractstat2 topology_file  
opaextractstat2 topology_file -c my_opamon.conf
```

5.3.7 opafindgood

Checks for hosts that are able to be pinged, accessed via SSH, and active on the Intel® Omni-Path Fabric. Produces a list of good hosts meeting all criteria. Typically used to identify good hosts to undergo further testing and benchmarking during initial cluster staging and startup.

The resulting `good` file lists each good host exactly once and can be used as input to create `mpi_hosts` files for running `mpi_apps` and the HFI-SW cable test. The files `alive`, `running`, `active`, `good`, and `bad` are created in the selected directory listing hosts passing each criteria.

This command assumes the Node Description for each host is based on the `hostname -s` output in conjunction with an optional `hfil_#` suffix. When using a `/etc/opa/hosts` file that lists the hostnames, this assumption may not be correct.

This command automatically generates the file `FF_RESULT_DIR/punchlist.csv`. This file provides a concise summary of the bad hosts found. This can be imported into Excel directly as a `*.csv` file. Alternatively, it can be cut/pasted into Excel, and the **Data/Text to Columns** toolbar can be used to separate the information into multiple columns at the semicolons.



A sample generated output is:

```
# opafindgood
3 hosts will be checked
2 hosts are pingable (alive)
2 hosts are ssh'able (running)
2 total hosts have FIs active on one or more fabrics (active)
No Quarantine Node Records Returned
1 hosts are alive, running, active (good)
2 hosts are bad (bad)
Bad hosts have been added to /root/punchlist.csv
# cat /root/punchlist.csv
2015/10/04 11:33:22;phs1fnivd13u07n1 hfi1_0 p1 phs1swivd13u06 p16;Link errors
2015/10/07 10:21:05;phs1swivd13u06;Switch not found in SA DB
2015/10/09 14:36:48;phs1fnivd13u07n4;Doesn't ping
2015/10/09 14:36:48;phs1fnivd13u07n3;No active port
```

For a given run, a line is generated for each failing host. Hosts are reported exactly once for a given run. Therefore, a host that does not ping is NOT listed as `can't ssh` nor `No active port`. There may be cases where ports could be active for hosts that do not ping, especially if Ethernet host names are used for the ping test. However, the lack of ping often implies there are other fundamental issues, such as PXE boot or inability to access DNS or DHCP to get proper host name and IP address. Therefore, reporting hosts that do not ping is typically of limited value.

Note that `opafindgood` queries the SA for `NodeDescriptions` to determine hosts with active ports. As such, ports may be active for hosts that cannot be accessed via SSH or pinged.

By default, `opafindgood` checks for and reports nodes that are quarantined for security reasons. To skip this, use the `-Q` option.

Syntax

```
opafindgood [-R|-A|-Q] [-d dir] [-f hostfile] [-h 'hosts']
[-t portsfile] [-p ports] [-T timelimit]
```

Options

<code>--help</code>	Produces full help text.
<code>-R</code>	Skips the running test (SSH). Recommended if password-less SSH is not set up.
<code>-A</code>	Skips the active test. Recommended if Intel® Omni-Path Fabric software or fabric is not up.
<code>-Q</code>	Skips the quarantine test. Recommended if Intel® Omni-Path Fabric software or fabric is not up.
<code>-d dir</code>	Specifies the directory in which to create alive, active, running, good, and bad files. Default is <code>/etc/opa</code> directory.
<code>-f hostfile</code>	Specifies the file with hosts in cluster. Default is <code>/etc/opa/hosts</code> directory.



- `-h hosts` Specifies the list of hosts to ping.
- `-t portsfile` Specifies the file with list of local HFI ports used to access fabric(s) for analysis. Default is `/etc/opa/ports` file.
- `-p ports` Specifies the list of local HFI ports used to access fabric(s) for analysis.
- Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format `hfi:port`, for example:
- `0:0` First active port in system.
- `0:y` Port `y` within system.
- `x:0` First active port on HFI `x`.
- `x:y` HFI `x`, port `y`.
- `-T timelimit` Specifies the time limit in seconds for host to respond to SSH. Default = 20 seconds.

Environment Variables

The following environment variables are also used by this command:

- `HOSTS` List of hosts, used if `-h` option not supplied.
- `HOSTS_FILE` File containing list of hosts, used in absence of `-f` and `-h`.
- `PORTS` List of ports, used in absence of `-t` and `-p`.
- `PORTS_FILE` File containing list of ports, used in absence of `-t` and `-p`.
- `FF_MAX_PARALLEL` Maximum concurrent operations.

Examples

```
opafindgood
opafindgood -f allhosts
opafindgood -h 'arwen elrond'
HOSTS='arwen elrond' opafindgood
HOSTS_FILE=allhosts opafindgood
opafindgood -p '1:1 1:2 2:1 2:2'
```

5.3.8 opalinkanalysis

(Switch) Encapsulates the capabilities for link analysis. Additionally, this tool includes cable and fabric topology verification capabilities. This tool is built on top of `opareport` (and its analysis capabilities), and accepts the same syntax for input topology and snapshot files.



In addition to being able to run assorted `opareport` link analysis reports, and generate human-readable output, this tool additionally analyzes the results and appends a concise summary of issues found to the `FF_RESULT_DIR/punchlist.csv` file.

Syntax

```
opalinkanalysis [-U] [-t portsfile] [-p ports] [-T topology_input]
[-X snapshot_input] [-x snapshot_suffix] [-c file] reports ...
```

Options

<code>--help</code>	Produces full help text.
<code>-U</code>	Omits unexpected devices and links in <code>punchlist</code> file from verify reports.
<code>-t <i>portsfile</i></code>	Specifies the file with list of local HFI ports used to access fabric(s) for analysis, default is <code>/etc/opa/ports</code> .
<code>-p <i>ports</i></code>	Specifies the list of local HFI ports used to access fabrics for analysis. Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format <code>hfi:port</code> , for example: <code>0:0</code> First active port in system. <code>0:y</code> Port <i>y</i> within system. <code>x:0</code> First active port on HFI <i>x</i> . <code>x:y</code> HFI <i>x</i> , port <i>y</i> .
<code>-T <i>topology_input</i></code>	Specifies the name of a topology input file to use. Any <code>%P</code> markers in this filename are replaced with the <code>hfi:port</code> being operated on (such as <code>0:0</code> or <code>1:2</code>). Default is <code>/etc/opa/topology.%P.xml</code> . If <code>NONE</code> is specified, does not use any <code>topology_input</code> files. See opareport on page 171 for more information on <code>topology_input</code> files.
<code>-X <i>snapshot_input</i></code>	Performs analysis using data in <code>snapshot_input</code> . <code>snapshot_input</code> must have been generated via a previous <code>opareport -o snapshot</code> run. If an errors report is specified, snapshot must have been generated with the <code>opareport -s</code> option. When this option is used, only one port may be specified to select a <code>topology_input</code> file (unless <code>-T</code> specified). When this option is used, <code>clearerrors</code> and <code>clearhwerrors</code> reports are not permitted.



<code>-x</code> <code>snapshot_suffix</code>	Creates a snapshot file per selected port. The files are created in <code>FF_RESULT_DIR</code> with names of the form: <code>snapshotSUFFIX.HFI:PORT.xml</code> .	
<code>-c file</code>	Specifies the error thresholds configuration file. The default is <code>/etc/opa/opamon.si.conf</code> .	
<code>reports</code>	Supports the following reports:	
	<code>errors</code>	Specifies link error analysis.
	<code>slowlinks</code>	Specifies links running slower than expected.
	<code>misconfiglinks</code>	Specifies links configured to run slower than supported.
	<code>misconnlks</code>	Specifies links connected with mismatched speed potential.
	<code>all</code>	Includes all reports above. (<code>errors</code> , <code>slowlinks</code> , <code>misconfiglinks</code> , and <code>misconnlks</code>)
	<code>verifylinks</code>	Verifies links against topology input.
	<code>verifyextlinks</code>	Verifies links against topology input. Limits analysis to links external to systems.
	<code>verifyfilinks</code>	Verifies links against topology input. Limits analysis to FI links.
	<code>verifyislinks</code>	Verifies links against topology input. Limits analysis to inter-switch links.
	<code>verifyextislinks</code>	Verifies links against topology input. Limits analysis to inter-switch links external to systems.
	<code>verifyfis</code>	Verifies FIs against topology input.
	<code>verifysws</code>	Verifies switches against topology input.
	<code>verifyrtrs</code>	Verifies routers against topology input.
	<code>verifynodes</code>	Verifies FIs, switches, and routers against topology input.
	<code>verifysms</code>	Verifies SMs against topology input.



<code>verifyall</code>	Verifies links, FIs, switches, routers, and SMs against topology input.
<code>clearerrors</code>	Clears error counters, uses PM if available.
<code>clearhwerrors</code>	Clears hardware error counters, bypasses PM.
<code>clear</code>	Includes <code>clearerrors</code> and <code>clearhwerrors</code> .

A punchlist of bad links is also appended to the file: `FF_RESULT_DIR/punchlist.csv`

Examples

```
opalinkanalysis errors
opalinkanalysis errors clearerrors
opalinkanalysis -p '1:1 1:2 2:1 2:2'
```

Environment Variables

The following environment variables are also used by this command:

<code>PORTS</code>	List of ports, used in absence of <code>-t</code> and <code>-p</code> .
<code>PORTS_FILE</code>	File containing list of ports, used in absence of <code>-t</code> and <code>-p</code> .
<code>FF_TOPOLOGY_FILE</code>	File containing <i>topology_input</i> , used in absence of <code>-T</code> .

5.3.9 opareport

(All) Provides powerful fabric analysis and reporting capabilities. Must be run on a host connected to the Intel® Omni-Path Fabric with the Intel® Omni-Path Fabric Suite FastFabric Toolset installed.

Syntax

```
opareport [-v][-q] [-h hfi] [-p port]
[-o report] [-d detail] [-P|-H] [-N] [-x]
[-X snapshot_input] [-T topology_input] [-s] [-r] [-V]
[-i seconds] [-b date_time] [-e date_time] [-C]
[-a] [-m] [-K mkey] [-M] [-A] [-c file] [-L]
[-F point] [-S point] [-D point] [-Q]
```

Options

<code>--help</code>	Produces full help text.
<code>-v/--verbose</code>	Returns verbose output.



<code>-q/--quiet</code>	Disables progress reports.
<code>-h/--hfi <i>hfi</i></code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p <i>port</i></code> port is a system-wide port number. (Default is 0.)
<code>-p/--port <i>port</i></code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-o/--output <i>report</i></code>	Specifies the report type for output. Refer to Report Types on page 174 for details.
<code>-d/--detail <i>level</i></code>	Specifies the level of detail 0-n for output. Default is 2.
<code>-P/--persist</code>	Only includes data persistent across reboots.
<code>-H/--hard</code>	Only includes permanent hardware data.
<code>-N/--noname</code>	Omits node and IOC names.
<code>-x/--xml</code>	Produces output in XML.
<code>-X/--infile <i>snapshot_input</i></code>	Generates a report using the data in the <code>snapshot_input</code> file. <code>snapshot_input</code> must have been generated during a previous <code>-o <i>snapshot</i></code> run. When used, the <code>-s</code> , <code>-i</code> , <code>-C</code> , and <code>-a</code> options are ignored. '-' may be used as the <code>snapshot_input</code> to specify stdin.
<code>-T/--topology <i>topology_input</i></code>	Uses <code>topology_input</code> file to augment and verify fabric information. When used, various reports can be augmented with information not available electronically (such as cable labels). '-' may be used to specify stdin.
<code>-s/--stats</code>	Gets performance statistics for all ports.
<code>-i/--interval <i>seconds</i></code>	Obtains performance statistics over interval <i>seconds</i> . Clears all statistics, waits interval <i>seconds</i> , then generates report. Implies <code>-s</code> option.
<code>-b/--begin <i>date_time</i></code>	Obtains past performance stats over an interval beginning at <i>date_time</i> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago."



<code>-e/--end <i>date_time</i></code>	Obtains past performance stats over an interval ending at <i>date_time</i> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago".
<code>-C/--clear</code>	Clears performance statistics for all ports. Only statistics with error thresholds are cleared. A clear occurs after generating the report.
<code>-a/--clearall</code>	Clears all performance statistics for all ports.
<code>-m/--smadirect</code>	Accesses fabric information directly from SMA.
<code>-K/--mkey <i>mkey</i></code>	Specifies the SMA M_Key for direct SMA query. Default is 0.
<code>-M/--pmadirect</code>	Accesses performance statistics using direct PMA.
<code>-A/--allports</code>	Gets PortInfo for down switch ports. Uses direct SMA to get this data. If used with <code>-M</code> , also gets PMA stats for down switch ports.
<code>-c/--config <i>file</i></code>	Specifies the error thresholds configuration file. Default is <code>/etc/opa/opamon.conf</code> file.
<code>-L/--limit</code>	Limits operation to exact specified focus with <code>-F</code> for port error counters check (<code>-o errors</code>) and port counters clear (<code>-C</code> or <code>-i</code>). Normally, the neighbor of each selected port is also checked/cleared. Does not affect other reports.
<code>-F/--focus <i>point</i></code>	Specifies the focus area for report. Limits output to reflect a subsection of the fabric. May not work with all reports. (For example, route, mcgroups, and the verify* reports may ignore the option or not generate useful results.)
<code>-S/--src <i>point</i></code>	Specifies the source for trace route. Default is local port.
<code>-D/--dest <i>point</i></code>	Specifies the destination for trace route.
<code>-Q/--quietfocus</code>	Excludes focus description from report.

-h and -p options permit a variety of selections:

<code>-h 0</code>	First active port in system (default).
-------------------	--



- h 0 -p 0 First active port in system.
- h x First active port on HFI x.
- h x -p 0 First active port on HFI x.
- h 0 -p y Port y within system (no matter which ports are active).
- h x -p y HFI x, port y.

Snapshot-Specific Options

- r/--routes Gets routing tables for all switches.
- V/--vltables Gets the P-Key tables for all nodes and the QoS VL-related tables for all ports.

Report Types

comps	Summary of all systems and SMs in fabric.
brcomps	Brief summary of all systems and SMs in fabric.
nodes	Summary of all node types and SMs in fabric.
brnodes	Brief summary of all node types and SMs in fabric.
ious	Summary of all IO units in the fabric.
lids	Summary of all LIDs in the fabric.
links	Summary of all links.
extlinks	Summary of links external to systems.
filinks	Summary of links to FIs.
islinks	Summary of inter-switch links.
extislinks	Summary of inter-switch links external to systems.
slowlinks	Summary of links running slower than expected.
slowconfiglinks	Summary of links configured to run slower than supported, includes <code>slowlinks</code> .
slowconnlinks	Summary of links connected with mismatched speed potential, includes <code>slowconfiglinks</code> .



<code>misconfiglinks</code>	Summary of links configured to run slower than supported.
<code>misconnlinks</code>	Summary of links connected with mismatched speed potential.
<code>errors</code>	Summary of links whose errors exceed counts in the configuration file.
<code>otherports</code>	Summary of ports not connected to the fabric.
<code>linear</code>	Summary of linear forwarding data base (FDB) for each switch.
<code>mcast</code>	Summary of multicast FDB for each switch in the fabric.
<code>mcgroups</code>	<p>Summary of multicast groups.</p> <p>When used in conjunction with <code>-d</code>, the following report details are possible:</p> <ul style="list-style-type: none"> • <code>-d0</code>: Shows the number of multicast groups • <code>-d1</code>: Shows a list of multicast groups • <code>-d2</code>: Shows a list of members per multicast group <p>This report can be used with option <code>-X</code>.</p>
<code>portusage</code>	Summary of ports referenced in linear FDB for each switch, broken down by NodeType of DLID.
<code>pathusage</code>	Summary of number of FI to FI paths routed through each switch port.
<code>treepathusage</code>	Analysis of number of FI to FI paths routed through each switch port for a FAT tree.
<code>portgroups</code>	Summary of adaptive routing port groups for each switch.
<code>quarantinednodes</code>	Summary of quarantined nodes.
<code>validateroutes</code>	Validates all routes in the fabric.
<code>validatevlroutes</code>	Validates all routes in the fabric using SLSC, SCSC, and SCVL tables.
<code>validatepgs</code>	Validates all port groups in the fabric.
<code>validatecreditloops</code>	Validates topology configuration of the fabric to identify any existing credit loops.



<code>validatevlcreditloops</code>	Validates topology configuration of the fabric including SLSC, SCSC, and SCVL tables to identify any existing credit loops.
<code>validatemcroutes</code>	Validates multicast routes of the fabric.
<code>vfinfo</code>	Summary of virtual fabric (vFabric) information.
<code>vfmember</code>	Summary of vFabric membership information.
<code>verifyfis</code>	Compares fabric (or snapshot) FIs to supplied topology and identifies differences and omissions.
<code>verifysws</code>	Compares fabric (or snapshot) switches to supplied topology and identifies differences and omissions.
<code>verifynodes</code>	Returns <code>verifyfis</code> and <code>verifysws</code> reports.
<code>verifysms</code>	Compares fabric (or snapshot) SMs to supplied topology and identifies differences and omissions.
<code>verifylinks</code>	Compares fabric (or snapshot) links to supplied topology and identifies differences and omissions.
<code>verifyextlinks</code>	Compares fabric (or snapshot) links to supplied topology and identifies differences and omissions. Limits analysis to links external to systems.
<code>verifyfilinks</code>	Compares fabric (or snapshot) links to supplied topology and identify differences and omissions. Limits analysis to links to FIs.
<code>verifyislinks</code>	Compares fabric (or snapshot) links to supplied topology and identify differences and omissions. Limits analysis to inter-switch links.
<code>verifyextislinks</code>	Compares fabric (or snapshot) links to supplied topology and identify differences and omissions. Limits analysis to inter-switch links external to systems.
<code>verifyall</code>	Returns <code>verifyfis</code> , <code>verifysws</code> , <code>verifysms</code> , and <code>verifylinks</code> reports.
<code>all</code>	Returns <code>comps</code> , <code>nodes</code> , <code>iOUS</code> , <code>links</code> , <code>extlinks</code> , <code>slowconnlinks</code> , and <code>errors</code> reports.
<code>route</code>	Traces route between <code>-S</code> and <code>-D</code> points.
<code>bfrctrl</code>	Reports Buffer Control Tables for all ports.



snapshot	Outputs snapshot of the fabric state for later use as <i>snapshot_input</i> . This implies <i>-x</i> . May not be combined with other reports. When selected, <i>-F</i> , <i>-P</i> , <i>-H</i> , and <i>-N</i> options are ignored.
topology	Outputs the topology of the fabric for later use as <i>topology_input</i> . This implies <i>-x</i> . May not be combined with other reports.
none	No report, useful to clear statistics.

Point Syntax

gid:value	<i>value</i> is numeric port GUID of form: subnet:guid.
lid:value	<i>value</i> is numeric LID.
lid:value:node	<i>value</i> is numeric LID, selects entire node with given LID.
lid:value:port:value2	<i>value</i> is numeric LID of node, <i>value2</i> is port number.
portguid:value	<i>value</i> is numeric port GUID.
nodeguid:value	<i>value</i> is numeric node GUID.
nodeguid:value1:port:value2	<i>value1</i> is numeric node GUID, <i>value2</i> is port number.
iocguid:value	<i>value</i> is numeric IOC GUID.
iocguid:value1:port:value2	<i>value1</i> is numeric IOC GUID, <i>value2</i> is port number.
systemguid:value	<i>value</i> is numeric system image GUID.
systemguid:value1:port:value2	<i>value1</i> is the numeric system image GUID, <i>value2</i> is port number.
ioc:value	<i>value</i> is IOC Profile ID String (IOC Name).
ioc:value1:port:value2	<i>value1</i> is IOC Profile ID String (IOC Name), <i>value2</i> is port number.
iocpat:value	<i>value</i> is glob pattern for IOC Profile ID String (IOC Name).



<code>iocpat:value1:port:value2</code>	<code>value1</code> is glob pattern for IOC Profile ID String (IOC Name), <code>value2</code> is port number.
<code>ioctype:value</code>	<code>value</code> is IOC type (SRP or OTHER).
<code>ioctype:value1:port:value2</code>	<code>value1</code> is IOC type (SRP or OTHER); <code>value2</code> is port number.
<code>node:value</code>	<code>value</code> is node description (node name).
<code>node:value1:port:value2</code>	<code>value1</code> is node description (node name), <code>value2</code> is port number.
<code>nodepat:value</code>	<code>value</code> is glob pattern for node description (node name).
<code>nodepat:value1:port:value2</code>	<code>value1</code> is the glob pattern for the node description (node name), <code>value2</code> is port number.
<code>nodedetpat:value</code>	<code>value</code> is glob pattern for node details.
<code>nodedetpat:value1:port:value2</code>	<code>value1</code> is the glob pattern for the node details, <code>value2</code> is port number.
<code>nodetype:value</code>	<code>value</code> is node type (SW, FI, or RT).
<code>nodetype:value1:port:value2</code>	<code>value1</code> is node type (SW, FI, or RT), <code>value2</code> is port number.
<code>rate:value</code>	<code>value</code> is string for rate (25g, 50g, 75g, 100g), omits switch mgmt port 0.
<code>portstate:value</code>	<code>value</code> is a string for state (down, init, armed, active, notactive, initarmed).
<code>portphysstate:value</code>	<code>value</code> is a string for PHYs state (polling, disabled, training, linkup, recovery, offline, test)
<code>mtucap:value</code>	<code>value</code> is MTU size (2048, 4096, 8192, 10240), omits switch mgmt port 0.
<code>labelpat:value</code>	<code>value</code> is glob pattern for cable label.
<code>lengthpat:value</code>	<code>value</code> is glob pattern for cable length.
<code>cabledetpat:value</code>	<code>value</code> is glob pattern for cable details.
<code>cabinflenpat:value</code>	<code>value</code> is glob pattern for cable info length.



<code>cabinfvendnamepat:value</code>	<i>value</i> is glob pattern for cable info vendor name.
<code>cabinfvendpnpat:value</code>	<i>value</i> is glob pattern for cable info vendor part number.
<code>cabinfvendrevpat:value</code>	<i>value</i> is glob pattern for cable info vendor revision.
<code>cabinfvendsnpat:value</code>	<i>value</i> is glob pattern for cable info vendor serial number.
<code>cabinftype:value</code>	<i>value</i> is either <code>optical</code> , <code>passive_copper</code> , <code>active_copper</code> , or <code>unknown</code> .
<code>linkdetpat:value</code>	<i>value</i> is glob pattern for link details.
<code>portdetpat:value</code>	<i>value</i> is glob pattern for port details.
<code>sm</code>	Specifies the master subnet manager (SM).
<code>smdetpat:value</code>	<i>value</i> is glob pattern for SM details.
<code>route:point1:point2</code>	Specifies all ports along the routes between the two given points.
<code>led:value</code>	<i>value</i> is either <code>on</code> or <code>off</code> for LED port beacon.
<code>linkqual:value</code>	Specifies the ports with a link quality equal to <i>value</i> .
<code>linkqualLE:value</code>	Specifies the ports with a link quality less than or equal to <i>value</i> .
<code>linkqualGE:value</code>	Specifies the ports with a link quality greater than or equal to <i>value</i> .

Examples

`opareport` can generate hundreds of different reports. Commonly generated reports include the following:

```
opareport -o comps -d 3
opareport -o errors -o slowlinks
opareport -o nodes -F portguid:0x00117500a000447b
opareport -o nodes -F nodeguid:0x001175009800447b:port:1
opareport -o nodes -F nodeguid:0x001175009800447b
opareport -o nodes -F 'node:duster hfil_0'
opareport -o nodes -F 'node:duster hfil_0:port:1'
opareport -o nodes -F 'nodepat:d*'
opareport -o nodes -F 'nodepat:d*:port:1'
opareport -o nodes -F 'nodedetpat:compute*'
opareport -o nodes -F 'nodedetpat:compute*:port:1'
```



```
opareport -o nodes -F nodetype:FI
opareport -o nodes -F nodetype:FI:port:1
opareport -o nodes -F lid:1
opareport -o nodes -F led:on
opareport -o nodes -F led:off
opareport -o nodes -F lid:1:node
opareport -o nodes -F lid:1:port:2
opareport -o nodes -F gid:0xfe80000000000000:0x00117500a000447b
opareport -o nodes -F systemguid:0x001175009800447b
opareport -o nodes -F systemguid:0x001175009800447b:port:1
opareport -o nodes -F iocguid:0x00117501300001e0
opareport -o nodes -F iocguid:0x00117501300001e0:port:2
opareport -o nodes -F 'ioc:Chassis 0x001175005000010C, Slot 2, IOC 1'
opareport -o nodes -F 'ioc:Chassis 0x001175005000010C, Slot 2, IOC 1:port:2'
opareport -o nodes -F 'iocpat:*Slot 2*'
opareport -o nodes -F 'iocpat:*Slot 2*:port:2'
opareport -o nodes -F ioctype:SRP
opareport -o nodes -F ioctype:SRP:port:2
opareport -o extlinks -F rate:100g
opareport -o extlinks -F portstate:armed
opareport -o extlinks -F portphysstate:linkup
opareport -o extlinks -F 'labelpat:S1345*'
opareport -o extlinks -F 'lengthpat:11m'
opareport -o extlinks -F 'cabledetpat:*hitachi*'
opareport -o extlinks -F 'linkdetpat:*core ISL*'
opareport -o extlinks -F 'portdetpat:*mgmt*'
opareport -o links -F mtucap:2048
opareport -o nodes -F sm
opareport -o nodes -F 'smdetpat:primary*'
opareport -o nodes -F 'route:node:duster hfil_0:node:cuda hfil_0'
opareport -o nodes -F 'route:node:duster hfil_0:port:1:node:cuda hfil_0:port:2'
opareport -s -o snapshot > file
opareport -o topology > topology.xml
opareport -o errors -X file
opareport -s --begin "2 days ago"
opareport -s --begin "12:30" --end "14:00"
```

Other Information

`opareport` also supports operation with the Fabric Manager Performance Manager (PM)/Performance Manager Agent (PMA). When `opareport` detects the presence of a PM, it automatically issues any required PortCounter queries and clears to the PM to access the PMs running totals. If a PM is not detected, then `opareport` directly accesses the PMAs on all the nodes. The `-M` option can force access to the PMA even if a PM is present.

`opareport` takes advantage of these interfaces to obtain extensive information about the fabric from the subnet manager and the end nodes. Using this information, `opareport` is able to cross-reference it and produce analysis greatly beyond what any single subnet manager request could provide. As such, it exceeds the capabilities previously available in tools such as `opasaquery` and `opafabricinfo`.

`opareport` obtains and displays counters from the Fabric Manager PM/PA or directly from the fabric PMAs using the `-M` option.

`opareport` internally cross-references all this information so its output can be in user-friendly form. Reports include GUIDs, LIDs, and names for components. Obviously, these reports are easiest to read if the end user has taken the time to provide unique names for all the components in the fabric (node names and IOC names). All Intel components support this capability. For hosts, the node names are



automatically assigned based on the network host name of the server. For switches and line cards, the names can be assigned using the element managers for each component.

Each run of `opareport` obtains up-to-date information from the fabric. At the start of the run `opareport` takes a few seconds to obtain all the fabric data, then it is output to `stdout`. The reports are sorted by GUIDs and other permanent information so they can be rerun in the future and produce output in the same order even if components have been rebooted. This is useful for comparison using simple tools like `diff`. `opareport` permits multiple reports to be requested for a single run (for example, one of each report type).

By default, `opareport` uses the first active port on the local system. However, if the Management Node is connected to more than one fabric (for example, a subnet), the Intel® Omni-Path Host Fabric Interface (HFI) and port may be specified to select the fabric to analyze.

For additional information, refer to [opareport Detailed Information](#) on page 183.

5.3.10 opareports

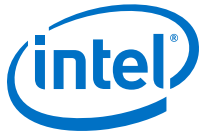
(All) `opareports` is a front end to `opareport` that provides many of the same options and capabilities. It can also run a report against multiple fabrics or subnets (for example, local host HFI ports). `opareports` can use an input file to augment the reports using additional details from the `topology_input` file.

Syntax

```
opareports [-t portsfile] [-p ports] [-T topology_input]
[opareport arguments]
```

Options

<code>--help</code>	Produces full help text.
<code>-t portsfile</code>	Specifies the file with list of local HFI ports used to access fabric for analysis. Default is <code>/etc/opa/ports</code> file.
<code>-p ports</code>	Specifies the list of local HFI ports used to access fabric for counter clear. Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format <code>hfi:port</code> , for example: 0:0 First active port in system. 0:y Port y within system. x:0 First active port on HFI x. x:y HFI x, port y.



`-T topology_input` Specifies the name of a topology input file to use. The filename may have %P as a marker which is replaced with the hfi:port being operated on, such as 0:0 or 1:2. The default filename is specified by FF_TOPOLOGY_FILE as /etc/opa/topology.%P.xml. If -T NONE is specified, no topology input file is used.

`opareport arguments` Options are passed to opareport. See [opareport](#) on page 171 for the full set of options.

Notes: When using opareport arguments, regard the following:

- The -h and -X options are not available.
- The meaning of -p is different for opareports than opareport.
- When run against multiple fabrics, the -x and -o snapshot options are not available.
- When run against multiple fabrics, the -F option is applied to all fabrics.

Examples

```
opareports
opareports -p '1:1 2:1'
```

Environment Variables

The following environment variables are also used by this command:

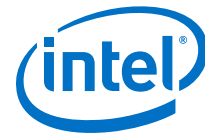
PORTS	List of ports, used in absence of -t and -p.
PORTS_FILE	File containing list of ports, used in absence of -t and -p.
FF_TOPOLOGY_FILE	File containing topology_input (may have %P marker in filename), used in absence of -T.

Details

For simple fabrics, the Intel® Omni-Path Fabric Suite FastFabric Toolset host is connected to a single fabric. By default, the first active port on the FastFabric Toolset host is used to analyze the fabric.

However, in more complex fabrics, the FastFabric Toolset host may be connected to more than one fabric or subnet. In this case, you can specify the ports or HFIs to use with one of the following methods:

- On the command line using the -p option.
- In a file specified using the -t option.
- Through the environment variables PORTS or PORTS_FILE.



- Using the `ports_file` configuration option in `/etc/opa/opafastfabric.conf`.

If the specified port does not exist or is empty, the first active port on the local system is used. In more complex configurations, you must specify the exact ports to use for all fabrics to be analyzed. For more information, refer to [Selection of Devices](#) on page 41.

You can specify the `topology_input` file to be used with one of the following methods:

- On the command line using the `-T` option.
- In a file specified through the environment variable `FF_TOPOLOGY_FILE`.
- Using the `ff_topology_file` configuration option in `opafastfabric.conf`.

If the specified file does not exist, no `topology_input` file is used. Alternately the filename can be specified as `NONE` to prevent use of an input file.

For additional information, refer to [opareport Detailed Information](#) on page 183.

5.3.11 opareport Detailed Information

This section provides additional information about using `opareport`.

5.3.11.1 opareport Basics

`opareport` can be run with no options at all. In this mode it provides a brief list of the nodes in the fabric, the `brnodes` report.

A sample of an `opareport` for a small fabric follows:

```
# opareport
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Node Type Brief Summary

4 Connected FIs in Fabric:
NodeGUID      Type Name
  Port LID    PortGUID      Width Speed
0x00117501016a35f0 FI coyote hfil_0
    1 0x0004 0x00117501016a35f0    4    25Gb
0x00117501016a361d FI goblin hfil_0
    1 0x0003 0x00117501016a361d    4    25Gb
0x00117501016a365f FI ogre hfil_0
    1 0x0005 0x00117501016a365f    4    25Gb
0x00117501016a366d FI duster hfil_0
    1 0x0001 0x00117501016a366d    4    25Gb

1 Connected Switches in Fabric:
NodeGUID      Type Name
  Port LID    PortGUID      Width Speed
0x00117500ff6a5619 SW edge1
    0 0x0002 0x00117500ff6a5619    1    25Gb
    12                4    25Gb
    31                4    25Gb
    35                4    25Gb
    39                4    25Gb
```



```
1 Connected SMs in Fabric:
State      GUID      Name
Master     0x00117501016a366d duster hfil_0
```

Each `opareport` allows for various levels of detail. Increasing detail is shown as further indentation of the additional information. The `-d` option to `opareport` controls the detail level. The default is 2. Values from 0–n are permitted. The maximum detail per report varies, but most have less than five detail levels.

For example, when the previous report is run at detail level 0, the output is as follows:

```
# opareport -d 0
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Node Type Brief Summary

4 Connected FIs in Fabric
1 Connected Switches in Fabric
1 Connected SMs in Fabric
```

A summary of fabric components is shown in the following example. This report is very similar to `opafabricinfo`. At the next level of detail, the report has more detail:

```
# opareport -d 1
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Node Type Brief Summary

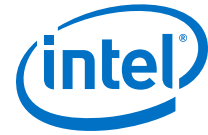
4 Connected FIs in Fabric:
NodeGUID      Type Name
0x00117501016a35f0 FI ogre hfil_0
0x00117501016a361d FI goblin hfil_0
0x00117501016a365f FI coyote hfil_0
0x00117501016a366d FI duster hfil_0

1 Connected Switches in Fabric:
NodeGUID      Type Name
0x00117500ff6a5619 SW edge1

1 Connected SMs in Fabric:
State      GUID      Name
Master     0x00117501016a366d duster hfil_0
```

The previous examples were all performed with a single report: the `brnodes` (Brief Nodes) report. This is just one of the many topology reports that `opareport` can generate.

Other reports summarize the present state of the fabric. Use these reports to analyze the configuration of the fabric and verify that the installation is consistent with the desired design and configuration. These reports include:



<code>nodes</code>	A more verbose form of <code>brnode</code> that provides much greater levels of detail to drill down into all the details of every node, even down to all the port state, IOUs/IOCs/Services, and Port counters.
<code>comps and brcomps</code>	<p>Very similar to <code>brnodes</code> and <code>nodes</code>, except the reports are organized around systems. The grouping into systems is based on system image GUIDs for each node. This report presents more complex systems (such as servers with multiple HFIs or large switches composed of multiple switch chips).</p> <p><i>Note:</i> All Intel switches implement a system image GUID and are therefore properly grouped. However, some third-party devices do not implement the system image GUID and may report a value of 0. In such a case, <code>opareport</code> treats each component as an independent system.</p>
<code>links</code>	Presents all the links in the fabric. The output is very concise and helps to identify the connectivity between nodes in the fabric. This includes both internal (inside a large switch or system) and external ports (cables).
<code>extlinks</code>	Lists all the external links in the fabric, for example, those between different systems. This report omits links internal to a single system. Identification of a system is through <code>SystemImageGuid</code> .
<code>lids</code>	Similar to <code>brnodes</code> , however it is organized and sorted by LID. The output is very concise and provides a simple cross reference of LIDs assigned to each HFI and Switch in the fabric. This information can be useful in interpreting the output from the <code>linear</code> , <code>mcast</code> , and <code>portusage</code> reports.
<code>ious</code>	Similar to the <code>nodes</code> reports, however the focus is around IOUs/IOCs and IO Services in the fabric. This report identifies various IO devices in the fabric and their capabilities, such as direct-attach storage.
<code>otherports</code>	Lists all ports that are not connected to this fabric. This report identifies additional ports on FIs or Switches that are not connected to this fabric. For switches, these represent unused ports. For FIs, these may be ports connected to other fabrics or unused ports.

Additionally, `opareport` has reports that analyze the operational characteristics of the fabric and identify bottlenecks and faulty components in the fabric. These reports include:

<code>slowlinks</code>	Identifies links that are running slower than expected, that pinpoints bad cables or components in the fabric. The analysis includes both link speed and width.
<code>slowconfiglinks</code>	Extends the <code>slowlinks</code> report to also report links that have been configured (typically by software) to run at a width or speed below their potential.



<code>slowconnl</code>	Extends on the <code>slowconfiglinks</code> report to also report links that are cabled such that one of the ends of the link can never run to its potential.
<code>misconfiglinks</code>	Similar to <code>slowconfiglinks</code> in that it reports links that have been configured to run below their potential. However, report does not include links that are running slower than expected.
<code>misconnl</code>	Similar to <code>slowconnl</code> in that it reports links that have been connected between ports of different speed potential. However, report does not include links that are running slower than expected, nor links that have been configured to run slower than their potential.
<code>errors</code>	Performs a single point in time analysis of the PMA port counters for every node and port in the fabric. All the counters are compared against configured thresholds. Defaults are listed in the <code>opamon.conf</code> file. Any link whose counters exceed these thresholds are listed. Depending on the detail level, the exact counter and threshold are reported. This is a powerful way to identify marginal links in the fabric such as bad or loose cables or damaged components. The <code>opamon.si.conf</code> file can also be used to check for any non-zero values for signal integrity (SI) counters.
<code>route</code>	Identifies two end points in the fabric (by node name, node GUID, port name, port GUID, system image GUID, LID, port GUID, IOC GUID, or IOC name), and obtains a list of all the links and components used when these two end points communicate. If there are multiple paths between the end points, such as an FI with 2 connected ports or a system with 2 FIs, the route for every available path is reported based on presently configured routing tables.
<code>linear</code>	Shows the linear forwarding table for each switch in the fabric. Used to manually review the routing of unicast traffic in the fabric. For each switch, every unicast LID is shown along with the port it is routed out (egress port), and the neighboring Node and Port. For large fabrics, this report can be quite large.
<code>mcast</code>	Shows the multicast forwarding table for each switch in the fabric. Used to manually review the routing of multicast traffic in the fabric. For each switch, every multicast LID is shown along with the list of ports it is routed out. For large fabrics, this report can be quite large.
<code>portusage</code>	Provides a summary analysis of the unicast routing in the fabric, in terms of how many LIDs of each node type are routed out a given port. Used for analysis of how balanced the routes in the fabric are, especially for ISLs and core switches. For each switch, all the ports are shown along with the counts of how many unicast LIDs are routed out each port. The total is shown along with HFI-All, HFI-Base, Switch, and Router.



- HFI-All includes all LIDs that correspond to an HFI, including LIDs that are the base LID of the HFI and LIDs that map to the HFI through LMC masking.
- HFI-Base includes only LIDs that correspond to the base LID of an HFI. HFI-Base is always a subset of HFI-All.
- Switch includes all LIDs that correspond to a Switch.
- Router includes all LIDs that correspond to a Router. Only Ports with a non-zero total are shown.

<code>pathusage</code>	Computes all the FI to FI dLID paths through the fabric and reports on the usage of each ISL Port (SW to SW link). The <code>-F</code> option indicates the switches and the ports on those switches to analyze. Switch Port 0 is always omitted from the analysis. These reports can also be run against snapshots that were performed with the <code>-r</code> option.
<code>treepathusage</code>	Similar to <code>pathusage</code> with the exception that <code>treepathusage</code> is applicable only to Fat Tree topologies and provides specific analysis of uplink and downlink paths, indicating what tier each switch is in within the fabric.

5.3.11.2 Simple Topology Verification

`opareport` provides a flexible way to identify changes to the fabric or the appropriate reassembly of the fabric after a move. For example, run `opareport` after staging and testing the fabric in a remote location before final installation at a customer site.

This type of report can be saved for later comparison to a future report. Since `opareport` produces simple text reports, standard tools such as `sdiff` (side by side diff) can be used for comparison and analysis of the changes.

In this mode of operation, all previous reports are available, however, you can filter the information that is output. Use the `all` report to include all reports of general interest.

Use the `-P` option to omit information that does not persist across a fabric reboot, for example, LIDs and error counters. In the report, the information is marked out with `xxx`.

If software configuration changes are anticipated, use the `opareport -H` option to only include hardware information. Use this option when adjusting the timeouts the SM configures in the fabric.

Use the `-N` option to omit all the node and IOC names from the report. If changes are anticipated in this area, this option can be used so future differences do not report changes in names.

5.3.11.3 Advanced Topology Verification

You can use the `-T` option for `opareport` to compare the state of the fabric against a previous state or a user-generated configuration for the fabric.



The XML description used by the `-T` option is the same as the XML format generated by the `-o links` or `-o extlinks` and/or `-o brnodes` reports when they are run with the `-x` option. The `opareport -o topology` argument is an easy way to generate such a report and is equivalent to specifying all three of these reports.

A simple way to perform topology verification against a previous configuration is to generate the previous topology using a command such as:

```
opareport -o topology -x > topology.xml
```

Later, the fabric can be compared against that topology using a command such as:

```
opareport -T topology.xml -o verifyall
```

Unlike simple `diff` comparisons discussed in [Simple Topology Verification](#), this method of topology verification performs a more context-sensitive comparison and presents information in terms of links, nodes, or SMs that are missing, unexpected, or incorrectly configured.

All the other capabilities of `opareport` are fully available when using a `topology_input` file. For example, `snapshot_input` files can also be used to generate or compare topologies based on previous fabric snapshots. In addition, the `-F` option may be used to focus the analysis.

Note: `verify*` reports may still report missing links, nodes, or SMs outside the scope of the desired focus.

There are multiple variations of advanced topology verification: `verifycas`, `verifysws`, `verifyrtrs`, `verifysms`, `verifylinks`, and `verifyextlinks`. In addition, `verifynodes` and `verifyall` can be used to generate combined reports.

`verifylinks` and `verifyextlinks` perform the same analysis, however, they differ in the scope of the analysis. `verifylinks` checks all links in the fabric. In contrast, `verifyextlinks` performs the following:

- Limits its verification to links outside of a system.
- Does not analyze links between nodes with the same `SystemImageGuid`, such as within a large Intel® Omni-Path Fabric Chassis.
- Ignores links from the `topology_input` file that specify a non-zero value for the XML tag `<Internal>` within the `<Link>` tag.

The XML format of `topology_input` file is shown in the following example. The example is purposely brief and omits many links, nodes, and SMs.

```
<?xml version="1.0" encoding="utf-8" ?>
<Report>
<LinkSummary>
<Link>
<Rate>25g</Rate>
<MTU>8192</MTU>
<Internal>0</Internal>
<LinkDetails>SampleHost1 to Switch</LinkDetails>
<Cable>
<CableLength>11m</CableLength>
<CableLabel>S4567</CableLabel>
```



```
<CableDetails>sample cable model xxx</CableDetails>
</Cable>
<Port>
<NodeGUID>0x0011750101660572</NodeGUID>
<PortGUID>0x0011750101660572</PortGUID>
<PortNum>1</PortNum>
<NodeType>FI</NodeType>
<NodeDesc>SampleHost1 HFI-1</NodeDesc>
<PortDetails>SampleHost1 primary port</PortDetails>
</Port>
<Port>
<NodeGUID>0x0011750007000df6</NodeGUID>
<PortNum>1</PortNum>
<NodeType>SW</NodeType>
<NodeDesc>SampleSwitch1 Leaf 4, Chip A</NodeDesc>
</Port>
</Link>
<Link>
<Rate>25g</Rate>
<MTU>8192</MTU>
<Internal>0</Internal>
<Port>
<NodeGUID>0x0011750101660574</NodeGUID>
<PortGUID>0x0011750101660574</PortGUID>
<PortNum>1</PortNum>
<NodeType>FI</NodeType>
<NodeDesc>SampleHost2 HFI-1</NodeDesc>
</Port>
<Port>
<NodeGUID>0x0011750007000e6d</NodeGUID>
<PortNum>4</PortNum>
<NodeType>SW</NodeType>
<NodeDesc>SampleSwitch1 Leaf 5, Chip A</NodeDesc>
</Port>
</Link>
</LinkSummary>
<Nodes>
<FIs>
<Node id="0x0011750101660576">
<NodeGUID>0x0011750101660576</NodeGUID>
<NodeDesc>SampleHost2 HFI-1</NodeDesc>
<NodeDetails>SampleHost2 only HFI</NodeDetails>
</Node>
</FIs>
<Switches>
<Node id="0x001175000600025a">
<NodeGUID>0x001175000600025a</NodeGUID>
<NodeDesc>SampleSwitch1 Spine 1, Chip A</NodeDesc>
<NodeDetails>core switch</NodeDetails>
</Node>
</Switches>
<SMs>
<SM id="0x0011750101660578:1">
<NodeGUID>0x0011750101660578</NodeGUID>
<NodeDesc>SampleHost2 HFI-1</NodeDesc>
<PortNum>1</PortNum>
<PortGUID>0x0011750101660579</PortGUID>
<NodeType>FI</NodeType>
<NodeType_Int>1</NodeType_Int>
<SMDetails>SampleHost2 SM</SMDetails>
</SM>
</SMs>
</Nodes>
</Report>
```

The XML tags have the following meanings:



<Report>	Primary top level tag. Exactly one such tag is permitted per file. Alternatively, this may be <Topology>.
<LinkSummary>	Container tag describing all the links expected in the fabric. Alternatively, <ExternalLinkSummary> may be used. <ExternalLinkSummary> should be used if the file only describes external links. If both external and internal links are described, <LinkSummary> should be used. Only one of these two choices is permitted per file.
<Link>	Container tag describing a single link. Many instances of this tag can occur per <LinkSummary> or <ExternalLinkSummary>. <Link> allows the following tags: <Rate> String describing the expected rate of the link. Valid values are 2.5g, 5g, 10g, 20g, 30g, 40g, 60g, 80g, or 120g. The value is case-insensitive but must contain no extra whitespace. Alternatively, an integer value <Rate_Int> may be provided based on the values for Rate from the SMA packets. If both <Rate> and <Rate_Int> are specified, whichever value appears later within the given link is used. If neither is specified, the rate of the link is not verified. <MTU> An integer describing the expected MTU of the link. Valid values are 256, 512, 1024, 2048, and 4096. If not specified, the MTU of the link is not verified. <Internal> A flag indicating if the link is internal or external. A value of 0 indicates external links that are processed by both verifylinks and verifyextlinks. A value of 1 indicates an internal link that is only processed by verifylinks. If omitted, the actual fabric link attributes or the attributes of the port are used to determine if the link should be processed. The value for this field is not verified against the actual fabric. <LinkDetails> A free form text field of up to 64 characters. This field is optional. When provided, this is output as a link attribute in all reports that show link details, such as links, extlinks, route, verifylinks, and verifyextlinks reports. Intel recommends you use this field to describe the purpose of the link. This field can also be used by the linkdetpat focus option to select the link.
<Cable>	A container tag providing additional information about the cable.



<Cable> allows the following tags:

<CableLength> A free form text field up to 10 characters. This field is optional. When provided, this is output as a link cable attribute in all reports that show link details, such as links, extlinks, route, verifylinks, and verifyextlinks reports. Intel recommends you use this field to describe the length of the cable using text such as 11m. This field can also be used by the `lengthpat` focus option to select the link.

<CableLabel> A free form text field up to 20 characters. This field is optional. When provided, this is output as a link cable attribute in all reports that show link details, such as links, extlinks, route, verifylinks, and verifyextlinks reports. Intel recommends you use this field to describe the identifying label attached to the cable using text such as S4576. This field can also be used by the `labelpat` focus option to select the link. Using this field to match the actual unique physical labels placed on the cables during installation can greatly help cross-referencing the reports to the physical cluster, such as when needing to identify or replace cables.

<CableDetails> A free form text field of up to 64 characters. This field is optional. When provided, this is output as a link attribute in all reports that show link details, such as links, extlinks, route, verifylinks, and verifyextlinks reports. Intel recommends you use this field to describe the type, model, and/or manufacturer of the cable. This field can also be used by the `cabledetpat` focus option to select the link.

<Port> A container tag providing additional information about the two ports that make up the link.

<Port> allows the following tags:

<NodeGUID> Node GUID reported by the SMA for the given FI, switch, or router.

<PortGUID> Port GUID reported by the SMA for the given FI, switch, or router.

Note: Switches only have PortGuids for port 0 (the internal management port), while FIs and routers have a unique GUID for every port.



<PortNum>	Port Number within the FI, switch, or router.
<NodeDesc>	Node Description reported by the FI, switch, or router. Intel recommends that you configure a unique value for this field in each node in your fabric. For example, Intel® Omni-Path Fabric Host Software Linux* hosts use the combination of Linux hostname and HFI number to create a unique NodeDesc.
<NodeType>	Node type reported by the node. Values include: FI, SW, or RT. Alternatively, an integer value <NodeType_Int> may be provided based on the values for NodeType from the SMA packets. If both <NodeType> and <NodeType_Int> are specified, whichever appears later within the given Port is used. If neither is specified, the node type of the port is not verified.
<PortDetails>	Free form text field of up to 64 characters. This field is optional. When provided, this is output as a port attribute in all reports that show port details, such as links, extlinks, route, comps, verifylinks, and verifyextlinks reports. Intel recommends you use this field to describe the purpose of the port. This field can also be used by the portdetpat focus option to select the port.

The previous fields are used to associate a port in the `topology_input` file with an actual port in the fabric, also called resolving the port. You need not provide all of the information. Association to an actual port in the fabric is performed using the following order of checks based on the tags that are specified:

- NodeGUID, PortNum
- NodeGUID, PortGUID
- NodeGUID – if given FI has exactly 1 port.
- NodeDesc, PortNum
- NodeDesc, PortGUID
- NodeDesc – if given FI has exactly 1 port.
- PortGUID, PortNum – useful to select ports other than 0 on a switch.
- PortGUID

If NodeDesc is used to specify ports, it is important that the fabric is configured such that each NodeDesc is unique. Otherwise, the <Port> may resolve to a different port than desired, which could result in incorrect results or errors during topology verification.



When redundant information is provided, the extra information is ignored while resolving the port. However, during `verifylinks` or `verifyextlinks` all the input provided is verified against the actual fabric and any discrepancies are reported.

Some examples of redundant information:

- NodeGuid, NodeDesc – NodeDesc is not used to resolve port.
- NodeGuid, PortNum, PortGuid – PortGuid is not used to resolve port.
- NodeDesc, PortNum, PortGuid – PortGuid is not used to resolve port.

The `<NodeType>` field is never used during resolution; it is only used during verification.

<code><Nodes></code>	Container tag describing all the nodes expected in the fabric.
<code><FIs></code>	Container tag describing all the FIs expected in the fabric. Many instances of this tag can occur per <code><Nodes></code> .
<code><Switches></code>	Container tag describing all the Switches expected in the fabric. Many instances of this tag can occur per <code><Nodes></code> .
<code><Routers></code>	Container tag describing all the Routers expected in the fabric. Many instances of this tag can occur per <code><Nodes></code> .
<code><SMs></code>	Container tag describing all the SMs expected in the fabric. Many instances of this tag can occur per <code><Nodes></code> .
<code><Node></code>	Container tag describing a single node (FI, SW, or RT). Many instances of this tag can occur per <code><FIs></code> , <code><Switches></code> , or <code><Routers></code> .

`<Node>` allows the following tags:

<code><NodeGUID></code>	Node GUID reported by the SMA for the given FI, Switch, or Router.
<code><NodeDesc></code>	Node Description reported by the FI, switch, or router. Intel recommends that you configure a unique value for this field in each node in your fabric. For example, Intel® Omni-Path Fabric Host Software Linux* hosts use the combination of Linux hostname and HFI number to create a unique NodeDesc.
<code><NodeDetails></code>	Free form text field of up to 64 characters. This field is optional. When provided, this is output as a node attribute in all reports that show node details, such as links, extlinks, route, comps, <code>verifycas</code> , <code>verifysws</code> , <code>verifyrts</code> , <code>verifylinks</code> , and



verifyextlinks reports. Intel recommends you use this field to describe the purpose and/or model of the node. This field can also be used by the nodedetpat focus option to select the node.

The previous fields are used to associate a Node (FI, Switch, or Router) in the topology_input file with an actual node in the fabric, also called resolving the node. You need not provide all of the information. Association to an actual node in the fabric is performed using the following order of checks based on the tags that are specified:

- NodeGUID
- NodeDesc

If NodeDesc is used to specify nodes, the fabric must be configured such that each NodeDesc is unique. Otherwise, the <Node> may resolve to a different node than desired, which could result in incorrect results or errors during topology verification.

When redundant information is provided, the extra information is ignored while resolving the node. However, during verifycas, verifysws, or verifyrtrs, all the input provided is verified against the actual fabric and any discrepancies are reported.

An example of redundant information:

- NodeGuid, NodeDesc - NodeDesc is not used to resolve node.

The node type (as implied by the container tag for the <Node>) is never used during resolution, it is only used during verification.

<SM>

Container tag describing a single SM. Many instances of this tag can occur per <SMs>.

<SM> allows the following tags:

<NodeGUID> Node GUID reported by the SMA for the given FI, switch, or router that is running the SM.

<NodeDesc> Node Description reported by the FI, switch, or router that is running the SM. Intel recommends that you configure a unique value for this field in each node in your fabric. For example, Intel® Omni-Path Fabric Host Software Linux* hosts use the combination of Linux hostname and HFI number to create a unique NodeDesc.

<PortGUID> Port GUID reported by the SMA for the given FI, switch, or router that is running the SM.



Note: Switches only have PortGuids for port 0 (the internal management port), while FIs and routers have a unique GUID for every port.

<PortNum>	Port Number within the FI, switch, or router that is running the SM.
<NodeType>	Node type reported by the node that is running the SM. Values include: FI, SW, or RT. Alternatively, an integer value <NodeType_Int> may be provided based on the values for NodeType from the SMA packets. If both <NodeType> and <NodeType_Int> are specified, whichever appears later within the given port is used. If neither is specified, the node type of the SM is not verified.
<SMDetails>	Free form text field of up to 64 characters. This field is optional. When provided, this is output as a SM attribute in all reports that show SM details, such as comps and verifysms reports. Intel recommends you use this field to describe the purpose of the SM. This field can also be used by the smdetpat focus option to select the SM.

The previous fields are used to associate a port running an SM in the `topology_input` file with an actual port in the fabric, also called resolving the SM. You need not provide all of the information. Association to an actual port in the fabric is performed using the following order of checks based on the tags that are specified:

- NodeGUID, PortNum
- NodeGUID, PortGUID
- NodeGUID – if given FI has exactly 1 active port or is a switch.
- NodeDesc, PortNum
- NodeDesc, PortGUID
- NodeDesc – if given FI has exactly 1 active port or is a switch.
- PortGUID, PortNum – limited usefulness.
- PortGUID

If NodeDesc is used to specify SM ports, the fabric must be configured such that each NodeDesc is unique. Otherwise, the <SM> may resolve to a different port than desired, which could result in incorrect results or errors during topology verification.



When redundant information is provided, the extra information is ignored while resolving the port for an SM. However, during `verifysms` all the input provided is verified against the actual fabric and any discrepancies are reported.

Some examples of redundant information:

- `NodeGuid, NodeDesc` – `NodeDesc` is not used to resolve port.
- `NodeGuid, PortNum, PortGuid` – `PortGuid` is not used to resolve port.
- `NodeDesc, PortNum, PortGuid` – `PortGuid` is not used to resolve port.

The `NodeType` field is never used during resolution, it is only used during verification.

5.3.11.4 Augmented Report Information

As discussed in [Advanced Topology Verification](#), a `topology_input` file includes additional information including cable (length, label, details), links (details), ports (details), nodes (details) and SMs (details).

A `topology_input` file can be used during any report to provide information about the fabric that is not electronically available. This can help cross-reference the output of the report against the physical fabric. For example, if the cable length field is supplied, reports can be focused on all cables of a given length. Similarly, if cable labels are supplied, the report output includes the labels, making it much easier to locate the actual cables for tasks such as rerouting or replacement.

5.3.11.5 Focused Reports

One of the more powerful features of `opareport` is the ability to focus a report on a subset of the fabric. Using the `-F` option, you can specify a node name, node name pattern, node GUID, node type, port GUID, IOC name, IOC name pattern, IOC GUID, IOC type, system image GUID, port GUID, port rate, port state, port physical state, MTU capability, LID, link quality indicator, cable info for cable length, cable info for vendor name, cable info for vendor part number, cable info for vendor rev, cable info for vendor serial number, or SM.

The subsequent report indicates the total components in the fabric but only reports on those that relate to the focus area. For example, in a nodes report, if a port is specified for focus, only the node containing that port is reported on. In a links report, if a port is specified for focus, only the link using that port is reported.

When a focus is used for fabric analysis, `-o errors`, `-C` or `-i`, the analysis includes all the ports selected by the focus as well as their neighbors. If desired, the `-L` option limits the operation to exactly the selected ports.

You may choose a focus level that is different from the orientation of the report. For example, if a node name is specified as the focus for a links report, a report of all the links to that node is provided. This includes multiple switch ports or FI ports.



You can perform reverse lookups by carefully using this feature of report focus. For example, requesting a `brnodes` report with a focus on a LID performs reverse lookup on that LID and indicates what node it is for.

When focusing a report, you can also specify a detail level. For detail 0, the report shows only a count of number of matches. For detail 1, the report shows only the highest level of the entity that matches.

5.3.11.6 Advanced Focus

As mentioned previously, you can focus a report on a subset of the fabric. In addition, you can further limit the report focus using the following methods.

The beginning of a focused report includes a summary of the items focused on. When the focus has a large scope, this list can be quite long. To omit the summary section from the report, use the `-Q` option.

- Port number specifier

The node name, node name pattern, node guid, node type, IOC name, IOC name pattern, IOC GUID, IOC type, and system image GUID also allow for a port number specifier. This limits the focus to the given port number. If the selection resolves to multiple switches or FIs, all ports on the present fabric matching the given port number are selected for the report. For example, in a system composed of multiple nodes, there may be multiple ports with the same port number.

- Route between points

This method focuses on all the ports involved in a particular route and can be an excellent way to determine a performance or error situation reported between two specific points in the fabric. For example, MPI may report `StatusTimeoutRetry` between two processes in its run.

* syadmin fields supplied in a topology file, typically generated by `opaxlattopology` or `opaxlattopology_cust` (deprecated), including cable labels, cable details, planned cable length, link details, port details, and SM details.

- glob-style patterns

You can use a wildcard focus for the node name, IOC name, node details, cable label, cable length, cable details, cable vendor name, cable vendor part number, cable vendor rev, cable vendor serial number, link details, port details, or SM details. If a consistent naming convention is used for fabric components, this method provides a powerful way to focus reports on nodes. If the host names are prefixed with an indication of their purpose, searches can be performed based on the purpose of the node.

For example, if you use a naming convention such as the following: `l###` = login node `###`, `n###` = compute node `###`, `s###` = storage node `###`, then you can create a report using one of the following patterns: `'l*'`, `'n*'`, or `'s*'`.

Note: A glob style pattern is a shell-style wildcard pattern as used by `bash` and other tools. If you use this style of pattern, you must also use single quotes so the shell does not try to expand them to match local file names.



5.3.11.7 Focus Examples

Examples of using the focus options are shown in the following list:

```
opareport -o nodes -F portguid:0x00117500a000447b
opareport -o nodes -F nodeguid:0x001175009800447b:port:1
opareport -o nodes -F nodeguid:0x001175009800447b
opareport -o nodes -F node:duster
opareport -o nodes -F node:duster:port:1
opareport -o nodes -F 'nodepat:d*'
opareport -o nodes -F 'nodepat:d*:port:1'
opareport -o nodes -F nodetype:FI
opareport -o nodes -F nodetype:FI:port:1
opareport -o nodes -F lid:1
opareport -o nodes -F lid:1:node
opareport -o nodes -F gid:0xfe80000000000000:0x00117500a000447b
opareport -o nodes -F systemguid:0x001175009800447b
opareport -o nodes -F systemguid:0x001175009800447b:port:1
opareport -o nodes -F iocguid:0x00117501300001e0
opareport -o nodes -F iocguid:0x00117501300001e0:port:2
opareport -o nodes -F 'ioc:Chassis 0x001175005000010C, Slot 2, IOC 1'
opareport -o nodes -F 'ioc:Chassis 0x001175005000010C, Slot 2, IOC 1:port:2'
opareport -o nodes -F 'iocpat:*Slot 2*'
opareport -o nodes -F 'iocpat:*Slot 2*:port:2'
opareport -o nodes -F ioctype:XXXX
opareport -o nodes -F ioctype:XXXX:port:2
opareport -o nodes -F sm
opareport -o nodes -F route:node:duster:node:cuda
opareport -o nodes -F route:node:duster:port:1:node:cuda:port:2
```

5.3.11.8 Scriptable Output

`opareport` permits custom scripting. As previously mentioned, options like `-H`, `-P`, and `-N` generate reports that can be compared to each other. The `-x` option permits output reports to be generated in XML format. The XML hierarchy is similar to the text-based reports. Using XML permits other XML tools (such as PERL XML extensions) to easily parse `opareport` output, enabling you to create scripts to further search and refine report output formats.

The `opaxmlextract` tool easily converts between XML files and delimited text files. For more information, see [opaxmlextract](#) on page 288.

You can integrate `opareport` into custom scripts. You can also generate customer-specific new report formats and cross-reference `opareport` with other site-specific information.

5.3.11.9 Monitor for Fabric Changes Using opareport

`opareport` can easily be used in other scripts. For example, the following simple script can be run as a `cron` job to identify if the fabric has changed from the initial design.

```
#!/bin/bash
# specify some filenames to use
expected_config=/usr/local/report.master # master copy of config previously
created
config=/tmp/report$$ # where we will generate new report
diffs=/tmp/report.diff$$ # where we will generate diffs

opareport -o all -d 5 -P > $config 2>/dev/null
if ! diff $config $expected_config > $diffs 2>/dev/null
```



```
then
# notify admin, for example mail the new report to the admin
cat $diffs $expected_config $config |
mail -s "fabric change detected" admin@somewhere
fi
rm -f $config $diffs
```

5.3.11.10 Sample Outputs

Analyze all ports in fabric for errors, inconsistent connections, bad cables

```
[root@duster root]# opareport -o errors -o slowlinks
Links running slower than expected Summary

Links running slower than expected:
Rate NodeGUID          Port Type Name
Active                                     Enabled
Lanes, Used(Tx), Used(Rx), Rate, Lanes, DownTo, Rates
100g 0x00117501025019ab 44 SW edge1
4 3 3 25Gb 1,2,3,4 3,4 25Gb
<-> 0x0011750102513139 44 SW edge2
4 3 3 25Gb 1,2,3,4 3,4 25Gb
100g 0x00117501025019ab 48 SW edge1
4 3 4 25Gb 1,2,3,4 3,4 25Gb
<-> 0x0011750102513139 48 SW edge2
4 4 3 25Gb 1,2,3,4 3,4 25Gb
96 of 96 Links Checked, 2 Errors found
-----
Links with errors > threshold Summary

Configured Error Thresholds:
LinkQualityIndicator      5
LinkDowned                3
RcvErrors                 100
ExcessiveBufferOverruns   3
LinkErrorRecovery         3
LocalLinkIntegrityErrors  3
XmitConstraintErrors      10
RcvConstraintErrors       10
CongDiscards              100
96 of 96 Links Checked, 0 Errors found
-----
```

Identify the route between two nodes

```
[root@goblin root]# opareport -o route -S node:"goblin hf1l_0" -D node:"orc
hf1l_0"
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Routes Summary Between:
Node: 0x001175010157409d FI goblin hf1l_0
and Node: 0x001175010157403d FI orc hf1l_0

Routes between ports:
0x001175010157409d 1 FI goblin hf1l_0
and 0x001175010157403d 1 FI orc hf1l_0
2 Paths
SGID: 0xfe80000000000000:001175010157409d
DGID: 0xfe80000000000000:001175010157403d
SLID: 0x0001 DLID: 0x0018 Reversible: Y PKey: 0x8001
Raw: N FlowLabel: 0x00000 HopLimit: 0x00 TClass: 0x00
SL: 0 Mtu: 8192 Rate: 100g PktLifeTime: 134 ms Pref: 0
```



```
Rate NodeGUID Port Type Name
100g 0x001175010157409d 1 FI goblin hfil_0
-> 0x00117501025131cb 44 SW edge1
100g 0x00117501025131cb 40 SW edge2
-> 0x001175010157403d 1 FI orc hfil_0
2 Links Traversed
SGID: 0xfe80000000000000:001175010157409d
DGID: 0xfe80000000000000:001175010157403d
SLID: 0x0001 DLID: 0x0018 Reversible: Y PKey: 0xffff
Raw: N FlowLabel: 0x00000 HopLimit: 0x00 TClass: 0x00
SL: 0 Mtu: 8192 Rate: 100g PktLifeTime: 134 ms Pref: 0
Rate NodeGUID Port Type Name
100g 0x001175010157409d 1 FI goblin hfil_0
-> 0x00117501025131cb 44 SW edge1
100g 0x00117501025131cb 40 SW edge1
-> 0x001175010157403d 1 FI orc hfil_0
2 Links Traversed
```

Obtain very detailed information about nodes

Note: To shorten the length of the output, the following example focuses on only one node.

```
[root@phwtppriv27 ~]$ opareport -o nodes -F node:"duster hfil_0" -d 5 -s
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Done Getting vFabric Records
Getting All Port Counters...
Done Getting All Port Counters
Node Type Summary
Focused on:
Node: 0x0002b3010100002b FI duster hfil_0

48 Connected FIs in Fabric:
Name: duster hfil_0
NodeGUID: 0x0002b3010100002b Type: FI
Ports: 1 PartitionCap: 16 SystemImageGuid: 0x0002b3010100002b
BaseVer: 128 SmaVer: 128 VendorID: 0x2b3 DeviceID: 0x7220 Rev: 0x60002
1 Connected Ports:
PortNum: 1 LID: 0x0006 GUID: 0x0002b3010100002b
Neighbor: Name: MOOSE_STL_SWITCH0
NodeGUID: 0x0002b30102000000 Type: SW PortNum: 5
LocalPort: 1 PortState: Active PhysState: LinkUp
IsSMConfigurationStarted: True NeighborNormal: True
PortType: Standard
LID: 0x0006 LMC: 0 Subnet:
0xfe80000000000000
SMLID: 0x0002 SMSL: 0 RespTimeout: 8 ms SubnetTimeout: 536
ms
M_KEY: 0x0000000000000000 Lease: 0 s Protect: Read-only
MTU Supported: (0x6) 8192 bytes
VLStallCount (per VL): 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0
MTU Active by VL:
0 00: 8192 01: 0 02: 0 03: 0 04: 0 05: 0 06: 0 07:
2048 08: 0 09: 0 10: 0 11: 0 12: 0 13: 0 14: 0 15:
0 16: 0 17: 0 18: 0 19: 0 20: 0 21: 0 22: 0 23:
0 24: 0 25: 0 26: 0 27: 0 28: 0 29: 0 30: 0 31:
0
LinkWidth: Active: 4 Supported: 1,2,3,4 Enabled: 1,2,3,4
LinkWidthDnGrade: ActiveTx: 4 Rx: 4 Supported: 1,2,3,4
Enabled: 3,4
PortLinkMode: Active: STL Supported: STL Enabled:
```




```

STL
25Gb      LinkSpeed: Active:      25Gb  Supported: 12.5Gb,25Gb  Enabled: 12.5Gb,
SM_TrapQP: 0x0  SA_QP: 0x1  IPAddr Prim/Sec: :: / 0.0.0.0
VLS:      Active:      8+1  Supported:      8+1
        HOQLife (Per VL):
        VL 0: 0x0 VL 1: 0x0 VL 2: 0x0 VL 3: 0x0 VL 4: 0x0
        VL 5: 0x0 VL 6: 0x0 VL 7: 0x0 VL 8: 0x0 VL 9: 0x0
        VL10: 0x0 VL11: 0x0 VL12: 0x0 VL13: 0x0 VL14: 0x0
        VL15: 0x0 VL16: 0x0 VL17: 0x0 VL18: 0x0 VL19: 0x0
        VL20: 0x0 VL21: 0x0 VL22: 0x0 VL23: 0x0 VL24: 0x0
        VL25: 0x0 VL26: 0x0 VL27: 0x0 VL28: 0x0 VL29: 0x0
        VL30: 0x0 VL31: 0x0
        VL Arb Cap: High:      16      Low:      16  HiLimit:      0
PreemptLimit 0
        VLFlowControlDisabledMask: 0x00000000
        NeighborMode MgmtAllowed: No  FwAuthenBypass: Off
NeighborNodeType: Switch
        Capability 0x00000000: -
        Capability3 0x0008: SS
        Violations: M_Key:      0  P_Key:      0  Q_Key:      0
        PortMode ActiveOptimize: Off  PassThrough: Off  VLMarker: Off
        FlitCtrlInterleave Distance Max: 0  Enabled: 0
        MaxNestLevelTxEnabled: 10  MaxNestLevelRxSupported: 10
        SmallPktLimit: 0x07  MaxSmallPktLimit: 0xff  PreemptionLimit: 0x10
        FlitCtrlPreemption MinInitial: 80  MinTail: 80  LargePktLim: 0x07
        BufferUnits: VL15Init 0x0040; VL15CreditRate 0x00; CreditAck 0x0;
BufferAlloc 0x3
        PortErrorActions: 0x172000: CE-UVLMCE-BCDCE-BTDCE-BHDR-BVLM
        ReplayDepth Buffer 0x00; Wire 0x00
        DiagCode: 0x0000
        OverallBufferSpace: 0x0900
        P_Key Enforcement: In: Off  Out: Off
Performance: Transmit
        Xmit Data      0 MB (5677 Flits)
        Xmit Pkts      63
        MC Xmt Pkts    0
Performance: Receive
        Rcv Data      0 MB (3762 Flits)
        Rcv Pkts      57
        MC Rcv Pkts    0
Performance: Congestion
        Congestion Discards      0
        Rcv FECN      0
        Rcv BECN      0
        Mark FECN      0
        Xmit Time Congestion      0
        Xmit Wait      0
Performance: Bubbles
        Xmit Wasted BW      0
        Xmit Wait Data      0
        Rcv Bubble      0
Errors: Signal Integrity
Link Qual Indicator      5 (Excellent)
        Uncorrectable Errors      0
        Link Downed      0
        Rcv Errors      0
        Exc. Buffer Overrun      0
        FM Config Errors      0
        Link Error Recovery      0
        Local Link Integ Err      0
        Rcv Rmt Phys Err      0
Errors: Security
        Xmit Constraint      0
        Rcv Constraint      0
Errors: Routing
        Rcv Sw Relay Err      0
        Xmit Discards      0
QSFP: ActiveCu , 1m  STL Simulator
        Power Class 5, 4.0W max S/N 0x00000000000033
    
```



```
OUI 0x0002B3
Cable Type: Linear active copper cable
Speed Sup: 0 Gb

1 Matching FIs Found

1 Connected Switches in Fabric:
0 Matching Switches Found

1 Connected SMs in Fabric:
0 Matching SMs Found

-----
```

Identify connections and links composing the fabric

```
[goblin1 root@goblin1]# opareport -o links
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Link Summary

96 Links in Fabric:
Rate NodeGUID Port Type Name
100g 0x001175010157401b 1 FI goblin8 hfi1_0
<-> 0x00117501025131cb 7 SW edge2
100g 0x001175010157403d 1 FI goblin2 hfi1_0
<-> 0x00117501025131cb 40 SW edge2
100g 0x0011750101574053 1 FI goblin12 hfi1_0
<-> 0x00117501025131cb 23 SW edge2
100g 0x001175010157405c 1 FI goblin16 hfi1_0
<-> 0x00117501025019ab 33 SW edge1
100g 0x001175010157406c 1 FI goblin13 hfi1_0
<-> 0x00117501025019ab 42 SW edge1
100g 0x0011750101574071 1 FI goblin18 hfi1_0
<-> 0x00117501025131cb 4 SW edge2
100g 0x0011750101574074 1 FI goblin15 hfi1_0
<-> 0x00117501025019ab 41 SW edge1
100g 0x0011750101574077 1 FI goblin20 hfi1_0
<-> 0x00117501025131cb 3 SW edge2
100g 0x001175010157409d 1 FI goblin1 hfi1_0
<-> 0x00117501025131cb 44 SW edge2
100g 0x00117501015740bb 1 FI goblin22 hfi1_0
<-> 0x00117501025019ab 2 SW edge1
100g 0x00117501015740bd 1 FI goblin3 hfi1_0
<-> 0x00117501025131cb 43 SW edge2
100g 0x00117501015740db 1 FI goblin21 hfi1_0
<-> 0x00117501025019ab 10 SW edge1
100g 0x00117501015740e0 1 FI goblin41 hfi1_0
<-> 0x00117501025019ab 22 SW edge1
100g 0x00117501015740e3 1 FI goblin33 hfi1_0
<-> 0x00117501025131cb 48 SW edge2
100g 0x0011750101574e8b 1 FI goblin25 hfi1_0
<-> 0x00117501025019ab 46 SW edge1
100g 0x0011750101574f08 1 FI goblin45 hfi1_0
<-> 0x00117501025019ab 18 SW edge1
100g 0x0011750101574f6c 1 FI goblin42 hfi1_0
<-> 0x00117501025019ab 30 SW edge1
100g 0x0011750101574fea 1 FI goblin29 hfi1_0
<-> 0x00117501025131cb 32 SW edge2
100g 0x0011750101575021 1 FI goblin46 hfi1_0
<-> 0x00117501025019ab 25 SW edge1
100g 0x001175010157504e 1 FI goblin47 hfi1_0
<-> 0x00117501025019ab 17 SW edge1
100g 0x0011750101575068 1 FI goblin10 hfi1_0
<-> 0x00117501025131cb 24 SW edge2
100g 0x0011750101575082 1 FI goblin23 hfi1_0
<-> 0x00117501025019ab 9 SW edge1
```



```

100g 0x00117501015750a1 1 FI goblin48 hfi1_0
<-> 0x00117501025019ab 26 SW edge1
100g 0x001175010157513c 1 FI goblin34 hfi1_0
<-> 0x00117501025131cb 47 SW edge2
100g 0x0011750101575153 1 FI goblin35 hfi1_0
<-> 0x00117501025131cb 36 SW edge2
100g 0x001175010157515e 1 FI goblin4 hfi1_0
<-> 0x00117501025131cb 39 SW edge2
100g 0x0011750101575188 1 FI goblin17 hfi1_0
<-> 0x00117501025131cb 16 SW edge2
100g 0x00117501015751b8 1 FI goblin11 hfi1_0
<-> 0x00117501025131cb 27 SW edge2
100g 0x00117501015751c9 1 FI goblin30 hfi1_0
<-> 0x00117501025131cb 19 SW edge2
100g 0x00117501015751d6 1 FI goblin6 hfi1_0
<-> 0x00117501025131cb 8 SW edge2
100g 0x00117501015751dd 1 FI goblin37 hfi1_0
<-> 0x00117501025019ab 14 SW edge1
100g 0x00117501015751df 1 FI goblin43 hfi1_0
<-> 0x00117501025019ab 29 SW edge1
100g 0x00117501015751e5 1 FI goblin31 hfi1_0
<-> 0x00117501025131cb 20 SW edge2
100g 0x00117501015751ef 1 FI goblin38 hfi1_0
<-> 0x00117501025019ab 5 SW edge1
100g 0x00117501015751ff 1 FI goblin19 hfi1_0
<-> 0x00117501025131cb 15 SW edge2
100g 0x0011750101575f28 1 FI goblin39 hfi1_0
<-> 0x00117501025019ab 6 SW edge1
100g 0x0011750101575f63 1 FI goblin26 hfi1_0
<-> 0x00117501025019ab 45 SW edge1
100g 0x0011750101575f6a 1 FI goblin44 hfi1_0
<-> 0x00117501025019ab 21 SW edge1
100g 0x0011750101575fa1 1 FI goblin40 hfi1_0
<-> 0x00117501025019ab 13 SW edge1
100g 0x0011750101575fba 1 FI goblin7 hfi1_0
<-> 0x00117501025131cb 11 SW edge2
100g 0x0011750101575feb 1 FI goblin14 hfi1_0
<-> 0x00117501025019ab 34 SW edge1
100g 0x001175010157e3d1 1 FI goblin36 hfi1_0
<-> 0x00117501025131cb 35 SW edge2
100g 0x001175010157e3f0 1 FI goblin24 hfi1_0
<-> 0x00117501025019ab 1 SW edge1
100g 0x001175010157e3f3 1 FI goblin32 hfi1_0
<-> 0x00117501025131cb 31 SW edge2
100g 0x001175010157e406 1 FI goblin27 hfi1_0
<-> 0x00117501025019ab 38 SW edge1
100g 0x001175010157e40e 1 FI goblin9 hfi1_0
<-> 0x00117501025131cb 28 SW edge2
100g 0x001175010157e418 1 FI goblin28 hfi1_0
<-> 0x00117501025019ab 37 SW edge1
100g 0x001175010157e427 1 FI goblin5 hfi1_0
<-> 0x00117501025131cb 12 SW edge2
100g 0x00117501025019ab 3 SW edge1
<-> 0x0011750102513145 3 SW edge4
100g 0x00117501025019ab 4 SW edge1
<-> 0x0011750102513145 4 SW edge4
100g 0x00117501025019ab 7 SW edge1
<-> 0x0011750102513145 7 SW edge4
100g 0x00117501025019ab 8 SW edge1
<-> 0x0011750102513145 8 SW edge4
100g 0x00117501025019ab 11 SW edge1
<-> 0x0011750102513139 11 SW edge3
100g 0x00117501025019ab 12 SW edge1
<-> 0x0011750102513139 12 SW edge3
100g 0x00117501025019ab 15 SW edge1
<-> 0x0011750102513139 15 SW edge3
100g 0x00117501025019ab 16 SW edge1
<-> 0x0011750102513139 16 SW edge3
100g 0x00117501025019ab 19 SW edge1
<-> 0x0011750102513145 19 SW edge4
100g 0x00117501025019ab 20 SW edge1

```



```
<-> 0x0011750102513145 20 SW edge4
100g 0x00117501025019ab 23 SW edge1
<-> 0x0011750102513145 23 SW edge4
100g 0x00117501025019ab 24 SW edge1
<-> 0x0011750102513145 24 SW edge4
100g 0x00117501025019ab 27 SW edge1
<-> 0x0011750102513139 27 SW edge3
100g 0x00117501025019ab 28 SW edge1
<-> 0x0011750102513139 28 SW edge3
100g 0x00117501025019ab 31 SW edge1
<-> 0x0011750102513139 31 SW edge3
100g 0x00117501025019ab 32 SW edge1
<-> 0x0011750102513139 32 SW edge3
100g 0x00117501025019ab 35 SW edge1
<-> 0x0011750102513145 35 SW edge4
100g 0x00117501025019ab 36 SW edge1
<-> 0x0011750102513145 36 SW edge4
100g 0x00117501025019ab 39 SW edge1
<-> 0x0011750102513145 39 SW edge4
100g 0x00117501025019ab 40 SW edge1
<-> 0x0011750102513145 40 SW edge4
100g 0x00117501025019ab 43 SW edge1
<-> 0x0011750102513139 43 SW edge3
100g 0x00117501025019ab 44 SW edge1
<-> 0x0011750102513139 44 SW edge3
100g 0x00117501025019ab 47 SW edge1
<-> 0x0011750102513139 47 SW edge3
100g 0x00117501025019ab 48 SW edge1
<-> 0x0011750102513139 48 SW edge3
100g 0x0011750102513139 1 SW edge3
<-> 0x00117501025131cb 1 SW edge2
100g 0x0011750102513139 2 SW edge3
<-> 0x00117501025131cb 2 SW edge2
100g 0x0011750102513139 5 SW edge3
<-> 0x00117501025131cb 5 SW edge2
100g 0x0011750102513139 6 SW edge3
<-> 0x00117501025131cb 6 SW edge2
100g 0x0011750102513139 17 SW edge3
<-> 0x00117501025131cb 17 SW edge2
100g 0x0011750102513139 18 SW edge3
<-> 0x00117501025131cb 18 SW edge2
100g 0x0011750102513139 21 SW edge3
<-> 0x00117501025131cb 21 SW edge2
100g 0x0011750102513139 22 SW edge3
<-> 0x00117501025131cb 22 SW edge2
100g 0x0011750102513139 33 SW edge3
<-> 0x00117501025131cb 33 SW edge2
100g 0x0011750102513139 34 SW edge3
<-> 0x00117501025131cb 34 SW edge2
100g 0x0011750102513139 37 SW edge3
<-> 0x00117501025131cb 37 SW edge2
100g 0x0011750102513139 38 SW edge3
<-> 0x00117501025131cb 38 SW edge2
100g 0x0011750102513145 9 SW edge4
<-> 0x00117501025131cb 9 SW edge2
100g 0x0011750102513145 10 SW edge4
<-> 0x00117501025131cb 10 SW edge2
100g 0x0011750102513145 13 SW edge4
<-> 0x00117501025131cb 13 SW edge2
100g 0x0011750102513145 14 SW edge4
<-> 0x00117501025131cb 14 SW edge2
100g 0x0011750102513145 25 SW edge4
<-> 0x00117501025131cb 25 SW edge2
100g 0x0011750102513145 26 SW edge4
<-> 0x00117501025131cb 26 SW edge2
100g 0x0011750102513145 29 SW edge4
<-> 0x00117501025131cb 29 SW edge2
100g 0x0011750102513145 30 SW edge4
<-> 0x00117501025131cb 30 SW edge2
100g 0x0011750102513145 41 SW edge4
<-> 0x00117501025131cb 41 SW edge2
```



```
100g 0x0011750102513145 42 SW edge4
<-> 0x00117501025131cb 42 SW edge2
100g 0x0011750102513145 45 SW edge4
<-> 0x00117501025131cb 45 SW edge2
100g 0x0011750102513145 46 SW edge4
<-> 0x00117501025131cb 46 SW edge2
```

Reverse lookup

The following example translates a LID or GUID into the information about the node or port represented.

```
[root@duster duster]# opareport -o nodes -F lid:5
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Node Type Summary
Focused on:
  Port: 1 0x0011750101574071
    in Node: 0x0011750101574071 FI goblin2 hfil_0

48 Connected FIs in Fabric:
  Name: goblin2 hfil_0
    NodeGUID: 0x0011750101574071 Type: FI
    Ports: 1 PartitionCap: 16 SystemImageGuid: 0x0011750101574071
    BaseVer: 128 SmaVer: 128 VendorID: 0x1175 DeviceID: 0x24f0 Rev: 0x0
  1 Connected Ports:
    PortNum: 1 LID: 0x0005 GUID: 0x0011750101574071
      Neighbor: Name: edgel
        NodeGUID: 0x00117501025131cb Type: SW PortNum: 4
        Width: 4 Speed: 25Gb Downgraded? No
1 Matching FIs Found

4 Connected Switches in Fabric:
0 Matching Switches Found

1 Connected SMs in Fabric:
0 Matching SMs Found
-----
```

Forward lookup

The following example returns information about nodes or IOCs listed by name.

```
[root@duster root]# opareport -o nodes -F "node:goblin2 hfil_0"
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Node Type Summary
Focused on:
  Node: 0x0011750101574071 FI goblin2 hfil_0

48 Connected FIs in Fabric:
  Name: goblin2 hfil_0
    NodeGUID: 0x0011750101574071 Type: FI
    Ports: 1 PartitionCap: 16 SystemImageGuid: 0x0011750101574071
    BaseVer: 128 SmaVer: 128 VendorID: 0x1175 DeviceID: 0x24f0 Rev: 0x0
  1 Connected Ports:
    PortNum: 1 LID: 0x0005 GUID: 0x0011750101574071
      Neighbor: Name: edgel
        NodeGUID: 0x00117501025131cb Type: SW PortNum: 4
        Width: 4 Speed: 25Gb Downgraded? No
```



```
1 Matching FIs Found

4 Connected Switches in Fabric:
0 Matching Switches Found

1 Connected SMs in Fabric:
0 Matching SMs Found
-----
```

Generate report for comparison

The following example generates a report so topology verification can be performed against a known good configuration.

Note: To shorten the length of the output, the following example focuses on only one node.

```
[root@phwtpriv27 ~]$ opareport -o nodes -F node:"goblin02 hfil_0" -d 5 -P
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Done Getting vFabric Records
Node Type Summary
Focused on:
  Node: 0x0002b3010100002b FI goblin02 hfil_0

48 Connected FIs in Fabric:
  Name: goblin02 hfil_0
  NodeGUID: 0x0002b3010100002b Type: FI
  Ports: 1 PartitionCap: 16 SystemImageGuid: 0x0002b3010100002b
  BaseVer: 128 SmaVer: 128 VendorID: 0x2b3 DeviceID: 0x7220 Rev: 0x60002
  1 Connected Ports:
    PortNum: 1 LID: xxxxxx GUID: 0x0002b3010100002b
      Neighbor: Name: MOOSE_STL_SWITCH0
        NodeGUID: 0x0002b30102000000 Type: SW PortNum: 5
        LocalPort: 1 PortState: Active PhysState: LinkUp
        IsSMConfigurationStarted: True NeighborNormal: True
        PortType: Standard
        LID: xxxxxx LMC: 0 Subnet:
0xfe80000000000000
ms      SMLID: xxxxxx SMSL: 0 RespTimeout: 8 ms SubnetTimeout: 536
      M KEY: 0x0000000000000000 Lease: 0 s Protect: Read-only
      MTU Supported: (0x6) 8192 bytes
      VLStallCount (per VL): 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
      MTU Active by VL:
0      00: 8192 01: 0 02: 0 03: 0 04: 0 05: 0 06: 0 07:
2048    08: 0 09: 0 10: 0 11: 0 12: 0 13: 0 14: 0 15:
0      16: 0 17: 0 18: 0 19: 0 20: 0 21: 0 22: 0 23:
0      24: 0 25: 0 26: 0 27: 0 28: 0 29: 0 30: 0 31:
0
      LinkWidth: Active: 4 Supported: 1,2,3,4 Enabled: 1,2,3,4
      LinkWidthDnGrade: ActiveTx: 4 Rx: 4 Supported: 1,2,3,4
Enabled: 3,4
      PortLinkMode: Active: STL Supported: STL Enabled:
STL
      LinkSpeed: Active: 25Gb Supported: 12.5Gb,25Gb Enabled: 12.5Gb,
25Gb
      SM_TrapQP: 0x0 SA_QP: 0x1 IPAddr Prim/Sec: :: / 0.0.0.0
      VLs: Active: 8+1 Supported: 8+1
          HOQLife (Per VL):
          VL 0: 0x0 VL 1: 0x0 VL 2: 0x0 VL 3: 0x0 VL 4: 0x0
```



```

VL 5: 0x0 VL 6: 0x0 VL 7: 0x0 VL 8: 0x0 VL 9: 0x0
VL10: 0x0 VL11: 0x0 VL12: 0x0 VL13: 0x0 VL14: 0x0
VL15: 0x0 VL16: 0x0 VL17: 0x0 VL18: 0x0 VL19: 0x0
VL20: 0x0 VL21: 0x0 VL22: 0x0 VL23: 0x0 VL24: 0x0
VL25: 0x0 VL26: 0x0 VL27: 0x0 VL28: 0x0 VL29: 0x0
VL30: 0x0 VL31: 0x0

VL Arb Cap: High:      16      Low:      16 HiLimit:    0
PreemptLimit 0
VlFlowControlDisabledMask: 0x00000000
NeighborMode MgmtAllowed: No FwAuthenBypass: Off
NeighborNodeType: Switch
Capability 0x00000000: -
Capability3 0x0008: SS
Violations: M_Key: xxxxx P_Key: xxxxx Q_Key: xxxxx
PortMode ActiveOptimize: Off PassThrough: Off VLMarker: Off
FlitCtrlInterleave Distance Max: 0 Enabled: 0
MaxNestLevelTxEnabled: 10 MaxNestLevelRxSupported: 10
SmallPktLimit: 0x07 MaxSmallPktLimit: 0xff PreemptionLimit: 0x10
FlitCtrlPreemption MinInitial: 80 MinTail: 80 LargePktLim: 0x07
BufferUnits: VL15Init 0x0040; VL15CreditRate 0x00; CreditAck 0x0;
BufferAlloc 0x3
PortErrorActions: 0x172000: CE-UVLMCE-BCDCE-BTDCE-BHDR-BVLM
ReplayDepth Buffer 0x00; Wire xxxx
DiagCode: 0x0000
OverallBufferSpace: 0x0900
P_Key Enforcement: In: Off Out: Off
QSFP: ActiveCu , 1m STL Simulator
Power Class 5, 4.0W max S/N 0x00000000000033
OUI 0x0002B3
Cable Type: Linear active copper cable
Speed Sup: 0 Gb

1 Matching FIs Found

1 Connected Switches in Fabric:
0 Matching Switches Found

1 Connected SMs in Fabric:
0 Matching SMs Found
-----

```

5.3.11.11 Snapshots

You can take a *snapshot* of the fabric state for later offline analysis using the `-o snapshot` report. This report generates an XML snapshot of the present fabric status in a format that `opareport` can parse.

Note: Intel recommends that you do **not** develop your own tools against this format because it may change in future versions of `opareport`.

The snapshot capability can be used to provide powerful analysis capabilities. Multiple reports can be run against the exact same fabric snapshot, which saves time by not requiring the subsequent reports to query the fabric. Also, historic snapshots can be retained for later offline analysis or historical tracking of the fabric.

When a snapshot is generated, no additional `-o` options are allowed during the run and certain `opareport` options are ignored. These include: `-F`, `-P`, `-H`, and `-N`. However, the following options are valid:

- `-s` includes port counters in the snapshot.
- `-r` includes switch routing tables in the snapshot.
- `-v` includes QoS VL-related tables in the snapshot.



- `-i`, `-L`, `-a`, and `-C` control the port counters.

Notes: Quarantined nodes cannot be obtained from a snapshot.

- `opareport -o quarantinednodes -X snapshot` does not give the quarantined nodes on a snapshot of the same fabric.
- Use `opareport -o quarantinednodes` to return the quarantined nodes on a fabric with quarantined nodes.

After a snapshot has been generated, it may then be used as input to generate many types of `opareport` reports. To do this, use the `-X snapshot_input` option, where the `snapshot_input` file is the output from a previous snapshot run. When using a snapshot as input, the fabric is not accessed and the node running `opareport` does not need to be attached to the fabric. Because this is a static report, certain options are not available, including `-i`, `-a`, `-C`, `-h HFI`, and `-p port`.

The report generated from the snapshot includes port counters **only** if the original snapshot was run with the `-s` option. If not, reports such as `-o errors` are not permitted against the snapshot.

Similarly, certain reports are permitted **only** if the original snapshot was run with the `-r` option. This includes: `-o linear`, `-o mcast`, `-o portusage`, `-o pathusage`, `-o treepathusage`, and `-o route`.

If you want to use standard input (`stdin`) for the snapshot file, then specify `-X`. This can be helpful if snapshots are piped through `gzip/gunzip` to conserve disk space.

Notes: Limitations of `-o route`:

- The Path Records reported may not be complete. The report shows the minimum valid value or an invalid value because certain fields such as `SLID`, `SL`, `PKey`, `MTU`, `Rate`, and `PktLifeTime` are not available. These values do not impact the actual route shown.
- Some routes reported may not be incomplete or not available to applications.

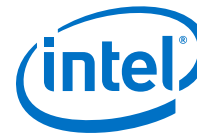
5.3.12 **opaverifyhosts**

Verifies basic node configuration and performance by running `FF_HOSTVERIFY_DIR/hostverify.sh` on all specified hosts.

Note: Prior to using `opaverifyhosts`, copy the sample file `/usr/share/opa/samples/hostverify.sh` to `FF_HOSTVERIFY_DIR` and edit it to set the appropriate configuration and performance expectations and select which tests to run by default. On the first run for a given node, use the `-c` option so that `hostverify.sh` gets copied to each node.

`FF_HOSTVERIFY_DIR` defines both the location of `hostverify.sh` and the destination of the `hostverify.res` output file. `FF_HOSTVERIFY_DIR` is configured in the `/etc/opa/opafastfabric.conf` file.

A summary of results is appended to the `FF_RESULT_DIR/verifyhosts.res` file. A punchlist of failures is also appended to the `FF_RESULT_DIR/punchlist.csv` file. Only failures are shown on stdout.



Syntax

```
opaverifyhosts [-kc] [-f hostfile] [-u upload_file] [-d upload_dir]
[-h hosts] [-T timelimit] [-F filename] [test ...]
```

Options

<code>--help</code>	Produces full help text.
<code>-k</code>	At start and end of verification, kills any existing hostverify or xhpl jobs on the hosts.
<code>-c</code>	Copies <code>hostverify.sh</code> to hosts first, useful if you have edited it.
<code>-f <i>hostfile</i></code>	Specifies the file with hosts in cluster. Default is <code>/etc/opa/hosts</code> .
<code>-h <i>hosts</i></code>	Specifies the list of hosts to ping.
<code>-u <i>upload_file</i></code>	Specifies the filename to upload <code>hostverify.res</code> to after verification to allow backup and review of the detailed results for each node. The default upload destination file is <code>hostverify.res</code> . If <code>-u ''</code> is specified, no upload occurs.
<code>-d <i>upload_dir</i></code>	Specifies the directory to upload result from each host to. Default is <code>uploads</code> .
<code>-T <i>timelimit</i></code>	Specifies the time limit in seconds for host to complete tests. Default is 300 seconds (5 minutes).
<code>-F <i>filename</i></code>	Specifies the filename of <code>hostverify</code> script to use. Default is <code>/root/hostverify.sh</code> .
<code><i>test</i></code>	Specifies one or more specific tests to run. See <code>/usr/share/opa/samples/hostverify.sh</code> for a list of available tests.

Note: Intel® Xeon Phi™ Processors operate in X2Apic Mode, which requires that the Intel® VT for Directed I/O (VT-d) remain enabled. As a result, the `vtd` test that checks if VT-D is disabled is not applicable.

Examples

```
opaverifyhosts -c
opaverifyhosts -h 'arwen elrond'
HOSTS='arwen elrond' opaverifyhosts
```

Environment Variables

HOSTS	List of hosts, used if <code>-h</code> option not supplied.
-------	---



HOSTS_FILE	File containing list of hosts, used in absence of <code>-f</code> and <code>-h</code> .
UPLOADS_DIR	Directory to upload to, used in absence of <code>-d</code> .
FF_MAX_PARALLEL	Maximum concurrent operations.

5.3.13 opaxlattopology

Generates a topology XML file of a cluster using `topology.csv`, `linksum_swd06.csv`, and `linksum_swd24.csv` as input. The topology file can be used to bring up and verify the cluster.

Note: The `topology.csv` input file must be present in the same directory from which the script operates, but the `linksum` CSV files are read from the `/usr/share/opa/samples` directory.

For more information, see [Sample Files](#) on page 49 and [topology.xlsx Overview](#) on page 53.

Syntax

```
opaxlattopology [-d level] [-v level] [-i level] [-K] [-s hfi_suffix] [source [dest]]
```

Options

<code>--help</code>	Produces full help text.
<code>-d level</code>	<p>Specifies the output detail level. Default = 0. Levels are additive.</p> <p>By default, the top level is always produced. Switch, rack, and rack group topology files can be added to the output by choosing the appropriate level. If the output at the group or rack level is specified, then group or rack names must be provided in the spreadsheet. Detailed output can be specified in any combination. A directory for each topology XML file is created hierarchically, with group directories (if specified) at the highest level, followed by rack and switch directories (if specified).</p> <ol style="list-style-type: none">1 Intel® Omni-Path Edge Switch 100 Series topology files.2 Rack topology files.4 Rack group topology files.
<code>-v level</code>	<p>Specifies the verbose level. Range = 0 - 8. Default = 2.</p> <ol style="list-style-type: none">0 No output.1 Progress output.



	2	Reserved.
	4	Time stamps.
	8	Reserved.
<code>-i level</code>		Specifies the output indent level. Default = 4.
<code>-K</code>		Specifies DO NOT clean temporary files. Prevents temporary files in each topology directory from being removed. Temporary files contain CSV formatted lists of links, HFIs, and switches used to create a topology XML file. Temporary files are not typically needed after a topology file is created, however they are used for creating <code>linksum_swd06.csv</code> and <code>linksum_swd24.csv</code> files, or can be retained for subsequent inspection or processing.
<code>-s hfi_suffix</code>		Used on Multi-Rail or Multi-Plane fabrics. Can be used to override the default <code>hfi1_0</code> . For Multi-Plane fabric, use the tool multiple times with a different hfi-suffix. For Multi-Rail fabric, specify HostName as "HostName HfiName" in the spreadsheet.

Description

The `opaxlattopology` script reads the `topology.csv` file from the local directory, and reads the other files from `/usr/share/opa/samples/linksum_swd06.csv` and `/usr/share/opa/samples/linksum_swd24.csv`. The `topology.csv` file is created from the `topology.xlsx` spreadsheet by saving the Fabric Links tab as a .CSV file to `topology.csv`. A sample `topology.xlsx` is located in the `/usr/share/opa/samples/` directory. Inspect the `topology.csv` file to ensure that each row contains the correct and same number of comma separators. Any extraneous entries in the spreadsheet can cause the CSV output to have extra fields.

The script outputs one or more topology files starting with `topology.0:0.xml`. The `topology.csv` input file must be present in the same directory from which the script operates, but the `linksum` CSV files are read from the `/usr/share/opa/samples` directory.

Example

```
opaxlattopology
# reads default input 'topology.csv' and creates default
# output 'topology.0:0.xml'

opaxlattopology fabric_2.csv
# reads input 'fabric_2.csv' and creates default output
```



See `topology.xlsx` for examples of links between HFI and Edge SW (rows 4-7), HFI and Core SW (rows 8-11), and Edge SW and Core SW (rows 12-15).

Environment Variables

The following environment variables allow user-specified MTU.

`MTU_SW_SW` If set, it overrides default MTU on switch-to-switch links. Default = 10240

`MTU_SW_HFI` If set, it overrides default MTU on switch-to-HFI links. Default = 10240

Creating linksum Files

The `linksum_swd06.csv` and `linksum_swd24.csv` files are provided as stand-alone files in the `/usr/share/opa/samples` directory. However, they can be recreated (or modified) from the spreadsheet, if needed, by performing the following steps:

1. Save each of the following from the `topology.xlsx` file as individual `.csv` files:
 - Internal SWD06 Links tab as `linksum_swd06.csv`
 - Internal SWD24 Links tab as `linksum_swd24.csv`
 - Fabric Links tab as `topology.csv`
2. For each saved `topology.csv` file, run the script with the `-K` option.
3. Upon completion of the script, save the top level `linksum.csv` file as `linksum_swd06.csv` or `linksum_swd24.csv` as appropriate.

Multi Rail and Multi Plane Fabrics

Note: *Planes* can also be referred to as *subnets* or *fabrics*. *Rails* are also referred to as *HFIs*.

By default, the `opaxlattopology` script considers all of the hosts to have a single HFI (`hfi1_0`).

For Multi Rail/Plane fabrics, the user has following options:

- For Multi Rail fabrics or for a Single Plane fabric with some multi-ported hosts, the user can edit the "Host Name" in `topology.csv` file to include the HFI Name, like "HostName HfiName" and then follow the standard procedure to generate `topology.xml`.
- For a Multi Plane fabric with Identical planes, the tool can be run multiple times on same `topology.csv` with different `"-s hfi_suffix"` options.

For example, if there are two identical fabric (`fabric_1` and `fabric_2`) connected to a single host with two HFIs (`hfi1_0` and `hfi1_1`), the tool can be run twice like this:

For `fabric_1`:

```
opaxlattopology topology.csv topology.xml
```

For `fabric_2`:

```
opaxlattopology -s hfi1_1 topology.csv topology.xml
```



Note: Default for "-s hfi_suffix" is hfi1_0

- For a fabric with both Multi Rail and Multi Plane segments, the user can use a combination of above techniques to generate desired topology.xml file.

For complete details on topology.xlsx and topology.csv, see [topology.xlsx Overview](#) on page 53.

5.3.14 opaxlattopology_cust

Note: This tool has been deprecated.

Customizable script for documenting cluster topology. Provides an alternative to the standard script (see [opaxlattopology](#) on page 210). Edit the sample topology_cust.xlsx to represent each external link in a cluster, then modify opaxlattopology_cust to translate the alternate CSV form to the standard CSV form used by opaxlattopology.

Syntax

```
opaxlattopology_cust [-t topology_prime] [-s topology_second] [-T topology_out]
[-v level] [-i level] [-K]
```

Options

<code>--help</code>	Produces full help text.
<code>-t topology_prime</code>	Specifies the primary topology CSV input file. Specifies the primary CSV input file and must be present.
<code>-s topology_second</code>	Specifies the secondary topology CSV input file. Specifies a secondary CSV input file. Appended to the primary for processing.
<code>-T topology_out</code>	Specifies the topology CSV output file. Specifies the CSV output file name and must be specified.
<code>-v level</code>	Specifies the verbose level. Range = 0 - 8, default = 2. 0 No output. 1 Progress output. 2 Reserved. 4 Time stamps. 8 Reserved.
<code>-i level</code>	Specifies the screen output indent level. Range = 0 - 15, default = 0.



- K Specifies DO NOT clean temporary files.
- Prevents temporary files from being removed. Temporary files contain CSV data used during processing. Temporary files are not needed after the standard-format CSV file is created, but they can be retained for subsequent inspection or processing.

Description

Each link contains source, destination, and cable fields with one link per row of the spreadsheet. Link fields must not contain commas. Source and Destination fields are each a concatenation of name and port information in the following forms. Names not of the form `ib` or `C` are assumed to be host names.

The following lists the `node type` and source/destination.

Host: `hostN` where N is a host number.

Edge Switch: `opaNpN` where N is a switch/port number.

Core Leaf: `Cn Lnnn pN` where N/n is a host/switch/port number.

Cable values, CableLength, and CableDetails are optional and have no special syntax. If present, they are placed in the standard-format CSV file exactly as they appear. CableLabel is created automatically by `opaxlattopology_cust` as the concatenation of Source and Destination.

Rack Group and Rack are not supported in `topology_cust.xlsx`. Therefore, `opaxlattopology_cust` leaves these fields empty in the standard-format CSV file.

5.4 Detailed Fabric Data Gathering

The CLIs described in this section are used for gathering general fabric data for further analysis. Some commands produce text files while others produce files in CSV format that may be imported into Microsoft® Excel.

5.4.1 opaextracterror

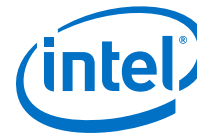
Produces a CSV file listing all or some of the errors in the current fabric. `opaextracterror` is a front end to the `opareport` tool. The output from this tool can be imported into a spreadsheet or parsed by other scripts.

Syntax

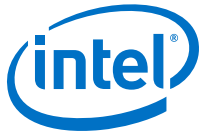
```
opaextracterror [opareport options]
```

Options

- help Produces full help text.



<code>opareport</code> <i>options</i>	The following options are passed to <code>opareport</code> . This subset is considered typical and useful for this command.	
<code>-h/--hfi hfi</code>		Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p/--port port</code>		Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-X/--infile snapshot_input</code>		Generates a report using the data in the <code>snapshot_input</code> file. <code>snapshot_input</code> must have been generated during a previous <code>-o snapshot</code> run. When used, the <code>-s</code> , <code>-i</code> , <code>-C</code> , and <code>-a</code> options are ignored. '-' may be used as the <code>snapshot_input</code> to specify <code>stdin</code> .
<code>-T/--topology topology_input</code>		Uses <code>topology_input</code> file to augment and verify fabric information. When used, various reports can be augmented with information not available electronically (such as cable labels). '-' may be used to specify <code>stdin</code> .
<code>-i/--interval seconds</code>		Obtains performance statistics over interval <i>seconds</i> . Clears all statistics, waits interval <i>seconds</i> , then generates report. Implies <code>-s</code> option.
<code>-b/--begin date_time</code>		Obtains past performance stats over an interval beginning at <i>date_time</i> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second] minute hour day](s) ago.
<code>-e/--end date_time</code>		Obtains past performance stats over an interval ending at <i>date_time</i> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second] minute hour day](s) ago.



<code>-C/--clear</code>	Clears performance statistics for all ports. Only statistics with error thresholds are cleared. A clear occurs after generating the report.
<code>-a/--clearall</code>	Clears all performance statistics for all ports.
<code>-M/--pmadirect</code>	Accesses performance statistics using direct PMA.
<code>-A/--allports</code>	Gets PortInfo for down switch ports. Uses direct SMA to get this data. If used with <code>-M</code> , also gets PMA stats for down switch ports.
<code>-F/--focus <i>point</i></code>	Specifies the focus area for report. Used for all reports except <code>route</code> to limit scope of report. Refer to Point Syntax on page 177 for details.

-h and -p options permit a variety of selections:

- `-h 0` First active port in system (default).
- `-h 0 -p 0` First active port in system.
- `-h x` First active port on HFI *x*.
- `-h x -p 0` First active port on HFI *x*.
- `-h 0 -p y` Port *y* within system (no matter which ports are active).
- `-h x -p y` HFI *x*, port *y*.

Examples

```
# List all the link errors in the fabric:
opaextracterror

# List all the link errors related to a switch named "coresw1":
opaextracterror -F "node:coresw1"

# List all the link errors for end-nodes:
opaextracterror -F "nodetype:FI"

# List all the link errors on the 2nd HFI's fabric of a multi-plane fabric:
opaextracterror -h 2
```

5.4.2 opaextractlids

Produces a CSV file listing all or some of the LIDs in the fabric. `opaextractlids` is a front end to the `opareport` tool. The output from this tool can be imported into a spreadsheet or parsed by other scripts.



Syntax

```
opaextractlids [opareport options]
```

Options

<code>--help</code>	Produces full help text.
<code>opareport options</code>	The following options are passed to <code>opareport</code> . This subset is considered typical and useful for this command.
<code>-h/--hfi hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p/--port port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-X/--infile snapshot_input</code>	Generates a report using the data in the <code>snapshot_input</code> file. <code>snapshot_input</code> must have been generated during a previous <code>-o snapshot</code> run. When used, the <code>-s</code> , <code>-i</code> , <code>-C</code> , and <code>-a</code> options are ignored. '-' may be used as the <code>snapshot_input</code> to specify stdin.
<code>-F/--focus point</code>	Specifies the focus area for report. Used for all reports except <code>route</code> to limit scope of report. Refer to Point Syntax on page 177 for details.

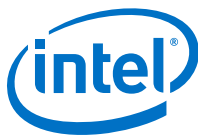
-h and -p options permit a variety of selections:

<code>-h 0</code>	First active port in system (default).
<code>-h 0 -p 0</code>	First active port in system.
<code>-h x</code>	First active port on HFI x.
<code>-h x -p 0</code>	First active port on HFI x.
<code>-h 0 -p y</code>	Port y within system (no matter which ports are active).
<code>-h x -p y</code>	HFI x, port y.

Examples

```
# List all the lids in the fabric:
opaextractlids

# List all the lids of end-nodes:
```



```
opaextractlids -F "nodetype:FI"

# List all the lids on the 2nd HFI's fabric of a multi-plane fabric:
opaextractlids -h 2
```

5.4.3 opaextractperf

Provides a report of all performance counters in a CVS format suitable for importing into a spreadsheet or parsed by other scripts for further analysis. It generates a detailed `opareport` component summary report and pipes the result to `opaxmlextract`, extracting element values for `NodeDesc`, `SystemImageGUID`, `PortNum`, and all the performance counters. Extraction is performed only from the Systems portion of the report, which does not contain Neighbor information (the Neighbor and SMs portions are suppressed).

Syntax

```
opaextractperf [opareport options]
```

Options

<code>--help</code>	Produces full help text.
<code>opareport options</code>	The following options are passed to <code>opareport</code> . This subset is considered typical and useful for this command.
<code>-h/--hfi hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p/--port port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-X/--infile snapshot_input</code>	Generates a report using the data in the <code>snapshot_input</code> file. <code>snapshot_input</code> must have been generated during a previous <code>-o snapshot</code> run. When used, the <code>-s</code> , <code>-i</code> , <code>-C</code> , and <code>-a</code> options are ignored. '-' may be used as the <code>snapshot_input</code> to specify <code>stdin</code> .
<code>-T/--topology topology_input</code>	Uses <code>topology_input</code> file to augment and verify fabric information. When used, various reports can be augmented with information not available electronically (such as cable labels). '-' may be used to specify <code>stdin</code> .
<code>-i/--interval seconds</code>	Obtains performance statistics over interval <code>seconds</code> . Clears all statistics, waits interval <code>seconds</code> , then generates report. Implies <code>-s</code> option.



<code>-b/--begin date_time</code>	Obtains past performance stats over an interval beginning at <i>date_time</i> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago".
<code>-e/--end date_time</code>	Obtains past performance stats over an interval ending at <i>date_time</i> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago".
<code>-C/--clear</code>	Clears performance statistics for all ports. Only statistics with error thresholds are cleared. A clear occurs after generating the report.
<code>-a/--clearall</code>	Clears all performance statistics for all ports.
<code>-M/--pmadirect</code>	Accesses performance statistics using direct PMA.
<code>-A/--allports</code>	Gets PortInfo for down switch ports. Uses direct SMA to get this data. If used with <code>-M</code> , also gets PMA stats for down switch ports.
<code>-F/--focus point</code>	Specifies the focus area for report. Used for all reports except <code>route</code> to limit scope of report. Refer to Point Syntax on page 177 for details.

-h and -p options permit a variety of selections:

<code>-h 0</code>	First active port in system (default).
<code>-h 0 -p 0</code>	First active port in system.
<code>-h x</code>	First active port on HFI x.
<code>-h x -p 0</code>	First active port on HFI x.



-h 0 -p y Port y within system (no matter which ports are active).

-h x -p y HFI x, port y.

The portion of the script that calls `opareport` and `opaxmlextract` follows:

```
opareport -o comps -s -x -d 10 $@ | opaxmlextract -d \;  
-e NodeDesc -e SystemImageGUID -e PortNum -e XmitDataMB  
-e XmitData -e XmitPkts -e RcvDataMB -e RcvData -e RcvPkts  
-e SymbolErrors -e LinkErrorRecovery -e LinkDowned -e PortRcvErrors  
-e PortRcvRemotePhysicalErrors -e PortRcvSwitchRelayErrors  
-e PortXmitDiscards -e PortXmitConstraintErrors  
-e PortRcvConstraintErrors -e LocalLinkIntegrityErrors  
-e ExcessiveBufferOverrunErrors -e VL15Dropped -s Neighbor -s SMs
```

Example .

```
opaextractperf  
opaextractperf -h 1 -p 2
```

5.4.4 opaextractstat

Performs an error analysis of a fabric and provides augmented information from a `topology_file`. The report provides cable information as well as symbol error counts.

`opaextractstat` generates a detailed `opareport` errors report that also has a topology file (see [opareport](#) on page 171 for more information about topology files). The report is piped to `opaxmlextract` which extracts values for Link, Cable and Port. (The port element names are context-sensitive.) Note that `opaxmlextract` generates two extraction records for each link (one for each port on the link); therefore, `opaextractstat` merges the two records into a single record and removes redundant link and cable information.

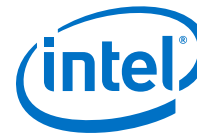
`opaextractstat` contains a `while read` loop that reads the CSV line-by-line, uses `cut` to remove redundant information, and outputs the data on a common line.

Syntax

```
opaextractstat topology_file [opareport options]
```

Options

<code>--help</code>	Produces full help text.
<code>topology_file</code>	Specifies <code>topology_file</code> to use.
<code>opareport options</code>	The following options are passed to <code>opareport</code> . This subset is considered typical and useful for this command.



<code>-h/--hfi hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p/--port port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-X/--infile snapshot_input</code>	Generates a report using the data in the <code>snapshot_input</code> file. <code>snapshot_input</code> must have been generated during a previous <code>-o snapshot</code> run. When used, the <code>-s</code> , <code>-i</code> , <code>-C</code> , and <code>-a</code> options are ignored. '-' may be used as the <code>snapshot_input</code> to specify <code>stdin</code> .
<code>-i/--interval seconds</code>	Obtains performance statistics over interval <code>seconds</code> . Clears all statistics, waits interval <code>seconds</code> , then generates report. Implies <code>-s</code> option.
<code>-b/--begin date_time</code>	Obtains past performance stats over an interval beginning at <code>date_time</code> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago.
<code>-e/--end date_time</code>	Obtains past performance stats over an interval ending at <code>date_time</code> . Implies <code>-s</code> option. <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago.
<code>-C/--clear</code>	Clears performance statistics for all ports. Only statistics with error thresholds are cleared. A clear occurs after generating the report.



<code>-a/--clearall</code>	Clears all performance statistics for all ports.
<code>-M/--pmadirect</code>	Accesses performance statistics using direct PMA.
<code>-A/--allports</code>	Gets PortInfo for down switch ports. Uses direct SMA to get this data. If used with <code>-M</code> , also gets PMA stats for down switch ports.
<code>-c/--config <i>file</i></code>	Specifies the error thresholds configuration file. Default is <code>/etc/opa/opamon.conf</code> file.
<code>-L/--limit</code>	Limits operation to exact specified focus with <code>-F</code> for port error counters check (<code>-o errors</code>) and port counters clear (<code>-C</code> or <code>-i</code>). Normally, the neighbor of each selected port is also checked/cleared. Does not affect other reports.
<code>-F/--focus <i>point</i></code>	Specifies the focus area for report. Used for all reports except <code>route</code> to limit scope of report. Refer to Point Syntax on page 177 for details.

-h and -p options permit a variety of selections:

- `-h 0` First active port in system (default).
- `-h 0 -p 0` First active port in system.
- `-h x` First active port on HFI *x*.
- `-h x -p 0` First active port on HFI *x*.
- `-h 0 -p y` Port *y* within system (no matter which ports are active).
- `-h x -p y` HFI *x*, port *y*.

The portion of the script that calls `opareport` and `opaxmlextract` follows:

```
opareport -x -d 10 -s -o errors -T $@ | opaxmlextract -d \;  
-e Rate -e MTU -e LinkDetails -e CableLength -e CableLabel  
-e CableDetails -e Port.NodeDesc -e Port.PortNum -e SymbolErrors.Value
```



Examples

```
opaextractstat topology_file
opaextractstat topology_file -c my_opamon.conf
```

5.4.5 opashowallports

(Switch and Host) Displays basic port state and statistics for all host nodes, chassis, or externally-managed switches.

Note: `opareport` and `opareports` are more powerful Intel® Omni-Path Fabric Suite FastFabric commands. For general fabric analysis, use `opareport` or `opareports` with options such as `-o errors` and `-o slowlinks` to perform an efficient analysis of link speeds and errors.

Syntax

```
opashowallports [-C] [-f hostfile] [-F chassisfile] [-h 'hosts']
                [-H 'chassis'] [-S]
```

Options

<code>--help</code>	Produces full help text.
<code>-C</code>	Performs operation against chassis. Default = host.
<code>-f hostfile</code>	Specifies the file containing the list of hosts in cluster. Default is <code>/etc/opa/hosts</code> file.
<code>-F chassisfile</code>	Specifies the file containing the list of chassis in cluster. Default is <code>/etc/opa/chassis</code> file.
<code>-h hosts</code>	Specifies the list of hosts for which to show ports.
<code>-H chassis</code>	Specifies the list of chassis for which to show ports.
<code>-S</code>	Securely prompts for password for admin on chassis.

Environment Variables

The following environment variables are also used by this command:

HOSTS	List of hosts, used if <code>-h</code> option not supplied. See discussion on Selection of Hosts .
CHASSIS	List of chassis, used if <code>-C</code> is used and <code>-H</code> and <code>-F</code> options not supplied. See discussion on Selection of Chassis .
HOSTS_FILE	File containing list of hosts, used in absence of <code>-f</code> and <code>-h</code> . See discussion on Selection of Hosts .



CHASSIS_FILE	File containing list of chassis, used in absence of <code>-F</code> and <code>-H</code> . See discussion on Selection of Chassis .
FF_CHASSIS_LOGIN_METHOD	How to log into chassis. Can be Telnet or SSH.
FF_CHASSIS_ADMIN_PASSWORD	Password for admin on all chassis. Used in absence of <code>-S</code> option.

Example

```
opashowallports
opashowallports -h 'elrond arwen'
HOSTS='elrond arwen' opashowallports
opashowallports -C
opashowallports -H 'chassis1 chassis2'
CHASSIS='chassis1 chassis2' opashowallports -C
```

Notes

When performing `opashowallports` against hosts, internally SSH is used. The command `opashowallports` requires that password-less SSH be set up between the host running the Intel® Omni-Path Fabric Suite FastFabric Toolset and the hosts `opashowallports` is operating against. The `opasetupssh` FastFabric tool can aid in setting up password-less SSH.

When performing operations against chassis, Intel recommends that you set up SSH keys (see [opasetupssh](#)). If SSH keys are not set up, Intel recommends that you use the `-S` option, to avoid keeping the password in configuration files.

When performing `opashowallports` against externally-managed switches, a node with Intel® Omni-Path Fabric Suite FastFabric Toolset installed is required. Typically, this is the node from which `opashowallports` is being run.

5.5 Configuration and Control for Chassis, Switch, and Host

The CLIs described in this section are used for general management of hosts in the fabric, as well as managed and externally-managed switches. There are also helper programs (for example, `opagenswitches`) to help produce the necessary configuration files.

5.5.1 opagenswitches

Analyzes the present fabric and produces a list of Externally Managed switches in the required format for the `/etc/opa/switches` file.

Syntax

```
opagenswitches [-t portsfile] [-p ports] [-R] [-L switches_file] [-o output_file]
[-T topology_file] [-X snapshot_file] [-s] [-v level] [-K]
```




Options

<code>--help</code>	Produces full help text.
<code>-t <i>portsfile</i></code>	Specifies the file with list of local HFI ports used to access fabric(s) for analysis. Default is <code>/etc/opa/ports</code> file.
<code>-p <i>ports</i></code>	Specifies the list of local HFI ports used to access fabrics for counter clear. Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format <code>hfi:port</code> , for example: <code>0:0</code> First active port in system. <code>0:y</code> Port <i>y</i> within system. <code>x:0</code> First active port on HFI <i>x</i> . <code>x:y</code> HFI <i>x</i> , port <i>y</i> .
<code>-R</code>	Does not attempt to get routes for computation of distance.
<code>-s</code>	Updates/resolves switch names using topology XML data.
<code>-L <i>switches_file</i></code>	Specifies the name of a pre-existing <i>switches_file</i> to be used as input in conjunction with a topology file. When specified, the file is used instead of switches data obtained from the actual fabric. The updated switches data is output to stdout (common to all <code>opagenswitches</code> operations). Does not generate switches data. Must also use <code>-s</code> option.
<code>-o <i>output_file</i></code>	Writes switches data to <i>output_file</i> . Default is stdout.
<code>-T <i>topology_file</i></code>	Specifies <i>topology_file</i> to use. May contain '%P'. Must also use <code>-s</code> . Link data in the topology file is compared to actual fabric link data (obtained by <code>opareport -o links</code> or <code>opareport -X snapshot -o links</code>). The data is also matched to a list of switch node GUIDs and the switch NodeDesc values are generated. This list is then applied to the switches data to update NodeDesc values. The comparison of topology link data to actual fabric link data starts with the host names. The host names in the actual fabric must match those in the topology file for the comparison to succeed. However, the comparison logic allows for some mismatches, which could be due to swapped or missing cables. Switch NodeDesc values are matched to GUIDs based on which switch has the greater number of matching links.



- `-X snapshot_file` Uses *snapshot_file* XML for fabric link information. May contain '%P'. Must also use `-s`.
- `-v level` Specifies the verbose level. Default = 0. Values include:
- 0 No output.
 - 1 Progress output.
 - 2 Reserved.
 - 4 Time stamps.
 - 8 Reserved.
- `-K` Does not clean temporary files. Temporary files are CSV format and contain lists of links used during script operation. The files are not normally needed after execution, but they can be retained for subsequent inspection or processing.

Environment Variables

The following environment variables are also used by this command:

- `PORTS` List of ports, used in absence of `-t` and `-p`.
- `PORTS_FILE` File containing list of ports, used in absence of `-t` and `-p`.
- `FF_TOPOLOGY_FILE` File containing topology XML data, used in absence of `-T`.

Examples

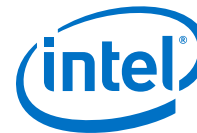
```
opagenswitches
opagenswitches -p '1:1 2:1'
opagenswitches -o switches
opagenswitches -s -o switches
opagenswitches -L switches -s -o switches
opagenswitches -s -T topology.%P.xml
opagenswitches -L switches -s -T topology.%P.xml -X snapshot.%P.xml
```

5.5.2 opagenchassis

Generates a list of IPv4, IPv6, and/or TCP names in a format acceptable for inclusion in the `/etc/opa/chassis` file.

Syntax

```
opagenchassis [-t portsfile] [-p ports]
```



Options

- `--help` Produces full help text.
- `-t portsfile` Specifies the file with list of local HFI ports used to access fabric for analysis. Default is `/etc/opa/ports` file.
- `-p ports` Specifies the list of local HFI ports used to access fabrics for counter clear.

Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format `hfi:port`, for example:

 - `0:0` First active port in system.
 - `0:y` Port *y* within system.
 - `x:0` First active port on HFI *x*.
 - `x:y` HFI *x*, port *y*.

Environment Variables

The following environment variables are also used by this command:

- `PORTS` List of ports, used in absence of `-t` and `-p`.
- `PORTS_FILE` File containing list of ports, used in absence of `-t` and `-p`.

Examples

```
opagenchassis
opagenchassis -p '1:1 1:2 2:1 2:2'
```

5.5.3 opagenesmchassis

Generates a list of chassis IPv4 and IPv6 addresses and/or TCP names where the Embedded Subnet Manager (ESM) is running, in a format acceptable for inclusion in the `/etc/opa/esm_chassis` file. This tool uses `opagenchassis` output to iterate through all the chassis.

Syntax

```
opagenesmchassis [-u user] [-S] [-t portsfile] [-p ports]
```

Options

- `--help` Produces full help text.
- `-u user` Performs command as *user*. For chassis, the default is `admin`.



- S Securely prompts for password for user on chassis.
- t *portsfile* Specifies the file with a list of local HFI ports used to access fabric(s) for analysis. Default is `/etc/opa/ports`
- p *ports* Specifies the list of local HFI ports used to access fabrics.
- Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format `hfi:port`, for example:
- 0:0 First active port in system.
- 0:y Port *y* within system.
- x:0 First active port on HFI *x*.
- x:y HFI *x*, port *y*.

Environment Variables

The following environment variables are also used by this command:

- FF_CHASSIS_ADMIN_PASSWORD Password for chassis, used in absence of -S.
- PORTS List of ports, used in absence of -t and -p.
- PORTS_FILE File containing list of ports, used in absence of -t and -p.

Examples

```
opagenesmchassis
opagenesmchassis -S -p '1:1 1:2 2:1 2:2'
```

Alternatively, while editing the file, use a `vi` command to include the output such as:

```
:r! opagenesmchassis
```

5.5.4 opachassisadmin

(Switch) Performs a number of multi-step chassis initialization and verification operations, including initial chassis setup, firmware upgrades, chassis reboot, and others.

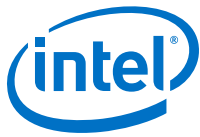
Syntax

```
opachassisadmin [-c] [-F chassisfile] [-H 'chassis'] [-P packages]
[-a action] [-I fm_bootstate] [-S] [-d upload_dir] [-s securityfiles]
operation ...
```



Options

<code>--help</code>	Produces full help text.	
<code>-c</code>	Overwrites the result files from any previous run before starting this run.	
<code>-F chassisfile</code>	Specifies the file with chassis in cluster. The default is <code>/etc/opa/chassis</code> .	
<code>-H chassis</code>	Specifies the list of chassis to execute the operation against.	
<code>-P packages</code>	Specifies the filenames and directories of firmware images to install. <ul style="list-style-type: none">For directories specified, all <code>.pkg</code>, <code>.dpkg</code>, and <code>.spkg</code> files in directory tree are used. <code>shell</code> wild cards may also be used within quotes.For <code>fmconfig</code>, filename of FM config file is used.For <code>fmgetconfig</code>, filename to upload to (default <code>opafm.xml</code>) is used.	
<code>-a action</code>	Specifies the action for the supplied file. The default is push.	
	For chassis upgrade:	
	push	Ensures firmware is in primary or alternate.
	select	Ensures firmware is in primary.
	run	Ensures firmware is in primary and running.
	For chassis fmconfig:	
	push	Ensures the configuration file is in chassis.
	run	After push, restarts FM on master, stops on secondary.
	runall	After push, restarts FM on all management modules.
	For chassis fmcontrol:	
	stop	Stops FM on all management modules.
	run	Ensures FM running on master, stopped on secondary.



	<code>runall</code>	Ensures FM running on all management modules.
	<code>restart</code>	Restarts FM on master, stops on secondary.
	<code>restartall</code>	Restarts FM on all MM.
For chassis		
<code>fmsecurityfiles:</code>	<code>push</code>	Ensures FM security files are in chassis.
	<code>restart</code>	After push, restarts FM on master, stop on slave.
	<code>restartall</code>	After push, restarts FM on all MM
<code>-I</code>	Specifies the <code>fmconfig</code> and <code>fmcontrol</code> install options.	
<code>fm_bootstate</code>	<code>disable</code>	Disables FM start at chassis boot.
	<code>enable</code>	Enables FM start on master at chassis boot.
	<code>enableall</code>	Enables FM start on all MM at chassis boot.
<code>-d</code>	<code>upload_dir</code>	Specifies the directory to upload FM configuration files to; default is <code>uploads</code> .
<code>-S</code>	Securely prompts for password for user on chassis.	
<code>-s</code>	Specifies the security files to install. Default is <code>*.pem</code> . For Chassis <code>fmsecurityfiles</code> , filenames/directories of security files to install. For directories specified, all security files in directory tree are used. Shell wildcards may also be used within quotes.	
<code>securityFiles</code>	For Chassis <code>fmgetsecurityfiles</code> , filename to upload to. Default is <code>*.pem</code>	
<code>operation</code>	Specifies the operation to perform. Can be one or more of:	
	<code>reboot</code>	Reboots chassis, ensures they go down and come back.
	<code>configure</code>	Runs wizard to perform chassis configuration.



upgrade	Upgrades install of all chassis.
getconfig	Gets basic configuration of chassis.
fmconfig	FM configuration operation on all chassis.
fmgetconfig	Fetches FM configuration from all chassis.
fmcontrol	Controls FM on all chassis.
fmsecurityfiles	FM security files operation on all chassis.
fmgetsecurityfiles	Fetches FM security files from all chassis.

For more information on the operations that can be performed, see [Operation Details](#) on page 232.

Example

```
opachassisadmin -c reboot
opachassisadmin -P /root/ChassisFw4.2.0.0.1 upgrade
opachassisadmin -H 'chassis1 chassis2' reboot
CHASSIS='chassis1 chassis2' opachassis_admin reboot
opachassisadmin -a run -P '*.pkg' upgrade
```

Environment Variables

The following environment variables are also used by this command:

CHASSIS	List of chassis, used if <code>-H</code> and <code>-F</code> option not supplied. Refer to Selection of Chassis on page 43 for more information.
CHASSIS_FILE	File containing list of chassis, used in absence of <code>-F</code> and <code>-H</code> . Refer to Selection of Chassis on page 43 for more information.
FF_MAX_PARALLEL	Maximum concurrent operations.
FF_SERIALIZE_OUTPUT	Serializes output of parallel operations (yes or no).
FF_TIMEOUT_MULT	Multiplier for all timeouts associated with this command. Used if the systems are slow for some reason.
UPLOADS_DIR	Directory to upload to, used in absence of <code>-d</code> .



Operation Details

(Switch) All chassis operations log into the chassis as chassis user admin. Intel recommends using the `-S` option to securely prompt for a password, in which case the same password is used for all chassis. Alternately, the password may be put in the environment or the `opafastfabric.conf` file using `FF_CHASSIS_ADMIN_PASSWORD`.

All versions of Intel® Omni-Path Switch 100 Series firmware permit SSH keys to be configured within the chassis for secure password-less login. In this case, there is no need to configure a `FF_CHASSIS_ADMIN_PASSWORD` and `FF_CHASSIS_LOGIN_METHOD` can be SSH. Refer to [Editing the Configuration Files for Chassis Setup](#) on page 68 for more information.

`upgrade` Upgrades the firmware on each chassis or slot specified. The `-P` option selects a directory containing `.pkg` files or provides an explicit list of `.pkg` files for the chassis and/or slots. The `-a` option selects the desired minimal state for the new firmware. For each chassis and/or slot selected for upgrade, the `.pkg` file applicable to that slot is selected and used. If more than one `.pkg` file is specified of a given card type, the operation is undefined.

The upgrade is intelligent and does not upgrade chassis that already have the desired firmware in the desired state (as specified by `-a`).

When the `-a` option specifies `run`, chassis that are not already running the desired firmware are rebooted. By selecting the proper `FF_MAX_PARALLEL` value, a rolling upgrade or a parallel upgrade may be accomplished. In most cases, a parallel upgrade is recommended for expediency.

For more information about chassis firmware, refer to the *Intel® Omni-Path Fabric Switches GUI User Guide* and *Intel® Omni-Path Fabric Switches Release Notes*.

`configure` Runs the chassis setup wizard, which asks a series of questions. Once the wizard has finished prompting for configuration information, all the selected chassis are configured through the CLI interface according to the responses. The following options may be configured for all chassis:

- Syslog server IP address, TCP/UDP port number, syslog facility code, and the chassis LogMode.
- NTP server
- Local time zone
- Link CRC Mode
- Link width supported
- Node description



<code>reboot</code>	Reboots the given chassis and ensures they go down and come back up by pinging them during the reboot process. By selecting the proper <code>FF_MAX_PARALLEL</code> value, a rolling reboot or a parallel reboot may be accomplished. In most cases, a parallel upgrade is recommended for expediency.
<code>getconfig</code>	Retrieves basic information from a chassis such as syslog, NTP configuration, timezone info, Link CRC Mode, Link Width, and node description.
<code>fmconfig</code>	Updates the Fabric Manager configuration file on each chassis specified. The <code>-P</code> option selects a file to transfer to the chassis. The <code>-a</code> option selects the desired minimal state for the new configuration and controls whether the FM is started/restarted after the file is updated. The <code>-I</code> option can be used to configure the FM start at boot for the selected chassis.
<code>fmgetconfig</code>	Uploads the FM configuration file from all selected chassis. The file is uploaded to the selected uploads directory. The <code>-P</code> option specifies the desired destination filename within the uploads directory.
<code>fmcontrol</code>	Allows the FM to be controlled on each chassis specified. The <code>-a</code> option selects the desired state for the FM. The <code>-I</code> option configures the FM start at boot for the selected chassis.
<code>fmsecurityfiles</code>	Updates the FM security files on each chassis specified. The <code>-s</code> option selects file(s) to transfer to the chassis. The <code>-a</code> option selects the desired minimal state for the new security files. In this release, <code>push</code> is the only supported action.
<code>fmgetsecurityfiles</code>	Uploads the FM security files from all selected chassis. The files are uploaded to the selected uploads directory. The <code>-s</code> option specifies the desired destination filename within the uploads directory.

Logging

`opachassisadmin` provides detailed logging of its results. During each run, the following files are produced:

<code>test.res</code>	This file is appended with summary results of run.
<code>test.log</code>	This file is appended with detailed results of run.
<code>save_tmp/</code>	This file contains a directory per failed test with detailed logs.
<code>test_tmp*/</code>	This file contains the intermediate results while the test is running.



The `-c` option removes all log files.

ssh Keys

When performing operations against chassis, Intel recommends setting up SSH keys. If SSH keys are not set up, all chassis must be configured with the same admin password. In this case, Intel recommends using the `-S` option. The `-S` option avoids the need to keep the password in configuration files.

Results

Results from `opachassisadmin` are grouped into test suites, test cases, and test items. A given run of `opachassisadmin` represents a single test suite. Within a test suite, multiple test cases occur; typically one test case per chassis being operated on. Some of the more complex operations may have multiple test items per test case. Each test item represents a major step in the overall test case.

Each `opachassisadmin` run appends to `test.res` and `test.log`, and creates temporary files in `test_tmp$PID` in the current directory. The `test.res` file provides an overall summary of operations performed and their results. The same information is also displayed while `opachassisadmin` is executing. `test.log` contains detailed information about what was performed, including the specific commands executed and the resulting output. The `test_tmp` directories contain temporary files that reflect tests in progress (or killed). The logs for any failures are logged in the `save_temp` directory with a directory per failed test case. If the same test case fails more than once, `save_temp` retains the information from the first failure. Subsequent runs of `opachassisadmin` are appended to `test.log`. Intel recommends reviewing failures and using the `-c` option to remove old logs before subsequent runs of `opachassisadmin`.

`opachassisadmin` implicitly performs its operations in parallel. However, as for the other tools, `FF_MAX_PARALLEL` can be exported to change the degree of parallelism. Twenty (20) parallel operations is the default.

5.5.5 opaswitchadmin

(Switch) Performs a number of multi-step initialization and verification operations against one or more externally managed Intel® Omni-Path switches. The operations include initial switch setup, firmware upgrades, chassis reboot, and others.

Syntax

```
opaswitchadmin [-c] [-N 'nodes'] [-L nodefile] [-O]
               [-P packages] [-a action] [-t portfile]
               [-p ports] operation ...
```

Options

- | | |
|--------------------|---|
| <code>-help</code> | Produces full help text. |
| <code>-c</code> | Overwrites result files from any previous run before starting this run. |



<code>-N nodes</code>	Specifies the list of nodes to execute the operation against.
<code>-L nodefile</code>	Specifies the file with nodes in the cluster. Default is <code>/etc/opa/switches</code> file.
<code>-P packages</code>	For upgrades: Specifies the file name or directory where the firmware image is to install. For the directory specified, <code>.emfw</code> file in the directory tree is used. <code>shell</code> wild cards may also be used within quotes.
<code>-t portsfile</code>	Specifies the file with list of local HFI ports used to access fabrics for switch access. Default is <code>/etc/opa/ports</code> file.
<code>-p ports</code>	<p>Specifies the list of local HFI ports used to access fabrics for switch access.</p> <p>Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format <code>hfi:port</code>, for example:</p> <p><code>0:0</code> First active port in system.</p> <p><code>0:y</code> Port <code>y</code> within system.</p> <p><code>x:0</code> First active port on HFI <code>x</code>.</p> <p><code>x:y</code> HFI <code>x</code>, port <code>y</code>.</p>
<code>-a action</code>	<p>Specifies an action for firmware file for switch upgrade. The <code>action</code> argument can be one or more of the following:</p> <p><code>select</code> Ensures firmware is in primary (default).</p> <p><code>run</code> Ensures firmware is in primary and running.</p>
<code>-O</code>	Specifies the override for firmware upgrades, bypasses the previous firmware version checks, and forces the update unconditionally.
<code>operation</code>	<p>Performs the specified <code>operation</code>, which can be one or more of the following:</p> <p><code>reboot</code> Reboots switches, ensures they go down and come back.</p> <p><code>configure</code> Runs wizard to set up switch configuration.</p> <p><code>upgrade</code> Upgrades installation of all switches.</p> <p><code>info</code> Reports firmware and hardware version, part number, and data rate capability of all nodes.</p>



hwvpd	Completes hardware Vital Product Data (VPD) report of all nodes.
ping	Pings all nodes and tests for presence.
fwverify	Reports integrity of failsafe firmware of all nodes.
getconfig	Gets port configurations of an externally managed switch.

For more information on operations, see [Operation Details](#) on page 237.

Example

```
opaswitchadmin -c reboot
opaswitchadmin -P /root/ChassisFwX.X.X.X.X upgrade
opaswitchadmin -a run -P '*.emfw' upgrade
```

Environment Variables

The following environment variables are also used by this command:

OPASWITCHES	List of nodes, used in absence of <code>-N</code> and <code>-L</code> options. See discussion in Selection of Switches on page 45.
OPASWITCHES_FILE	File containing list of nodes, used in absence of <code>-N</code> and <code>-L</code> options. See discussion in Selection of Switches on page 45.
FF_MAX_PARALLEL	Maximum concurrent operations.
FF_SERIALIZE_OUTPUT	Serialize output of parallel operations (yes or no).
FF_TIMEOUT_MULT	Multiplier for all timeouts associated with this command. Used if the systems are slow for some reason.

Details

`opaswitchadmin` provides detailed logging of its results. During each run, the following files are produced:

- `test.res`: Appended with summary results of run.
- `test.log`: Appended with detailed results of run.
- `save_tmp/`: Contains a directory per failed test with detailed logs.
- `test_tmp*/`: Intermediate result files while test is running.

The `-c` option removes all log files.



Results from `opaswitchadmin` are grouped into test suites, test cases, and test items. A given run of `opaswitchadmin` represents a single test suite. Within a test suite, multiple test cases occur; typically one test case per chassis being operated on. Some of the more complex operations may have multiple test items per test case. Each test item represents a major step in the overall test case.

Each `opaswitchadmin` run appends to `test.res` and `test.log` and creates temporary files in `test_tmp$PID` in the current directory. The `test.res` file provides an overall summary of operations performed and their results. The same information is also displayed while `opaswitchadmin` is executing. `test.log` contains detailed information about what was performed, including the specific commands executed and the resulting output. The `test_tmp` directories contain temporary files that reflect tests in progress (or killed). The logs for any failures are logged in the `save_temp` directory with a directory per failed test case. If the same test case fails more than once, `save_temp` retains the information from the first failure. Subsequent runs of `opaswitchadmin` are appended to `test.log`. Intel recommends reviewing failures and using the `-c` option to remove old logs before subsequent runs of `opaswitchadmin`. `opaswitchadmin` also appends to `punchlist.csv` for failing switches.

`opaswitchadmin` implicitly performs its operations in parallel. However, as for the other tools, `FF_MAX_PARALLEL` can be exported to change the degree of parallelism. Twenty (20) parallel operations is the default.

Operation Details

(Switch) All operations against Intel® Omni-Path Fabric externally-managed switches (except ping) securely access the selected switches. If a password has been set, the `-S` option must be used to securely prompt for a password. In this case, the same password is used for all switches.

`reboot` Reboots the given switches.

Use the `FF_MAX_PARALLEL` value to select either a rolling reboot or a parallel reboot. In most cases, a parallel reboot is recommended for expediency.

`upgrade` Upgrades the firmware on each specified switch. The `-P` option selects a directory containing a `.emfw` file or provides an explicit `.emfw` file for the switches. If more than one `.emfw` file is specified, the operation is undefined. The `-a` option selects the desired minimal state for the new firmware. Only the `select` and `run` options are valid for this operation.

When the `-a` option specifies `run`, switches are rebooted. Use the `FF_MAX_PARALLEL` value to select a rolling upgrade or a parallel upgrade. In most cases, a parallel upgrade is recommended for expediency.

The upgrade process also sets the switch name. See discussion on [Selection of Devices](#) on page 41.

The upgrade process is used to set, clear, or change the password of the switches using the `-s` option. When this option is specified, you are prompted for a new password to be set on the switches. To reset (clear)



the password, press **Enter** when prompted. This option can be used to configure the switches to not require a password for subsequent operations. A change to the password does not take effect until the next reboot of the switch.

For more information about switch firmware, refer to the *Intel® Omni-Path Fabric Switches GUI User Guide* and *Intel® Omni-Path Fabric Switches Release Notes*.

`configure` Runs the switch setup wizard, which asks a series of questions. Once the wizard has finished prompting for configuration information, all the selected switches are configured according to the entered responses. The following items are configurable for all Intel® Omni-Path Switch 100 Series:

- FM Enabled
- Link CRC Mode
- Link Width Supported
- OPA Node Description

Note: If 4X capability is not enabled in the user selection, 4X capability is added to port 1 for each switch being configured. This provides a *rescue* capability for the switch using FastFabric, in case the link is unable to connect to a link width other than 4X.

Note: Typically, the Node Description is updated automatically during a firmware upgrade, if it is configured properly in the `switches` file. Updating the node description is also available using the `configure` option without performing a firmware upgrade.

`info` Queries the switches and displays the following information:

- Firmware version
- Hardware version
- Hardware part number, including revision information
- Speed capability
- Fan status
- Power supply status

This operation also outputs a summary of various configuration settings for each switch within a fabric.

For example, in a fabric with seven switches, a report similar to the following is displayed.

```
Summary:
count - info
7 - Capability:QDR
7 - Fan 1 status:Normal/Normal
7 - Fan 2 status:Normal/Normal
6 - F/W ver:6.0.2.0.28
1 - F/W ver:6.1.0.0.72
```



```
7 - H/W pt num:220058-004-E
7 - H/W ver:004-E
7 - PS1 Status:N/A
7 - PS2 Status:ENGAGED
```

- hwvpd** Queries the switches and displays the Vital Product Data (VPD) including:
- Serial number
 - Part number
 - Model name
 - Hardware version
 - Manufacturer
 - Product description
 - Manufacturer ID
 - Manufacture date
 - Manufacture time
- ping** Issues an inband packet to the switches to test for presence and reports on presence/non-presence of each selected switch.
- Note:* It is not necessary to supply a password (using `-s`) for this operation.
- fwverify** Verifies the integrity of the firmware images in the EEPROMs of the selected switches.
- getconfig** Gets port configurations of an externally managed switch. This operation also outputs a summary of various configuration settings for each switch within a fabric. For example, in a fabric with seven switches, a report similar to the following is displayed.

```
Summary:
count - configuration
7 - Link Speed : 2.5-10Gb
1 - Link Width : 1-8x
6 - Link Width : 4x
```

This summary helps determine if all switches have the same configuration, and if not, indicates how many have each value. If some of the values are not as expected, view the `test.res` file to identify which switches have the undesirable values.

5.5.6 opahostadmin

(Host) Performs a number of multi-step host initialization and verification operations, including upgrading software or firmware, rebooting hosts, and other operations. In general, operations performed by `opahostadmin` involve a login to one or more host systems.



Syntax

```
opahostadmin [-c] [-i ipoib_suffix] [-f hostfile] [-h 'hosts']  
[-r release] [-I install_options] [-U upgrade_options] [-d dir]  
[-T product] [-P packages] [-m netmask] [-S] operation ...
```

Options

<code>--help</code>	Produces full help text.		
<code>-c</code>	Overwrites the result files from any previous run before starting this run.		
<code>-i <i>ipoib_suffix</i></code>	Specifies the suffix to apply to host names to create IPoIB host names. Default is <code>-opa</code> .		
<code>-f <i>hostfile</i></code>	Specifies the file with the names of hosts in a cluster. Default is <code>/etc/opa/hosts</code> file.		
<code>-h <i>hosts</i></code>	Specifies the list of hosts to execute the operation against.		
<code>-r <i>release</i></code>	Specifies the software version to load/upgrade to. Default is the version of Intel® Omni-Path Software presently being run on the server.		
<code>-d <i>dir</i></code>	Specifies the directory to retrieve <code>product.release.tgz</code> for load or upgrade.		
<code>-I <i>install_options</i></code>	Specifies the software install options.		
<code>-U <i>upgrade_options</i></code>	Specifies the software upgrade options.		
<code>-T <i>product</i></code>	Specifies the product type to install. Default = <code>IntelOPA-Basic.RHEL72-x86_64</code> Other options include: <code>IntelOPA-Basic.<distro></code> , <code>IntelOPA-IFS.<distro></code> where <code><distro></code> is the distribution and CPU.		
<code>-P <i>packages</i></code>	Specifies the packages to install. Default = <code>oftools ipoib psm_mpi</code>		
<code>-m <i>netmask</i></code>	Specifies the IPoIB netmask to use for <code>configipoib</code> operation.		
<code>-S</code>	Securely prompts for user password on remote system.		
<code><i>operation</i></code>	Performs the specified <i>operation</i> , which can be one or more of the following: <table><tr><td><code>load</code></td><td>Starts initial installation of all hosts.</td></tr></table>	<code>load</code>	Starts initial installation of all hosts.
<code>load</code>	Starts initial installation of all hosts.		



<code>upgrade</code>	Upgrades installation of all hosts.
<code>configipoib</code>	Creates <code>ifcfg-ib1</code> using host IP address from <code>/etc/hosts</code> file.
<code>reboot</code>	Reboots hosts, ensures they go down and come back.
<code>sacache</code>	Confirms <code>sacache</code> has all hosts in it.
<code>ipoibping</code>	Verifies this host can ping each host through IPoIB.
<code>mpiperf</code>	Verifies latency and bandwidth for each host.
<code>mpiperfdeviation</code>	Verifies latency and bandwidth for each host against a defined threshold (or relative to average host performance).

Example

```
opahostadmin -c reboot
opahostadmin upgrade
opahostadmin -h 'elrond arwen' reboot
HOSTS='elrond arwen' opahostadmin reboot
```

Details

`opahostadmin` provides detailed logging of its results. During each run, the following files are produced:

- `test.res`: Appended with summary results of run.
- `test.log`: Appended with detailed results of run.
- `save_tmp/`: Contains a directory per failed test with detailed logs.
- `test_tmp*/`: Intermediate result files while test is running.

The `-c` option removes all log files.

Results from `opahostadmin` are grouped into test suites, test cases, and test items. A given run of `opahostadmin` represents a single test suite. Within a test suite, multiple test cases occur; typically one test case per host being operated on. Some of the more complex operations may have multiple test items per test case. Each test item represents a major step in the overall test case.

Each `opahostadmin` run appends to `test.res` and `test.log`, and creates temporary files in `test_tmp$PID` in the current directory. `test.res` provides an overall summary of operations performed and their results. The same information is also displayed while `opahostadmin` is executing. `test.log` contains detailed information about what was performed, including the specific commands executed and



the resulting output. The `test_tmp` directories contain temporary files which reflect tests in progress (or killed). The logs for any failures are logged in the `save_temp` directory with a directory per failed test case. If the same test case fails more than once, `save_temp` retains the information from the first failure. Subsequent runs of `opahostadmin` are appended to `test.log`. Intel recommends reviewing failures and using the `-c` option to remove old logs before subsequent runs of `opahostadmin`.

`opahostadmin` implicitly performs its operations in parallel. However, as for the other tools, `FF_MAX_PARALLEL` can be exported to change the degree of parallelism. Twenty (20) parallel operations is the default.

Environment Variables

The following environment variables are also used by this command:

<code>HOSTS</code>	List of hosts, used if <code>-h</code> option not supplied.
<code>HOSTS_FILE</code>	File containing list of hosts, used in absence of <code>-f</code> and <code>-h</code> .
<code>FF_MAX_PARALLEL</code>	Maximum concurrent operations are performed.
<code>FF_SERIALIZE_OUTPUT</code>	Serialize output of parallel operations (yes or no).
<code>FF_TIMEOUT_MULT</code>	Multiplier for all timeouts associated with this command. Used if the systems are slow for some reason.

opahostadmin Operation Details

(Host) Intel recommends that you set up password SSH or SCP for use during this operation. Alternatively, the `-S` option can be used to securely prompt for a password, in which case the same password is used for all hosts. Alternately, the password may be put in the environment or the `opafastfabric.conf` file using `FF_PASSWORD` and `FF_ROOTPASS`.

<code>load</code>	Performs an initial installation of Intel® Omni-Path Software on a group of hosts. Any existing installation is uninstalled and existing configuration files are removed. Subsequently, the hosts are installed with a default Intel® Omni-Path Software configuration. The <code>-I</code> option can be used to select different install packages. Default = <code>oftools ipoib mpi</code> The <code>-r</code> option can be used to specify a release to install other than the one that this host is presently running. The <code>FF_PRODUCT.FF_PRODUCT_VERSION.tgz</code> file (for example, <code>IntelOPA-Basic.version.tgz</code>) is expected to exist in the directory specified by <code>-d</code> . Default is the current working directory. The specified software is copied to all the selected hosts and installed.
<code>upgrade</code>	Upgrades all selected hosts without modifying existing configurations. This operation is comparable to the <code>-U</code> option when running <code>./INSTALL</code> manually. The <code>-r</code> option can be used to upgrade to a release different from this host. The



default is to upgrade to the same release as this host. The `FF_PRODUCT_FF_PRODUCT_VERSION.tgz` file (for example, `IntelOPA-Basic.version.tgz`) is expected to exist in the directory specified by `-d`. (The default is the current working directory.) The specified software is copied to all the end nodes and installed.

Note: Only components that are currently installed are upgraded. This operation fails for hosts that do not have Intel® Omni-Path Software installed.

<code>configipoib</code>	<p>Creates a <code>ifcfg-ib1</code> configuration file for each node using the IP address found using the resolver on the node. The standard Linux* resolver is used through the <code>host</code> command. (If running OFA Delta, this option configures <code>ifcfg-ib0</code>.)</p> <p>If the host is not found, <code>/etc/hosts</code> on the node is checked. The <code>-i</code> option specifies an IPoIB suffix to apply to the host name to create the IPoIB host name for the node. The default suffix is <code>-ib</code>. The <code>-m</code> option specifies a netmask other than the default for the given class of IP address, such as when dividing a class A or B address into smaller IP subnets. IPoIB is configured for a static IP address and is autostarted at boot. For the Intel® OP Software Stack, the default <code>/etc/ipoib.cfg</code> file is used, which provides a redundant IPoIB configuration using both ports of the first HFI in the system.</p> <p>Note: <code>opahostadmin configipoib</code> now supports DHCP (auto or static options) for configuring the IPoIB interface. You must specify these options in <code>/etc/opa/opafastfabric.conf</code> against the <code>FF_IPOIB_CONFIG</code> variable. If no options are found, the static IP configuration is used by default. If <code>auto</code> is specified, then one IP address from either <code>static</code> or <code>dhcp</code> is chosen. Static is used if the IP address can be obtained out of <code>/etc/hosts</code> or the resolver, otherwise DHCP is used.</p>
<code>reboot</code>	<p>Reboots the given hosts and ensures they go down and come back up by pinging them during the reboot process. The ping rate is slow (5 seconds), so if the servers boot faster than this, false failures may be seen.</p>
<code>sacache</code>	<p>Verifies the given hosts can properly communicate with the SA and any cached SA data that is up to date. To run this command, Intel® Omni-Path Fabric software must be installed and running on the given hosts. The subnet manager and switches must be up. If this test fails: <code>opacmdall 'opasaquery -o desc'</code> can be run against any problem hosts.</p>



Note: This operation requires that the hosts being queried are specified by a resolvable TCP/IP host name. This operation FAILS if the selected hosts are specified by IP address.

`ipoibping`

Verifies IPoIB basic operation by ensuring that the host can ping all other nodes through IPoIB. To run this command, Intel® Omni-Path Fabric software must be installed, IPoIB must be configured and running on the host, and the given hosts, the SM, and switches must be up. The `-i` option can specify an alternate IPoIB hostname suffix.

`mpiperf`

Verifies that MPI is operational and checks MPI end-to-end latency and bandwidth between pairs of nodes (for example, 1-2, 3-4, 5-6). Use this to verify switch latency/hops, PCI bandwidth, and overall MPI performance. The `test.res` file contains the results of each pair of nodes tested.

Note: This option is available for the Intel® Omni-Path Fabric Host Software OFA Delta packaging, but is not presently available for other packagings of OFED.

To obtain accurate results, this test should be run at a time when no other stressful applications (for example, MPI jobs or high stress file system operations) are running on the given hosts.

Bandwidth issues typically indicate server configuration issues (for example, incorrect slot used, incorrect BIOS settings, or incorrect HFI model), or fabric issues (for example, symbol errors, incorrect link width, or speed). Assuming `opareport` has previously been used to check for link errors and link speed issues, the server configuration should be verified.

Note that BIOS settings and differences between server models can account for 10-20% differences in bandwidth. For more details about BIOS settings, consult the documentation from the server supplier and/or the server PCI chipset manufacturer.

`mpiperfdeviation`

Specifies the enhanced version of `mpiperf` that verifies MPI performance. Can be used to verify switch latency/hops, PCI bandwidth, and overall MPI performance. It performs assorted pair-wise bandwidth and latency tests, and reports pairs outside an acceptable tolerance range. The tool identifies specific nodes that have problems and provides a concise summary of results. The `test.res` file contains the results of each pair of nodes tested.

By default, concurrent mode is used to quickly analyze the fabric and host performance. Pairs that have 20% less bandwidth or 50% more latency than the average pair are reported as failures.



The tool can be run in a sequential or a concurrent mode. Sequential mode runs each host against a reference host. By default, the reference host is selected based on the best performance from a quick test of the first 40 hosts. In concurrent mode, hosts are paired up and all pairs are run concurrently. Since there may be fabric contention during such a run, any poor performing pairs are then rerun sequentially against the reference host.

Concurrent mode runs the tests in the shortest amount of time, however, the results could be slightly less accurate due to switch contention. In heavily oversubscribed fabric designs, if concurrent mode is producing unexpectedly low performance, try sequential mode.

Note: This option is available for the Intel® Omni-Path Fabric Host Software OFA Delta packaging, but is not presently available for other packagings of OFED.

To obtain accurate results, this test should be run at a time when no other stressful applications (for example, MPI jobs, high stress file system operations) are running on the given hosts.

Bandwidth issues typically indicate server configuration issues (for example, incorrect slot used, incorrect BIOS settings, or incorrect HFI model), or fabric issues (for example, symbol errors, incorrect link width, or speed). Assuming `opareport` has previously been used to check for link errors and link speed issues, the server configuration should be verified.

Note that BIOS settings and differences between server models can account for 10-20% differences in bandwidth. A result 5-10% below the average is typically not cause for serious alarm, but may reflect limitations in the server design or the chosen BIOS settings.

For more details about BIOS settings, consult the documentation from the server supplier and/or the server PCI chipset manufacturer.

The deviation application supports a number of parameters which allow for more precise control over the mode, benchmark and pass/fail criteria. The parameters to use can be selected using the `FF_DEVIATION_ARGS` configuration parameter in `opafastfabric.conf`

Available parameters for deviation application:

```
[-bwtol bwtol] [-bwdelta MBs] [-bwthres MBs]
[-bwloop count] [-bwsize size] [-lattol latol]
[-latdelta usec] [-latthres usec] [-latloop count]
[-latsize size] [-c] [-b] [-v] [-vv]
[-h reference_host]
```



<code>-bwtol</code>	Specifies the percent of bandwidth degradation allowed below average value.
<code>-bwbidir</code>	Performs a bidirectional bandwidth test.
<code>-bwunidir</code>	Performs a unidirectional bandwidth test (default).
<code>-bwdelta</code>	Specifies the limit in MB/s of bandwidth degradation allowed below average value.
<code>-bwthres</code>	Specifies the lower limit in MB/s of bandwidth allowed.
<code>-bwloop</code>	Specifies the number of loops to execute each bandwidth test.
<code>-bwsiz</code>	Specifies the size of message to use for bandwidth test.
<code>-lattol</code>	Specifies the percent of latency degradation allowed above average value.
<code>-latdelta</code>	Specifies the limit in μ sec of latency degradation allowed above average value.
<code>-latthres</code>	Specifies the lower limit in μ sec of latency allowed.
<code>-latloop</code>	Specifies the number of loops to execute each latency test.
<code>-latsiz</code>	Specifies the size of message to use for latency test.
<code>-c</code>	Runs test pairs concurrently instead of the default of sequential.
<code>-b</code>	When comparing results against tolerance and delta, uses best instead of average.
<code>-v</code>	Specifies the verbose output.
<code>-vv</code>	Specifies the very verbose output.
<code>-h</code>	Specifies the reference host to use for sequential pairing.

Both `bwtol` and `bwdelta` must be exceeded to fail bandwidth test.



When `bwthres` is supplied, `bwtol` and `bwdelta` are ignored.

Both `lattol` and `latdelta` must be exceeded to fail latency test.

When `latthres` is supplied, `lattol` and `latdelta` are ignored.

For consistency with OSU benchmarks, MB/s is defined as 1000000 bytes/s.

5.5.7 Interpreting the `opahostadmin`, `opachassisadmin`, and `opaswitchadmin` log files

Each run of `opahostadmin`, `opachassisadmin`, and `opaswitchadmin` creates `test.log` and `test.res` files in the current directory.

The `test.res` file summarizes which tests have failed and identifies servers that have failed. If the problem is not immediately obvious, check the `test.log` file. The most recent results are at the end of the file. The `save_tmp/*/test.log` files are easier to read since they represent the logs for a single test case, typically against a single chassis, switch, or host.

The keyword `FAILURE` is used to mark any failures. Due to the roll up of error messages, the first instance of `FAILURE` in a given sequence shows the operations in process at the time of failure. The log also shows the exact sequence of commands issued to the target host and/or chassis and the resulting output from that host and/or chassis before the `FAILURE` keyword.

If there is a `FAILURE` message indicating time-out, it means the expected output did not occur within a reasonable time limit. The time limits used are generous, so such failures often indicate a host, chassis, or switch is offline. It could also indicate unexpected prompts, such as a password prompt when password-less SSH is expected. Review the `test.log` first for such prompts. Also verify that the host can SSH to the target host or chassis with the expected password behavior.

One common source of time-out errors is incorrect host shell command prompts. Verify that both this host and the target host meet the following criteria for command prompts:

- The command line prompt must end in `#` or `$`
- There must be a space after either character.

Another common source of time-outs is typographical errors in selected host or chassis names. Verify that the host, chassis, or switch names in the `test.log` file match the intended host names.

When IPoIB host names are used, verify that the correct name is formed based on the `opahostadmin -i '<IPOIB SUFFIX>'` argument. This argument applies a suffix to host names to create IPoIB host names. The default is `-ib`. Use `-i ''` to indicate no suffix.



5.6 Basic Setup and Administration Tools

The tools described in this section are available on a node that has Intel® Omni-Path Fabric Suite installed.

5.6.1 opapingall

(All) Pings a group of hosts or chassis to verify that they are powered on and accessible through TCP/IP ping.

Syntax

```
opapingall [-C] [-p] [-f hostfile] [-F chassisfile] [-h 'hosts'] [-H 'chassis']
```

Options

--help	Produces full help text.
-C	Performs a ping against a chassis. The default is hosts.
-p	Pings all hosts/chassis in parallel.
-f <i>hostfile</i>	Specifies the file with hosts in cluster. Default is /etc/opa/hosts.
-F <i>chassisfile</i>	Specifies the file with chassis in cluster. Default is /etc/opa/chassis.
-h <i>hosts</i>	Specifies the list of hosts to ping.
-H <i>chassis</i>	Specifies the list of chassis to ping.

Example

```
opapingall
opapingall -h 'arwen elrond'
HOSTS='arwen elrond' opapingall
opapingall -C
```

Note: This command pings all hosts/chassis found in the specified host/chassis file. The use of -C option merely selects the default file and/or environment variable to use. For this command, it is valid to use a file that lists both hosts and chassis.

```
opapingall -C -H 'chassis1 chassis2'
CHASSIS='chassis1 chassis2' opapingall -C
```

Environment Variables

HOSTS	List of hosts, used if -h option not supplied.
CHASSIS	List of chassis, used if -H option not supplied.



HOSTS_FILE	File containing list of hosts, used in absence of <code>-f</code> and <code>-h</code> .
CHASSIS_FILE	File containing list of chassis, used in absence of <code>-F</code> and <code>-H</code> .
FF_MAX_PARALLEL	When <code>-p</code> option is used, maximum concurrent operations are performed.

5.6.2 opasetupssh

(Linux or Switch) Creates SSH keys and configures them on all hosts or chassis so the system can use SSH and SCP into all other hosts or chassis without a password prompt. Typically, during cluster setup this tool enables the root user on the Management Node to log into the other hosts (as root) or chassis (as admin) using password-less SSH.

Syntax

```
opasetupssh [-C|p|U] [-f hostfile] [-F chassisfile]
[-h 'hosts'] [-H 'chassis'] [-i ipoib_suffix]
[-u user] [-S] [-R|P]
```

Options

<code>--help</code>	Produces full help text.
<code>-C</code>	Performs operation against chassis. Default is hosts.
<code>-p</code>	Performs operation against all chassis or hosts in parallel.
<code>-U</code>	Performs connect only (to enter in local hosts, known hosts). When run in this mode, the <code>-S</code> option is ignored.
<code>-f <i>hostfile</i></code>	Specifies the file with hosts in cluster. Default is <code>/etc/opa/hosts file</code> .
<code>-F <i>chassisfile</i></code>	Specifies the file with chassis in cluster. Default is <code>/etc/opa/chassis file</code> .
<code>-h <i>hosts</i></code>	Specifies the list of hosts to set up.
<code>-H <i>chassis</i></code>	Specifies the list of chassis to set up.
<code>-i <i>ipoib_suffix</i></code>	Specifies the suffix to apply to host names to create IPoIB host names. Default is <code>-opa</code> .
<code>-u <i>user</i></code>	Specifies the user on remote system to allow this user to SSH to. Default is current user code for host(s) and admin for chassis.
<code>-S</code>	Securely prompts for password for user on remote system.



- R Skips setup of SSH to local host.
- P Skips ping of host (for SSH to devices on Internet with ping firewalled).

Examples

Operations on Hosts

```
opasetupssh -S -i ''  
opasetupssh -U  
opasetupssh -h 'arwen elrond' -U  
HOSTS='arwen elrond' opasetupssh -U
```

Operations on Chassis

```
opasetupssh -C  
opasetupssh -C -H 'chassis1 chassis2'  
CHASSIS='chassis1 chassis2' opasetupssh -C
```

Environment Variables

The following environment variables are also used by this command:

HOSTS_FILE	File containing list of hosts, used in absence of -f and -h. See discussion on Selection of Hosts .
CHASSIS_FILE	File containing list of chassis, used in absence of -F and -H. See discussion on Selection of Chassis .
HOSTS	List of hosts, used if -h option not supplied. See discussion on Selection of Hosts .
CHASSIS	List of chassis, used if -C is used and -H and -F options not supplied. See discussion on Selection of Chassis .
FF_MAX_PARALLEL	When -p option is used, maximum concurrent operations.
FF_IPOIB_SUFFIX	Suffix to append to hostname to create IPoIB hostname. Used in absence of -i.
FF_CHASSIS_LOGIN_METHOD	How to log into chassis. Can be Telnet or SSH.
FF_CHASSIS_ADMIN_PASSWORD	Password for admin on all chassis. Used in absence of -S option.



Description

The Intel® Omni-Path Fabric Suite FastFabric Toolset provides additional flexibility in the translation between IPoIB and management network hostnames. Refer to [Configuration of IPoIB Name Mapping](#) on page 49 for more information.

`opasetupssh` provides an easy way to create SSH keys and distribute them to the hosts or chassis in the cluster. Many of the FastFabric tools (as well as many versions of MPI) require that SSH is set up for password-less operation. Therefore, `opasetupssh` is an important setup step.

This tool also sets up SSH to the local host and the local host's IPoIB name. This capability is required by selected FastFabric Toolset commands and may be used by some applications (such as MPI).

`opasetupssh` has two modes of operation. The mode is selected by the presence or absence of the `-U` option. Typically, `opasetupssh` is first run without the `-U` option, then it may later be run with the `-U` option.

Host Initial Key Exchange

When run without the `-U` option, `opasetupssh` performs the initial key exchange and enables password-less SSH and SCP. The preferred way to use `opasetupssh` for initial key exchange is with the `-S` option. This requires that all hosts are configured with the same password for the specified "user" (typically root). In this mode, the password is prompted for once and then SSH and SCP are used in conjunction with that password to complete the setup for the hosts. This mode also avoids the need to set up `rsh/rcp/rlogin` (which can be a security risk).

`opasetupssh` configures password-less SSH/SCP for both the management network and IPoIB. Typically, the management network is used for FastFabric Toolset operations while IPoIB is used for MPI and other applications.

During initial cluster installation, where the Intel® Omni-Path Fabric software is not yet installed on all the hosts, IPoIB is not yet running. In this situation, use the `-i` option with an empty string as follows:

```
opasetupssh -i ''
```

This causes the last part of the setup of SSH for IPoIB to be skipped.

Refreshing Local Systems Known Hosts

If aspects of the host have changed, such as IP addresses, MAC addresses, software installation, or server OS reinstallation, you can refresh the local host's SSH `known_hosts` file by running `opasetupssh` with the `-U` option. This option does not transfer the keys, but instead connects to each host (management network and IPoIB) to refresh the SSH keys. Existing entries for the specified hosts are replaced within the local `known_hosts` file. When run in this mode, the `-S` option is ignored. This mode assumes SSH has previously been set up for the hosts, as such no files are transferred to the specified hosts and no passwords should be required.

Typically after completing the installation and booting of Intel® Omni-Path Fabric software, `opasetupssh` must be rerun with the `-U` option to update the `known_hosts` file.



Chassis Initial Key Exchange

When run without the `-U` option, `opasetupssh` performs the initial key exchange and enables password-less SSH and SCP. For chassis, the key exchange uses SCP and the chassis CLI. During this command you log into the chassis using the configured mechanism for chassis login.

The preferred way to use `opasetupssh` for initial key exchange is with the `-S` option. This requires that all chassis are configured with the same password for admin. In this mode, you are prompted for the password once and then the `FF_CHASSIS_LOGIN_METHOD` and SCP are used in conjunction with that password to complete the setup for the chassis. This method also avoids the need to setup the chassis password in `/etc/opa/opafastfabric.conf` (which can be a security risk).

For chassis, the `-i` option is ignored.

Chassis Refreshing Local Systems Known Hosts

If aspects of the chassis have changed, such as IP addresses or MAC addresses, you can refresh the local host's SSH `known_hosts` file by running `opasetupssh` with the `-U` option. This option does not transfer the keys, but instead connects to each chassis to refresh the SSH keys. Existing entries for the specified chassis are replaced within the local `known_hosts` file. When run in this mode, the `-S` option is ignored. This mode assumes SSH has previously been set up for the chassis, because no files are transferred to the specified hosts and no passwords are required.

5.6.3 opacmdall

(Linux and Switch) Executes a command on all hosts or Intel® Omni-Path Chassis. This powerful command can be used for configuring servers or chassis, verifying that they are running, starting and stopping host processes, and other tasks.

Note: `opacmdall` depends on the Linux* convention that utilities return 0 for success and >0 for failure. If `opacmdall` is used to execute a non-standard utility like `diff` or a program that uses custom exit codes, then `opacmdall` may erroneously report "Command execution FAILED" when it encounters a non-zero exit code. However, command output is still returned normally and the error may be safely ignored.

Syntax

```
opacmdall [-CpqPS] [-f hostfile] [-F chassisfile]
[-h hosts] [-H chassis] [-u user]
[-m marker] [-T timelimit] cmd
```

Options

<code>--help</code>	Produces full help text.
<code>-C</code>	Performs command against chassis. Default is hosts.
<code>-p</code>	Runs command in parallel on all hosts/chassis.



<code>-q</code>	Quiet mode, do not show command to execute.
<code>-P</code>	Outputs the hostname/chassis name as prefix to each output line. This can make script processing of output easier.
<code>-S</code>	Securely prompts for password for user on chassis.
<code>-f <i>hostfile</i></code>	Specifies the file with hosts in cluster. Default is <code>/etc/opa/hosts file</code> .
<code>-F <i>chassisfile</i></code>	Specifies the file with chassis in cluster. Default is <code>/etc/opa/chassis file</code> .
<code>-h <i>host</i></code>	Specifies the list of hosts to execute command on.
<code>-H <i>chassis</i></code>	Specifies the list of chassis to execute command on.
<code>-u <i>user</i></code>	Specifies the user to perform the command as: <ul style="list-style-type: none"> For hosts, the default is current user code. For chassis, the default is <code>admin</code>.
<code>-m <i>marker</i></code>	Specifies the marker for end of chassis command output. If omitted, defaults to chassis command prompt. This may be a regular expression.
<code>-T <i>timelimit</i></code>	Specifies the time limit in seconds when running host commands. Default is <code>-1</code> (infinite).

Examples

Operations on Host

```
opacmdall date
opacmdall 'uname -a'
opacmdall -h 'elrond arwen' date
HOSTS='elrond arwen' opacmdall date
```

Operations on Chassis

```
opacmdall -C 'ismPortStats -noprompt'
opacmdall -C -H 'chassis1 chassis2' ismPortStats -noprompt'
CHASSIS='chassis1 chassis2' opacmdall -C ismPortStats -noprompt'
```

Environment Variables

The following environment variables are also used by this command:

HOSTS	List of hosts, used if <code>-h</code> option not supplied. See discussion on Selection of Devices on page 41.
-------	--



CHASSIS	List of chassis, used if <code>-C</code> is used and <code>-H</code> and <code>-F</code> options not supplied. See discussion on Selection of Devices on page 41.
HOSTS_FILE	File containing list of hosts, used in absence of <code>-f</code> and <code>-h</code> . See discussion on Selection of Devices on page 41.
CHASSIS_FILE	File containing list of chassis, used in absence of <code>-F</code> and <code>-H</code> . See discussion on Selection of Devices on page 41.
FF_MAX_PARALLEL	When <code>-p</code> option is used, maximum concurrent operations are performed.
FF_SERIALIZE_OUTPUT	Serialize output of parallel operations (yes or no).
FF_CHASSIS_LOGIN_METHOD	How to log into chassis. Can be Telnet or SSH.
FF_CHASSIS_ADMIN_PASSWORD	Password for admin on all chassis. Used in absence of <code>-S</code> option.

Notes

All commands performed with `opacmdall` must be non-interactive in nature. `opacmdall` waits for the command to complete before proceeding. For example, when running host commands such as `rm`, the `-i` option (interactively prompt before removal) should not be used. (Note that this option is sometimes part of a standard bash alias list.) Similarly, when running chassis commands such as `fwUpdateChassis`, the `-reboot` option should not be used because this option causes an immediate reboot and therefore the command never returns. Also, the chassis command `reboot` should not be executed using `opacmdall`. Instead, use the `opachassisadmin reboot` command to reboot one or more chassis. For further information about individual chassis CLI commands, consult the *Intel® Omni-Path Fabric Switches Command Line Interface Reference Guide*. For further information about Linux* operating system commands, consult the man pages.

When performing `opacmdall` against hosts, internally SSH is used. The command `opacmdall` requires that password-less SSH be set up between the host running the Intel® Omni-Path Fabric Suite FastFabric Toolset and the hosts `opacmdall` is operating against. The `opasetupssh` FastFabric tool can aid in setting up password-less SSH.

When performing `opacmdall` against a set of chassis, all chassis must be configured with the same admin password. Alternatively, the `opasetupssh` FastFabric tool can be used to set up password-less SSH to the chassis.

When performing operations against chassis, Intel recommends that you set up SSH keys (see [opasetupssh](#)). If SSH keys are not set up, Intel recommends that you use the `-S` option, to avoid keeping the password in configuration files.



5.6.4 opacaptureall

(Chassis and Host) Captures supporting information for a problem report from all hosts or Intel® Omni-Path Chassis and uploads to this system.

For Hosts: When a host `opacaptureall` is performed, `opacapture` is run to create the specified capture file within `~root` on each host (with the `.tgz` suffix added as needed). The files are uploaded and unpacked into a matching directory name within `upload_dir/hostname/` on the local system. The default file name is `hostcapture`.

For Chassis: When a chassis `opacaptureall` is performed, `opacapture` is run on each chassis and its output is saved to `upload_dir/chassisname/file` on the local system. The default file name is `chassiscapture`.

For both host and chassis capture, the uploaded captures are combined into a `.tgz` file with the file name specified and the suffix `.all.tgz` added.

Syntax

```
opacaptureall [-C] [-p] [-f hostfile] [-F chassisfile] [-h 'hosts']
[-H 'chassis'] [-t portsfile] [-d upload_dir] [-S] [-D detail_level]
[file]
```

Options

<code>--help</code>	Produces full help text.
<code>-C</code>	Performs capture against chassis. Default is <code>hosts</code> .
<code>-p</code>	Performs capture upload in parallel on all host/chassis. For a host capture, this only affects the upload phase.
<code>-f hostfile</code>	Specifies the file with hosts in cluster. Default is <code>/etc/opa/hosts</code> file.
<code>-F chassisfile</code>	Specifies the file containing a list of chassis in the cluster. Default is <code>/etc/opa/chassis</code> file.
<code>-h hosts</code>	Specifies the list of hosts on which to perform a capture.
<code>-H chassis</code>	Specifies the list of chassis on which to perform a capture.
<code>-t portsfile</code>	Specifies the file with list of local HFI ports used to access fabric(s) for switch access, default is <code>/etc/opa/ports</code> file.
<code>-d upload_dir</code>	Specifies the directory to upload to; default is <code>uploads</code> . If not specified, the environment variable <code>UPLOADS_DIR</code> is used. If that is not exported, the default (<code>./uploads</code>) is used.
<code>-S</code>	Securely prompts for password for administrator on a chassis.



<code>-D detail_level</code>	Specifies the level of detail of the capture passed to host opacapture. (Only used for host captures; ignored for chassis captures.)	
1 (Local)		Obtains local information from each host.
2 (Fabric)		In addition to <i>Local</i> , also obtains basic fabric information by queries to the SM and fabric error analysis using <code>opareport</code> .
3 (Fabric +FDB)		In addition to <i>Fabric</i> , also obtains the Forwarding Database (FDB), which includes the switch forwarding tables from the SM.
4 (Analysis)		In addition to <i>Fabric+FDB</i> , also obtains <code>opaallanalysis</code> results. If <code>opaallanalysis</code> has not yet been run, it is run as part of the capture.

Note: Detail levels 2-4 can be used when fabric operational problems occur. If the problem is node-specific, detail level 1 should be sufficient. Detail levels 2-4 require an operational Intel® Omni-Path Fabric Suite Fabric Manager. Typically your support representative requests a given detail level. If a given detail level takes excessively long or fails to be gathered, try a lower detail level.

For detail levels 2-4, the additional information is only gathered on the node running the `opacaptureall` command. The information is gathered for every fabric specified in the `/etc/opa/ports` file.

`file` Specifies the name for capture file. The suffix `.tgz` is appended if it is not specified in the name.

Examples

Host Capture Examples

```
opacaptureall
# Creates a hostcapture directory in upload_dir/hostname/ for each host in
/etc/opa/hosts file, then creates hostcapture.all.tgz.

opacaptureall mycapture
# Creates a mycapture directory in upload_dir/hostname/ for each host in
/etc/opa/hosts file, then creates mycapture.all.tgz.

opacaptureall -h 'arwen elrond' 030127capture
# Gets the list of hosts from arwen elrond file and creates
030127capture.tgz file.
```




Chassis Capture Examples

```
opacaptureall -C
# Creates a chassiscapture file in upload_dir/chassisname/ for each chassis
in /etc/opa/chassis file, then creates chassiscapture.all.tgz.

opacaptureall -C mycapture
# Creates a mycapture.tgz file in upload_dir/chassisname/ for each chassis
in /etc/opa/chassis file, then creates mycapture.all.tgz.

opacaptureall -C -H 'chassis1 chassis2' 030127capture
# Captures from chassis1 and chassis2, and creates 030127capture.tgz file.
```

Environment Variables

The following environment variables are also used by this command:

HOSTS	List of hosts, used if <code>-h</code> option not supplied. See discussion on Selection of Devices on page 41.
CHASSIS	List of chassis, used if <code>-C</code> is used and <code>-h</code> option is not supplied. See discussion on Selection of Devices on page 41.
HOSTS_FILE	File containing a list of hosts, used in the absence of <code>-f</code> and <code>-h</code> . See discussion on Selection of Devices on page 41.
CHASSIS_FILE	File containing a list of chassis, used in the absence of <code>-F</code> and <code>-H</code> . See discussion on Selection of Devices on page 41.
UPLOADS_DIR	Directory to upload to, used in the absence of <code>-d</code> .
FF_MAX_PARALLEL	When <code>-p</code> option is used, maximum concurrent operations are performed.
FF_CHASSIS_LOGIN_METHOD	How to log into chassis. Can be Telnet or SSH.
FF_CHASSIS_ADMIN_PASSWORD	Password for administrator on all chassis. Used in absence of <code>-S</code> option.

More Information

When performing `opacaptureall` against hosts, internally SSH is used. The command `opacaptureall` requires that password-less SSH be set up between the host running Intel® Omni-Path Fabric Suite FastFabric Toolset and the hosts `opacaptureall` is operating against. The `opasetupssh` command can aid in setting up password-less SSH.

When performing operations against chassis, set up of SSH keys is recommended (see [opasetupssh](#) on page 249). If SSH keys are not set up, all chassis must be configured with the same admin password and use of the `-S` option is recommended. The `-S` option avoids the need to keep the password in configuration files.



Note: The resulting host capture files can require significant amounts of space on the Intel® Omni-Path Fabric Suite FastFabric Toolset host. Actual size varies, but sizes can be multiple megabytes per host. Intel recommends that you ensure adequate space is available on the Intel® Omni-Path Fabric Suite FastFabric Toolset system. In many cases, it may not be necessary to run `opacaptureall` against all hosts or chassis; instead, a representative subset may be sufficient. Consult with your support representative for further information.

5.7 File Management Tools

The tools described in this section aid in copying files to and from large groups of nodes in the fabric. Internally, these tools make use of SCP.

The tools require that password-less SSH/SCP is set up between the host running the FastFabric Toolset and the hosts that are being transferred to and from. Use `opasetupssh` to set up password-less SSH/SCP.

5.7.1 `opascpall`

(Linux) Copies files or directories from the current system to multiple hosts in the fabric. When copying large directory trees, use the `-t` option to improve performance. This option tars and compresses the tree, transfers the resulting compressed tarball to each node, and untars it on each node.

Use this tool for copying data files, operating system files, or applications to all the hosts (or a subset of hosts) within the fabric.

- Notes:**
- This tool can only copy from this system to a group of systems in the cluster. To copy from hosts in the cluster to this host, use `opauploadall`.
 - `user@` style syntax cannot be used when specifying filenames.

Syntax

```
opascpall [-p] [-r] [-f hostfile] [-h 'hosts'] [-u user] source_file ... dest_file
opascpall [-t] [-p] [-f hostfile] [-h 'hosts'] [-u user] [source_dir [dest_dir]]
```

Options

<code>--help</code>	Produces full help text.
<code>-p</code>	Performs copy in parallel on all hosts.
<code>-r</code>	Performs recursive copy of directories.
<code>-t</code>	Performs optimized recursive copy of directories using tar. <code>dest_dir</code> is optional. If <code>dest_dir</code> is not specified, it defaults to the current directory name. If both <code>source_dir</code> and <code>dest_dir</code> are omitted, they both default to the current directory name.
<code>-h hosts</code>	Specifies the list of hosts to copy to.



<code>-f hostfile</code>	Specifies the file with hosts in cluster. Default is <code>/etc/opa/hosts</code> file.
<code>-u user</code>	Specifies the user to perform copy to. Default is current user code.
<code>source_file</code>	Specifies the a file or list of source files to copy.
<code>source_dir</code>	Specifies the name of the source directory to copy. If omitted <code>.</code> is used.
<code>dest_file</code> or <code>dest_dir</code>	Specifies the name of the destination file or directory to copy to. If more than one source file, this must be a directory. If omitted current directory name is used.

Example

```
# copy a single file
opascall MPI-PMB /root/MPI-PMB

# efficiently copy an entire directory tree
opascall -t -p /usr/src/opa/mpi_apps /usr/src/opa/mpi_apps

# copy a group of files
opascall a b c /root/tools/

# copy to an explicitly specified set of hosts
opascall -h 'arwen elrond' a b c /root/tools
HOSTS='arwen elrond' opascall a b c /root/tools
```

Environment Variables

The following environment variables are also used by this command:

HOSTS	List of hosts; used if <code>-h</code> option not supplied. See discussion on Selection of Devices on page 41.
HOSTS_FILE	File containing list of hosts; used in absence of <code>-f</code> and <code>-h</code> . See discussion on Selection of Devices on page 41.
FF_MAX_PARALLEL	When the <code>-p</code> option is used, maximum concurrent operations are performed.

5.7.2 opauploadall

(Linux) Copies one or more files from a group of hosts to this system. Since the file name is the same on each host, a separate directory on this system is created for each host and the file is copied to it. This is a convenient way to upload log files or configuration files for review. This tool can also be used in conjunction with `opadownloadall` to upload a host specific configuration file, edit it for each host, and download the new version to all the hosts.

Note: To copy files from this host to hosts in the cluster, use `opascall` or `opadownloadall`. `user@` style syntax cannot be used when specifying filenames.



Syntax

```
opauploadall [-rp] [-f hostfile] [-d upload_dir] [-h 'hosts']  
[-u user] source_file ... dest_file
```

Options

<code>--help</code>	Produces full help text.
<code>-p</code>	Performs copy in parallel on all hosts.
<code>-r</code>	Performs recursive upload of directories.
<code>-f <i>hostfile</i></code>	Specifies the file with hosts in cluster. Default is <code>/etc/opa/hosts</code> file.
<code>-h <i>hosts</i></code>	Specifies the list of hosts to upload from.
<code>-u <i>user</i></code>	Specifies the user to perform copy to. Default is current user code.
<code>-d <i>upload_dir</i></code>	Specifies the directory to upload to. Default is <code>uploads</code> . If not specified, the environment variable <code>UPLOADS_DIR</code> is used. If that is not exported, the default, <code>/uploads</code> , is used.
<code><i>source_file</i></code>	Specifies the name of files to copy to this system, relative to the current directory. Multiple files may be listed.
<code><i>dest_file</i></code>	<p>Specifies the name of the file or directory on this system to copy to. It is relative to <code>upload_dir/HOSTNAME</code>.</p> <p>A local directory within <code>upload_dir/</code> is created for each host. Each uploaded file is copied to <code>upload_dir/HOSTNAME/<i>dest_file</i></code> within the local system. If more than one source file is specified, <code>dest_file</code> is treated as a directory name.</p>

Example

```
# upload two files from 2 hosts  
opauploadall -h 'arwen elrond' capture.tgz /etc/init.d/ipoib.cfg .  
  
# upload two files from all hosts  
opauploadall -p capture.tgz /etc/init.d/ipoib.cfg .  
  
# upload network config files from all hosts  
opauploadall capture.tgz /etc/init.d/ipoib.cfg pre-install
```

Environment Variables

The following environment variables are also used by this command:

HOSTS	List of hosts; used if <code>-h</code> option not supplied. See discussion on Selection of Devices on page 41.
-------	--



HOSTS_FILE	File containing list of hosts; used in absence of <code>-f</code> and <code>-h</code> . See discussion on Selection of Devices on page 41.
UPLOADS_DIR	Directory to upload to, used in absence of <code>-d</code> .
FF_MAX_PARALLEL	When the <code>-p</code> option is used, maximum concurrent operations are performed.

5.7.3 opadownloadall

(Linux) Copies one or more files to a group of hosts from a system. Since the file contents to copy may be different for each host, a separate directory on this system is used for the source files for each host. This can also be used in conjunction with `opauploadall` to upload a host-specific configuration file, edit it for each host, and download the new version to all the hosts.

Note: The tool `opadownloadall` can only copy from this system to a group of hosts in the cluster. To copy files from hosts in the cluster to this host, use `opauploadall`.

Syntax

```
opadownloadall [-rp] [-f hostfile] [-d download_dir] [-h 'HOSTS']
[-u user] source_file ... dest_file
```

Options

<code>--help</code>	Produces full help text.
<code>-r</code>	Performs recursive download of directories.
<code>-p</code>	Performs copy in parallel on all hosts.
<code>-f <i>hostfile</i></code>	Specifies the file with hosts in cluster. The default is <code>/etc/opa/hosts</code> .
<code>-d <i>download_dir</i></code>	Specifies the directory to download files from. The default is <code>downloads</code> . If not specified, the environment variable <code>DOWNLOADS_DIR</code> is used. If that is not exported, the default is used.
<code>-h <i>HOSTS</i></code>	Specifies the list of hosts to download files to.
<code>-u <i>user</i></code>	Specifies the user to perform the copy. The default is the current user code.
	Note: The <code>user@</code> style syntax cannot be used in the arguments to <code>opadownloadall</code> .
<code><i>source_file</i></code>	Specifies the list of source files to copy from the system.



The option *source_file* is relative to *download_dir/hostname*. A local directory within *download_dir/* must exist for each host being downloaded to. Each downloaded file is copied from *download_dir/hostname/source_file*.

dest_file

Specifies the name of the file or directory on the destination hosts to copy to.

If more than one source file is specified, *dest_file* is treated as a directory name. The given directory must already exist on the destination host. The copy fails for hosts where the directory does not exist.

Example

```
opadownloadall -h 'arwen elrond' irqbalance vncservers /etc
# Copies two files to 2 hosts

opadownloadall -p irqbalance vncservers /etc
# Copies two files to all hosts
```

Environment Variables

The following environment variables are also used by this command:

HOSTS	List of hosts; used if <code>-h</code> option not supplied. See discussion on Selection of Devices on page 41.
HOSTS_FILE	File containing list of hosts; used in absence of <code>-f</code> and <code>-h</code> . See discussion on Selection of Devices on page 41.
FF_MAX_PARALLEL	When the <code>-p</code> option is used, the maximum concurrent operations are performed.
DOWNLOADS_DIR	Directory to download from, used in absence of <code>-d</code> .

5.7.4 Simplified Editing of Node-Specific Files

(Linux) The combination of `opauploadall` and `opadownloadall` provide a powerful yet simple to use mechanism for reviewing or editing node-specific files without the need to log in to each node.

For example, assume the file `/etc/network-scripts/ifcfg-ib1` needs to be reviewed and edited for each host. This file typically contains the IP configuration information for IPoIB and may contain a unique IP address per host. Perform the following steps:

1. To upload the file from all the hosts, use the command: `uploadall /etc/network-scripts/ifcfg-ib1 ifcfg-ib1`
2. Edit the uploaded files with an editor, such as `vi` with the command: `vi uploads/*/ifcfg-ib1`



3. If the file was changed for some or all of the hosts, it can then be downloaded to all the hosts with the command: `opadownloadall -d uploads ifcfg-ib1 /etc/network-scripts/ifcfg-ib1`

Alternatively, you can download the file to a subset of hosts using the `-h` option or by creating an alternate host list file: `opadownloadall -d uploads -h 'host1 host32' ifcfg-ib1 /etc/network-scripts/ifcfg-ib1`

Note: When downloading to a subset of hosts, make sure that only the hosts uploaded from are specified.

5.7.5 Simplified Setup of Node-Generic Files

(Linux) `opascpall` can provide a powerful yet simple to use mechanism for transferring generic files to all nodes.

For example, assume all nodes in the cluster use the same DNS server and TCP/IP name resolution. Perform the following steps:

1. Create an appropriate local file with the desired information. For example: `vi resolv.conf`
2. Copy the file to all hosts with the command: `opascpall resolv.conf /etc/resolv.conf`

5.8 Fabric Link and Port Control

The CLIs described in this section are used for manipulation of device and port states in the fabric.

5.8.1 opadisableports

(Linux) Accepts a CSV file listing links to disable. For each HFI-SW link, the switch side of the link is disabled. For each SW-SW link, the side of the link with the lower LID (typically, the side closest to the SM) is disabled. This approach generally permits a future `opaenableports` operation to re-enable the port once the issue is corrected or ready to be retested. When using the `-R` option, this tool does not look at the routes, it disables the switch ports with the lower value LID. The list of disabled ports is tracked in `/etc/opa/disabled*.csv`.

Syntax

```
opadisableports [-R] [-h hfi] [-p port]
[reason] < disable.csv
```

Options

- | | |
|---------------------|---|
| <code>--help</code> | Produces full help text. |
| <code>-R</code> | Does not attempt to get routes for computation of distance. Instead, disables switch port with lower LID assuming that it is closer to this node. |



<code>-h hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>reason</code>	Specifies optional text describing why ports are being disabled. If used, text is saved in the reason field of the output file.
<code>disable.csv</code>	<p>Specifies the input file listing the links to disable. The list is of the form:</p> <pre>NodeGUID;PortNum;NodeType;NodeDesc;NodeGUID;PortNum; NodeType;NodeDesc;Reason</pre> <p>For each listed link, the switch port closer to this node is disabled. The <code>reason</code> field is optional. An input file such as this can be generated by using <code>opaextractbadlinks</code>, <code>opaextractmissinglinks</code>, or <code>opaextractsellinks</code>.</p> <p>Information about the links disabled and the reason is saved (in the same format) to an output file named <code>/etc/opa/disabled:hfi:port.csv</code> where the <code>hfi:port</code> part of the file name is replaced by the HFI number and the port number being operated on (such as 1:1 or 2:1). This CSV file can be used as input to <code>opaenableports</code>.</p>

-h and -p options permit a variety of selections:

<code>-h 0</code>	First active port in system (default).
<code>-h 0 -p 0</code>	First active port in system.
<code>-h x</code>	First active port on HFI x.
<code>-h x -p 0</code>	First active port on HFI x.
<code>-h 0 -p y</code>	Port y within system (no matter which ports are active).
<code>-h x -p y</code>	HFI x, port y.

Examples

```
opadisableports 'bad cable' < disable.csv
opadisableports -h 1 -p 1 'dead servers' < disable.csv
opaextractsellinks -F lid:3 | opadisableports 'bad server'
opaextractmissinglinks -T /etc/opa/topology.0:0.xml | opadisableports
```




5.8.2 opaenableports

(Linux) Accepts a disabled ports input file and re-enables the specified ports. The input file can be `/etc/opa/disabled*.csv` or a user-created subset of such a file. After enabling the port, it is removed from `/etc/opa/disabled*.csv`.

Syntax

```
opaenableports [-h hfi] [-p port] < disabled.csv
```

Options

<code>--help</code>	Produces full help text.
<code>-h <i>hfi</i></code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p <i>port</i></code> port is a system-wide port number. (Default is 0.)
<code>-p <i>port</i></code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>disabled.csv</code>	Specifies the input file listing the ports to enable. The list is of the form: <code>NodeGUID;PortNum;NodeType;NodeDesc;Ignored</code> . An input file like this is generated in <code>/etc/opa/disabled*</code> by <code>opadisableports</code> .

-h and -p options permit a variety of selections:

<code>-h 0</code>	First active port in system (default).
<code>-h 0 -p 0</code>	First active port in system.
<code>-h x</code>	First active port on HFI x.
<code>-h x -p 0</code>	First active port on HFI x.
<code>-h 0 -p y</code>	Port y within system (no matter which ports are active).
<code>-h x -p y</code>	HFI x, port y.

Examples

```
opaenableports < disabled.csv
opaenableports < /etc/opa/disabled:1:1.csv
opaenableports -h 1 -p 1 < disabled.csv
```



Other Information

For messages containing `skipping` ports, either the device is offline or the other end of the link has been disabled and the device is no longer accessible in-band. The end of the link previously disabled by `opedisableports` or `opadisablehosts` can be found in `/etc/opa/disabled:1:1.csv`.

5.8.3 opadisablehosts

(Linux) Searches for a set of hosts in the fabric and disables their corresponding switch port.

Syntax

```
opadisablehosts [-h hfi] [-p port] reason host ...
```

Options

- | | |
|-----------------------------|--|
| <code>--help</code> | Produces full help text. |
| <code>-h <i>hfi</i></code> | Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p <i>port</i></code> port is a system-wide port number. (Default is 0.) |
| <code>-p <i>port</i></code> | Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.) |
| <code><i>reason</i></code> | Specifies the text describing the reason hosts are being disabled. <code><i>reason</i></code> is saved in the <code><i>reason</i></code> field of the output file. |

Information about the links disabled is written to a CSV file. By default, this file is named `/opa/disabled:hfi:port.csv` where the `hfi:port` part of the file name is replaced by the HFI number and the port number being operated on (such as 1:1 or 2:1). This CSV file can be used as input to `opaenableports`.

The list is of the form:

`NodeGUID;PortNum;NodeType;NodeDesc;NodeGUID;`

`PortNum;NodeType;NodeDesc;Reason` For each listed link, the switch port closer to this is the one that has been disabled.

`host ...` Defines one or more hosts that are affected by the `reason`.

-h and -p options permit a variety of selections:

- | | |
|------------------------|--|
| <code>-h 0</code> | First active port in system (default). |
| <code>-h 0 -p 0</code> | First active port in system. |
| <code>-h x</code> | First active port on HFI x. |
| <code>-h x -p 0</code> | First active port on HFI x. |



-h 0 -p y Port y within system (no matter which ports are active).

-h x -p y HFI x, port y.

Examples

```
opadisablehosts 'bad DRAM' compute001 compute045
opadisablehosts -h 1 -p 2 'crashed' compute001 compute045
```

5.8.4 opaswdisableall

(Linux) Disables all unused switch ports.

Syntax

```
opaswdisableall [-t portsfile] [-p ports] [-F focus] [-K mkey]
```

Options

- help Produces full help text.
- t *portsfile* Specifies the file with list of local HFI ports used to access fabrics when clearing counters. Default is `/etc/opa/ports` file.
- p *ports* Specifies the list of local HFI ports used to access fabrics for counter clear.

Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format `hfi:port`, for example:

0:0 First active port in system.

0:y Port y within system.

x:0 First active port on HFI x.

x:y HFI x, port y.
- F *focus* Specifies the an `opareport`-style focus argument to limit the scope of operation. For more information, see [Advanced Focus](#) on page 197.
- K *mkey* Specifies the SM management key to access remote ports.

Examples

```
opaswdisableall
opaswdisableall -p '1:1 1:2 2:1 2:2'
```



Environment Variables

The following environment variables are also used by this command:

PORTS List of ports, used in absence of `-t` and `-p`.

PORTS_FILE File containing list of ports, used in absence of `-t` and `-p`.

5.8.5 opaswenableall

(Linux) Re-enables all unused (or disabled) switch ports.

Syntax

```
opaswenableall [-t portsfile] [-p ports] [-F focus] [-K mkey]
```

Options

`--help` Produces full help text.

`-t portsfile` Specifies the file with list of local HFI ports used to access fabrics for operation. Default is `/etc/opa/ports` file.

`-p ports` Specifies the list of local HFI ports used to access fabrics for operation.

Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format `hfi:port`, for example:

`0:0` First active port in system.

`0:y` Port *y* within system.

`x:0` First active port on HFI *x*.

`x:y` HFI *x*, port *y*.

`-F focus` Specifies an `opareport`-style focus argument to limit the scope of operation. For more information, see [Advanced Focus](#) on page 197.

`-K mkey` Specifies the SM management key to access remote ports.

Examples

```
opaswenableall
opaswenableall -p '1:1 1:2 2:1 2:2'
```

Environment Variables

The following environment variables are also used by this command:



PORTS List of ports, used in absence of `-t` and `-p`.

PORTS_FILE File containing list of ports, used in absence of `-t` and `-p`.

5.8.6 opaledports

Toggles the beaconing LED state of HFIs, switches, and switch ports. `opaledports` is a useful aid for finding specific physical nodes in a crowded data center. It supports the CSV link format provided by `opaextractsellinks`.

Syntax

```
opaledports [-h hfi] [-p port] [-C] [-s|-d] [on|off] < portlist.csv
```

Options

- `--help` Produces full help text.
- `-h hfi` Specifies the HFI, numbered 1..n. Using 0 specifies that the `-p port` port is a system-wide port number. (Default is 0.)
- `-p port` Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
- `-C` Clears beaconing LED on all ports.
NOTE: If `-C` is entered, no other options are valid.
- `-s` Affects source side (first node) of link only.
- `-d` Affects destination side (second node) of link only.
- `on|off` Turns on or off the beaconing LED. Options include:

 - `on` Turns on beaconing LED.
 - `off` Turns off beaconing LED.
- `portlist.csv` Specifies the file listing the links to process. The list is of the form:


```
NodeGUID;PortNum;NodeType;NodeDesc;NodeGUID;PortNum;
NodeType;NodeDesc;Dontcare
```

Examples

```
echo "0x001175010165acld;1;FI;phkpstl035 hfi1_0"|opaledports on
opaledports on < portlist.csv
opaextractsellinks -F led:on | opaledports off
opaledports -C
```



5.9 Fabric Debug

The CLIs described in this section are used for gathering various fabric information from the FM for debug and analysis purposes.

5.9.1 opafequery

Note: This tool is being deprecated. Functionality will be moved to `opasaquery` and `opapaquery` in a future release.

(All) Used for testing or debugging performance administration (PA) operations to the Fabric Executive (FE). This tool performs custom PA client/server queries. The output formats and arguments are very similar to `opapaquery`.

Syntax

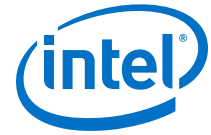
```
opafequery [-v] [-a ipAddr | -h hostName] [-E] [-T paramsfile] -o type  
[SA options | PA options]
```

General Options

<code>--help</code>	Produces full help text.
<code>-v/--verbose</code>	Specifies the verbose output.
<code>-a/--ipAddr <i>ipAddr</i></code>	Specifies the IP address of node running the FE. This options supports IPv4 and IPv6 addresses with port number; for example, <code>127.0.0.1:3245</code> or <code>:::1:3245</code> .
<code>-h/--hostName <i>hostName</i></code>	Specifies the host name of node running the FE. This option supports host name with port number; for example, <code>localhost:3245</code> .
<code>-o/--output <i>output</i></code>	Specifies the output type. See SA Output Types and PA Output Types for details.
<code>-E/--feEsm</code>	Specifies an ESM FE to indicate what SSL/TLS version to use when connecting.
<code>-T/--sslParmsFile <i>filename</i></code>	Specifies the SSL/TLS parameters XML file. Default = <code>/etc/opa/opaff.xml</code>

SA Specific Options

<code>-I/--IB</code>	Issues query in legacy InfiniBand* format.
<code>-l/--lid <i>lid</i></code>	Queries a specific LID.



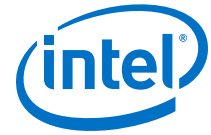
<code>-k/--pkey pkey</code>	Queries a specific pkey.
<code>-i/--vfindex vfindex</code>	Queries a specific vfindex.
<code>-S/--serviceId serviceId</code>	Queries a specific service ID.
<code>-L/--SL SL</code>	Queries by service level.
<code>-t/--type type</code>	Queries by node type.
<code>-s/--sysguid guid</code>	Queries by system image GUID.
<code>-n/--nodeguid guid</code>	Queries by node GUID.
<code>-p/--portguid guid</code>	Queries by port GUID.
<code>-u/--portgid gid</code>	Queries by port GUID.
<code>-m/--mcgid gid</code>	Queries by multicast GUID.
<code>-d/--desc name</code>	Queries by node name/description.
<code>-P/--guidpair 'guid guid'</code>	Queries by a pair of port GUIDs.
<code>-G/--gidpair 'gid gid'</code>	Queries by a pair of GUIDs.
<code>-B/--guidlist 'sguid ...;dguid ...'</code>	Queries by a list of port GUIDs.
<code>-A/--gidlist 'sgid ...;dgid ...'</code>	Queries by a list of GUIDs.
<code>-x/--sourcegid gid</code>	Specifies a source GUID for certain queries.

PA Specific Options

<code>-g/--groupName groupName</code>	Queries by group name for groupInfo.
<code>-l/--lid lid</code>	Queries by LID of node for portCounters.
<code>-N/--portNumber</code>	Queries by port number for portCounters.
<code>-f/--delta</code>	Queries by delta flag for portCounters. Values include: 0 or 1.
<code>-j/--begin date_time</code>	Obtains portCounters over an interval beginning at <i>date_time</i> . <i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago.



<code>-q/--end date_time</code>	<p>Obtains portCounters over an interval ending at <i>date_time</i>.</p> <p><i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago.</p>
<code>-U/--userCnters</code>	<p>Queries by user-controlled counters flag for portCounters.</p>
<code>-e/--select</code>	<p>Specifies the 32-bit select flag for clearing port counters select bits. 0 is least significant (rightmost).</p> <p>Bit descriptions are listed below in the order "mask - bit - location":</p> <ul style="list-style-type: none">• 0x80000000 - 31 - Transmit Data (XmitData)• 0x40000000 - 30 - Receive Data (RcvData)• 0x20000000 - 29 - Transmit Packets (XmitPkts)• 0x10000000 - 28 - Receive Packets (RcvPkts)• 0x08000000 - 27 - Multicast Transmit Packets (MulticastXmitPkts)• 0x04000000 - 26 - Multicast Receive Packets (MulticastRcvPkts)• 0x02000000 - 25 - Transmit Wait (XmitWait)• 0x01000000 - 24 - Congestion Discards (CongDiscards)• 0x00800000 - 23 - Receive FECN (RcvFECN)• 0x00400000 - 22 - Receive BECN (RcvBECN)• 0x00200000 - 21 - Transmit Time Congestion (XmitTimeCong)• 0x00100000 - 20 - Transmit Time Wasted BW (XmitWastedBW)• 0x00080000 - 19 - Transmit Time Wait Data (XmitWaitData)• 0x00040000 - 18 - Receive Bubble (RcvBubble)• 0x00020000 - 17 - Mark FECN (MarkFECN)• 0x00010000 - 16 - Receive Constraint Errors (RcvConstraintErrors)• 0x00008000 - 15 - Receive Switch Relay (RcvSwitchRelayErrors)• 0x00004000 - 14 - Transmit Discards (XmitDiscards)• 0x00002000 - 13 - Transmit Constraint Errors (XmitConstraintErrors)• 0x00001000 - 12 - Receive Remote Physical Errors (RcvRemotePhysicalErrors)• 0x00000800 - 11 - Local Link Integrity (LocalLinkIntegrityErrors)



- 0x00000400 - 10 - Receive Errors (RcvErrors)
- 0x00000200 - 9 - Excessive Buffer Overrun (ExcessiveBufferOverruns)
- 0x00000100 - 8 - FM Configuration Errors (FMConfigErrors)
- 0x00000080 - 7 - Link Error Recovery (LinkErrorRecovery)
- 0x00000040 - 6 - Link Error Downed (LinkDowned)
- 0x00000020 - 5 - Uncorrectable Errors (UncorrectableErrors)

`-c/--focus`
focus

Specifies the focus select value for getting focus ports. Values include:

<code>utilhigh</code>	Sorted by utilization - highest first.
<code>pktrate</code>	Sorted by packet rate - highest first.
<code>utillow</code>	Sorted by utilization - lowest first.
<code>integrity</code>	Sorted by integrity category - highest first.
<code>congestion</code>	Sorted by congestion category - highest first.
<code>smacongesion</code>	Sorted by SMA congestion category - highest first.
<code>bubbles</code>	Sorted by bubble category - highest first.
<code>security</code>	Sorted by security category - highest first.
<code>routing</code>	Sorted by routing category - highest first.

`-w/--start`

Specifies the start of window for focus ports - should always be 0.

`-r/--range`
range

Specifies the size of window for focus ports list.

`-b/--imgNum`

Specifies the 64-bit image number. May be used with `groupInfo`, `groupConfig`, `portCounters` (delta) outputs.

`-O/--imgOff`

Specifies the image offset. May be used with `groupInfo`, `groupConfig`, `portCounters` (delta) outputs.

`-y/--imgTime`

Specifies the image time. May be used with `imageInfo`, `groupInfo`, `groupInfo`, `groupConfig`, `freezeImage`, `focusPorts`, `vfInfo`, `vfConfig`, and `vfFocusPorts`. Will return closest image within image interval if possible. See `--begin/--end` above for format.



- `-F/--moveImgNum` Specifies the 64-bit image number. Used with `moveFreeze` output to move a freeze image.
- `-M/--moveImgOff`
`ImgOff` Specifies the image offset. May be used with `moveFreeze` output to move a freeze image.
- `-V/--vfName` Queries by VF name for `vfInfo`.

SA Output Types

Output types include:

- `saclassPortInfo` Specifies the class port info.
- `systemguid` Lists the system image GUIDs.
- `nodeguid` Lists the node GUIDs.
- `portguid` Lists the port GUIDs.
- `lid` Lists the LIDs.
- `desc` Lists the node descriptions/names.
- `path` Lists the path records.
- `node` Lists the node records.
- `portinfo` Lists the port info records.
- `sminfo` Lists the SM info records.
- `swinfo` Lists the switch info records.
- `link` Lists the link records.
- `scsc` Lists the SC to SC mapping table records.
- `slsc` Lists the SL to SC mapping table records.
- `scsl` Lists the SC to SL mapping table records.
- `scvlt` Lists the SC to VLt table records.
- `scvltnt` Lists the SC to VLnt table records.
- `vlarb` Lists the VL arbitration table records.
- `pkey` Lists the PKey table records.



<code>service</code>	Lists the service records.
<code>mcmember</code>	Lists the multicast member records.
<code>inform</code>	Lists the inform info records.
<code>linfdb</code>	Lists the switch linear forwarding database (FDB) records.
<code>mcfdb</code>	Lists the switch multicast FDB records.
<code>trace</code>	Lists the trace records.
<code>vfinfo</code>	Lists the vFabrics.
<code>vfinfocsv</code>	Lists the vFabrics in CSV format.
<code>vfinfocsv2</code>	Lists the vFabrics in CSV format with enums.
<code>fabricinfo</code>	Provides a summary of fabric devices.
<code>quarantine</code>	Lists the quarantined nodes.
<code>conginfo</code>	Lists the Congestion Info Records.
<code>swcongset</code>	Lists the Switch Congestion Settings.
<code>hficongset</code>	Lists the HFI Congestion Settings.
<code>hficongcon</code>	Lists the HFI Congestion Control Settings.
<code>bfrctrl</code>	Lists the buffer control tables.
<code>cableinfo</code>	Lists the Cable Info records.
<code>portgroup</code>	Lists the AR Port Group records.
<code>portgroupfdb</code>	Lists the AR Port Group FWD records.
<code>swcost</code>	Lists the switch cost records.

PA Output Types

Output types include:

<code>paclassPortInfo</code>	Specifies the class port info.
<code>groupList</code>	Lists the PA groups.



<code>groupInfo</code>	Provides a summary statistics of a PA group. Requires <code>-g</code> option for <code>groupName</code> .
<code>groupConfig</code>	Specifies the configuration of a PA group. Requires <code>-g</code> option for <code>groupName</code> .
<code>portCounters</code>	Specifies the port counters of fabric port. Requires <code>-l lid</code> and <code>-N port</code> options. Optionally, use the <code>-f delta</code> option.
<code>clrPortCounters</code>	Clears port counters of fabric port. Requires <code>-l lid</code> , <code>-N port</code> , and <code>-e select</code> options.
<code>clrAllPortCounters</code>	Clears all port counters in fabric.
<code>pmConfig</code>	Retrieves PM configuration information.
<code>freezeImage</code>	Creates freeze frame for image ID. Requires <code>-b imgNum</code> .
<code>releaseImage</code>	Releases freeze frame for image ID. Requires <code>-b imgNum</code> .
<code>renewImage</code>	Renews lease for freeze frame for image ID. Requires <code>-b imgNum</code> .
<code>moveFreeze</code>	Moves freeze frame from image ID to new image ID. Requires <code>-b imgNum</code> and <code>-F moveImgNum</code> .
<code>focusPorts</code>	Gets sorted list of ports using utilization or error values (from group buckets).
<code>imageInfo</code>	Gets information about a PA image (timestamps and other details). Requires <code>-b imgNum</code> .
<code>vfList</code>	Lists the virtual fabrics.
<code>vfInfo</code>	Provides a summary statistics of a virtual fabric. Requires <code>-V vfName</code> option.
<code>vfConfig</code>	Specifies the configuration of a virtual fabric. Requires <code>-V vfName</code> option.
<code>vfPortCounters</code>	Specifies the port counters of fabric port. Requires <code>-V vfName</code> , <code>-l lid</code> , and <code>-N port</code> options. Optionally, use the <code>-f delta</code> option.
<code>vfFocusPorts</code>	Gets sorted list of virtual fabric ports using utilization or error values (from VF buckets). Requires <code>-V vfName</code> option.
<code>clrVfPortCounters</code>	Clears VF port counters of fabric port. Requires <code>-l lid</code> , <code>-N port</code> , <code>-e select</code> , and <code>-V vfName</code> options.



Examples

```
opafequery -o saclassPortInfo
opafequery -h stewart -o paclassPortInfo
opafequery -a 172.21.2.155 -o saclassPortInfo
opafequery -o groupList
opafequery -o groupInfo -g All
opafequery -o groupConfig -g All
opafequery -h stewart -o groupInfo -g All
opafequery -a 172.21.2.155 -o groupInfo -g All
opafequery -o portCounters -l 1 -N 1 -d 1
opafequery -o portCounters -l 1 -N 1 -d 1 -e 0x20000000d02 -O 1
opafequery -o pmConfig
opafequery -o freezeImage 0x20000000d02
opafequery -o releaseImage -b 0xd01
opafequery -o renewImage -b 0xd01
opafequery -o moveFreeze -b 0xd01 -m 0x20000000d02 -M -2
opafequery -o focusPorts -g All -f 0x00030001 -w 0 -r 20
opafequery -o imageInfo -b 0x20000000d02
```

5.9.2 opapaquery

(All) Performs various queries of the performance management (PM)/performance administration (PA) agent and provides details about fabric performance. Refer to the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide* for a description of the operation and client services of the PM/PA.

By default, `opapaquery` queries the most recent data. However, if an image number (`imgNum`) and/or image offset (`imgOff`) is provided, the query returns previous sweep data. Queries that access previous sweep data return with the absolute image number representing that data, and therefore have an image offset of zero.

`opapaquery`'s operation is dependent on an Intel® Omni-Path Fabric Suite Fabric Manager version 6.0 or greater running as master SM/PM in the fabric.

By default, `opapaquery` uses the first active port on the local system. However, if the Fabric Management Node is connected to more than one fabric (for example, a subnet), the HFI and port may be specified to select the fabric whose PA is to be queried.

Syntax

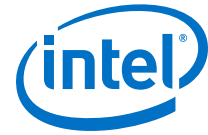
```
opapaquery [-v] [-h hfi] [-p port] [-o type] [-g groupName] [-l nodeId]
[-P portNumber] [-d delta] [-j date_time] [-q date_time] [-U] [-s select]
[-f focus] [-S start] [-r range] [-n imgNum] [-O imgOff] [-y imgTime]
[-m moveImgNum] [-M moveImgOff] [-V vfName]
```

Options

<code>-v/--verbose</code>	Specifies the verbose output.
<code>-h/--hfi hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)
<code>-p/--port port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)



<code>-o/--output <i>type</i></code>	Specifies the output type, default is <code>groupList</code> . See Output Types on page 281.
<code>-g/--groupName <i>groupName</i></code>	Specifies the group name for <code>groupInfo</code> query.
<code>-l/--lid <i>lid</i></code>	Specifies the LID of node for <code>portCounters</code> query.
<code>-P/--portNumber <i>portNumber</i></code>	Specifies the port number for <code>portCounters</code> query.
<code>-d/--delta <i>delta</i></code>	Specifies the delta flag for <code>portCounters</code> query - 0 or 1.
<code>-j/--begin <i>date_time</i></code>	<p>Obtains <code>portCounters</code> over an interval beginning at <i>date_time</i>.</p> <p><i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago."</p>
<code>-q/--end <i>date_time</i></code>	<p>Obtains <code>portCounters</code> over an interval ending at <i>date_time</i>.</p> <p><i>date_time</i> may be a time entered as HH:MM[:SS] or date as mm/dd/YYYY, dd.mm.YYYY, YYYY-mm-dd or date followed by time; for example, "2016-07-04 14:40". Relative times are taken as "x [second minute hour day](s) ago."</p>
<code>-U/--userCnters</code>	Queries by user-controlled counters flag for <code>portCounters</code> .
<code>-s/--select <i>select</i></code>	<p>Specifies the 32-bit select flag for clearing port counters.</p> <p>Select bits for <code>clrPortCounters</code>. 0 is the least significant bit (rightmost). The <code>clrPortCounters</code> bit descriptions are listed in the order "mask - bit - location" below:</p> <ul style="list-style-type: none">• 0x80000000 - 31 - Transmit Data (XmitData)• 0x40000000 - 30 - Receive Data (RcvData)• 0x20000000 - 29 - Transmit Packets (XmitPkts)• 0x10000000 - 28 - Receive Packets (RcvPkts)• 0x08000000 - 27 - Multicast Transmit Packets (MulticastXmitPkts)• 0x04000000 - 26 - Multicast Receive Packets (MulticastRcvPkts)• 0x02000000 - 25 - Transmit Wait (XmitWait)• 0x01000000 - 24 - Congestion Discards (CongDiscards)• 0x00800000 - 23 - Receive FECN (RcvFECN)• 0x00400000 - 22 - Receive BECN (RcvBECN)



- 0x00200000 - 21 - Transmit Time Congestion (XmitTimeCong)
- 0x00100000 - 20 - Transmit Time Wasted BW (XmitWastedBW)
- 0x00080000 - 19 - Transmit Time Wait Data (XmitWaitData)
- 0x00040000 - 18 - Receive Bubble (RcvBubble)
- 0x00020000 - 17 - Mark FECN (MarkFECN)
- 0x00010000 - 16 - Receive Constraint Errors (RcvConstraintErrors)
- 0x00008000 - 15 - Receive Switch Relay (RcvSwitchRelayErrors)
- 0x00004000 - 14 - Transmit Discards (XmitDiscards)
- 0x00002000 - 13 - Transmit Constraint Errors (XmitConstraintErrors)
- 0x00001000 - 12 - Receive Remote Physical Errors (RcvRemotePhysicalErrors)
- 0x00000800 - 11 - Local Link Integrity (LocalLinkIntegrityErrors)
- 0x00000400 - 10 - Receive Errors (RcvErrors)
- 0x00000200 - 9 - Excessive Buffer Overrun (ExcessiveBufferOverruns)
- 0x00000100 - 8 - FM Configuration Errors (FMConfigErrors)
- 0x00000080 - 7 - Link Error Recovery (LinkErrorRecovery)
- 0x00000040 - 6 - Link Error Downed (LinkDowned)
- 0x00000020 - 5 - Uncorrectable Errors (UncorrectableErrors)

Select bits for `clrVfPortCounters`. 0 is the least significant bit (rightmost). The `clrVfPortCounters` bit descriptions are listed in the order "mask - bit - location" below:

- 0x80000000 - 31 - VL Transmit Data (VLXmitData)
- 0x40000000 - 30 - VL Receive Data (VLRcvData)
- 0x20000000 - 29 - VL Transmit Packets (VLXmitPkts)
- 0x10000000 - 28 - VL Receive Packets (VLRcvPkts)
- 0x08000000 - 27 - VL Transmit Discards (VLXmitDiscards)
- 0x04000000 - 26 - VL Congestion Discards (VLCongDiscards)
- 0x02000000 - 25 - VL Transmit Wait (VLXmitWait)
- 0x01000000 - 24 - VL Receive FECN (VLRcvFECN)



- 0x00800000 - 23 - VL Receive BECN (VLRcvBECN)
- 0x00400000 - 22 - VL Transmit Time Congestion (VLXmitTimeCong)
- 0x00200000 - 21 - VL Transmit Wasted BW (VLXmitWastedBW)
- 0x00100000 - 20 - VL Transmit Wait Data (VLXmitWaitData)
- 0x00080000 - 19 - VL Receive Bubble (VLRcvBubble)
- 0x00040000 - 18 - VL Mark FECN (VLMarkFECN)
- Bits 17-0 reserved

<code>-f/--focus <i>focus</i></code>	Specifies the focus select value for getting <i>focus</i> ports. <i>focus</i> select values are:
<code>utilhigh</code>	Sorted by utilization - highest first.
<code>pktrate</code>	Sorted by packet rate - highest first.
<code>utillow</code>	Sorted by utilization - lowest first.
<code>integrity</code>	Sorted by integrity category - highest first.
<code>congestion</code>	Sorted by congestion category - highest first.
<code>smacongestion</code>	Sorted by SMA congestion category - highest first.
<code>bubbles</code>	Sorted by bubble category - highest first.
<code>security</code>	Sorted by security category - highest first.
<code>routing</code>	Sorted by routing category - highest first.
<code>-S/--start <i>start</i></code>	Specifies the start of window for focus ports, should always be 0.
<code>-r/--range <i>range</i></code>	Specifies the size of window for focus ports list.
<code>-n/--imgNum <i>imgNum</i></code>	Specifies the 64-bit image number. Can be used with <code>groupInfo</code> , <code>groupConfig</code> , <code>portCounters</code> (<code>delta</code>).
<code>-O/--imgOff <i>imgOff</i></code>	Specifies the image offset. Can be used with <code>groupInfo</code> , <code>groupConfig</code> , <code>portCounters</code> (<code>delta</code>).



<code>-y/--imgTime</code>	Specifies the image time. May be used with <code>imageinfo</code> , <code>groupInfo</code> , <code>groupInfo</code> , <code>groupConfig</code> , <code>freezeImage</code> , <code>focusPorts</code> , <code>vfInfo</code> , <code>vfConfig</code> , and <code>vfFocusPorts</code> . Will return closest image within image interval if possible. See <code>--begin/--end</code> above for format.
<code>-m/--moveImgNum</code> <code>moveImgNum</code>	Specifies the 64-bit image number. Used with <code>moveFreeze</code> to move a freeze image.
<code>-M/--moveImgOff</code> <code>moveImgOff</code>	Specifies the image offset. Can be used with <code>moveFreeze</code> to move a freeze image.
<code>-V/--vfName</code> <i>vfName</i>	Specifies the VF name for <code>vfInfo</code> query.

-h and -p options permit a variety of selections:

<code>-h 0</code>	First active port in system (default).
<code>-h 0 -p 0</code>	First active port in system.
<code>-h x</code>	First active port on HFI x.
<code>-h x -p 0</code>	First active port on HFI x.
<code>-h 0 -p y</code>	Port y within system (no matter which ports are active).
<code>-h x -p y</code>	HFI x, port y.

Output Types

<code>classPortInfo</code>	Specifies the class port info.
<code>groupList</code>	Specifies the list of PA groups.
<code>groupInfo</code>	Specifies the summary statistics of a PA group. Requires <code>-g</code> option for <i>groupName</i> .
<code>groupConfig</code>	Specifies the configuration of a PA group. Requires <code>-g</code> option for <i>groupName</i> .
<code>portCounters</code>	Specifies the port counters of fabric port. Requires <code>-l lid</code> and <code>-P port</code> options, <code>-d delta</code> is optional.
<code>clrPortCounters</code>	Clears port counters of fabric port. Requires <code>-l lid</code> and <code>-P port</code> , and <code>-s select</code> options.
<code>clrAllPortCounters</code>	Clears all port counters in fabric.
<code>pmConfig</code>	Retrieves PM configuration information.



freezeImage	Creates freeze frame for image ID. Requires <code>-n imgNum</code> .
releaseImage	Releases freeze frame for image ID. Requires <code>-n imgNum</code> .
renewImage	Renews lease for freeze frame for image ID. Requires <code>-n imgNum</code> .
moveFreeze	Moves freeze frame from image ID to new image ID. Requires <code>-n imgNum</code> and <code>-m moveImgNum</code> .
focusPorts	Gets sorted list of ports using utilization or error values (from group buckets). Requires <code>-g groupname</code> , <code>-f focus</code> , <code>-S start</code> , <code>-r range</code> .
imageInfo	Gets configuration of a PA image (timestamps, etc.). Requires <code>-n imgNum</code> .
vfList	Specifies the list of virtual fabrics.
vfInfo	Specifies the summary statistics of a virtual fabric. Requires <code>-V</code> option for <code>vfName</code> .
vfConfig	Specifies the configuration of a virtual fabric. Requires <code>-V</code> option for <code>vfName</code> .
vfPortCounters	Specifies the port counters of fabric port. Requires <code>-V vfName</code> , <code>-l lid</code> and <code>-P port</code> options, <code>-d delta</code> is optional.
vfFocusPorts	Gets sorted list of virtual fabric ports using utilization or error values (from VF buckets). Requires <code>-V vfname</code> , <code>-f focus</code> , <code>-S start</code> , <code>-r range</code> .
clrVfPortCounters	Clears VF port counters of fabric port. Requires <code>-l lid</code> , <code>-P port</code> , <code>-s select</code> , and <code>-V vfname</code> options.

Examples

```
opapaquery -o classPortInfo
opapaquery -o groupList
opapaquery -o groupInfo -g All
opapaquery -o groupConfig -g All
opapaquery -o portCounters -l 1 -P 1 -d 1
opapaquery -o portCounters -l 1 -P 1 -d 1 -n 0x20000000d02 -O 1
opapaquery -o portCounters -l 1 -P 1 -d 1 -j 13:30 -q 14:20
opapaquery -o clrPortCounters -l 1 -P 1 -s 0xC0000000
#clears XmitData & RcvData
opapaquery -o clrAllPortCounters -s 0xC0000000
#clears XmitData & RcvData on all ports
opapaquery -o PMConfig
opapaquery -o freezeImage -n 0x20000000d02
opapaquery -o releaseImage -n 0xd01
opapaquery -o renewImage -n 0xd01
opapaquery -o moveFreeze -n 0xd01 -m 0x20000000d02 -M -2
opapaquery -o focusPorts -g All -f 0x00030001 -S 0 -r 20
```



```
opapaquery -o imageInfo -n 0x20000000d02
opapaquery -o imageInfo -y "1 hour ago"
opapaquery -o vfList
opapaquery -o vfInfo -V Default
opapaquery -o vfConfig -V Default
opapaquery -o vfPortCounters -l 1 -P 1 -d 1 -V Default
opapaquery -o clrVfPortCounters -l 1 -P 1 -s 0xC0000000 -V Default
#clears VLXmitData & VLRcvData
opapaquery -o vfFocusPorts -V Default -f 0x00030001 -S 0 -r 20
```

5.9.3 opashowmc

(Linux) Displays the Intel® Omni-Path Multicast groups created for the fabric along with the Intel® Omni-Path Host Fabric Interface (HFI) ports which are a member of each multicast group. This command can be helpful when attempting to analyze or debug Intel® Omni-Path multicast usage by applications or ULPs such as IPoIB.

Syntax

```
opashowmc [-v] [-t portsfile] [-p ports]
```

Options

- `--help` Produces full help text.
- `-v` Returns verbose output and shows name of each member.
- `-t portsfile` Specifies the file with list of local HFI ports used to access fabric(s) for analysis. Default is `/etc/opa/ports` file.
- `-p ports` Specifies the list of local HFI ports used to access fabric(s) for analysis.

Default is first active port. The first HFI in the system is 1. The first port on an HFI is 1. Uses the format `hfi:port`, for example:

 - `0:0` First active port in system.
 - `0:y` Port *y* within system.
 - `x:0` First active port on HFI *x*.
 - `x:y` HFI *x*, port *y*.

Examples

```
opashowmc
opashowmc -p '1:1 1:2 2:1 2:2'
```

Environment Variables

The following environment variables are also used by this command:



PORTS List of ports, used in absence of `-t` and `-p`.

PORTS_FILE File containing list of ports, used in absence of `-t` and `-p`.

5.10 FastFabric Utilities

The CLIs described in this section are used for miscellaneous information about the fabric. They are also available for custom scripting.

5.10.1 opa2rm

Permits the generation of configuration files for FastFabric or resource managers from a topology xml file.

When using a topology spreadsheet and `opaxlattopology` to design and prepare for deployment verification of a fabric, `opa2rm` may be used to generate resource manager configuration from the planned cluster design. Using this approach will allow the resulting configuration files to be complete, even if some nodes in the fabric have not yet been installed or made operational. Alternatively, `opareport -o topology` can be used to generate a topology XML file for input to `opa2rm`. In this case, only the currently present nodes will be included.

When working with SLURM, the `opa2rm -o slurm` option should typically be used. This option will generate a SLURM configuration file that lists the hosts directly connected to each switch in a syntax that can be used by SLURM's topology/tree plugin. It also generates a single "fake" switch shown as connecting all the other switches together. This approach allows for SLURM job placement to be improved while avoiding undo overhead in SLURM. This option also allows for topologies that are not a pure fat-tree.

When the configuration is a pure fat tree or oversubscribed fat tree, the `opa2rm -o slurmfat` option may be used to generate the full description of the fabric, including all intermediate and core switches in the fat tree topology. This option may permit better job placement optimization than the output from the `opa2rm -o slurm` option. However for larger fabrics, it may also increase the overhead within SLURM.

Syntax

```
opa2rm [-v] [-q] -o output [-g|-u|-t] [-F point] [-p prefix] [-s suffix]
topology_input
```

Options

<code>--help</code>	Produces full help text.
<code>-v/--verbose</code>	Specifies verbose output.
<code>-q/--quiet</code>	Disables progress reports.
<code>-o/--output output</code>	Specifies the output type:



<code>slurm</code>	SLURM tree nodes. Supports a variety of topologies.
<code>slurmful</code>	SLURM fat tree nodes and ISLs. Only supports pure trees.
<code>hosts</code>	FastFabric hosts file omitting this host
<code>-g/--guid</code>	Specifies the output switch GUIDs instead of names.
<code>-u/--underscore</code>	Changes spaces in switch names to underscores.
<code>-t/--trunc</code>	Truncates switch names at first space. This will treat large Director switches as a single, big switch. If <code>-g</code> , <code>-u</code> or <code>-t</code> are not specified, the switch name's suffix, after the first space, will be placed at the start of the name. For example, 'core5 Leaf 101' becomes 'Leaf101_core5'.
<code>-F/--focus point</code>	Specifies the focus area for output. Limits the scope of output to links that match any of the given focus points.
<code>-p/--prefix prefix</code>	Specifies the prefix to prepend to all FI hostnames.
<code>-s/--suffix suffix</code>	Specifies the suffix to append to all FI hostnames.
<code>topology_input</code>	Specifies the topology_input file to use. '-' may be used to specify stdin.

Point Syntax

<code>node:value</code>	<i>value</i> is node description (node name).
<code>node:value1:port:value2</code>	<i>value1</i> is node description (node name); <i>value2</i> is port number.
<code>nodepat:value</code>	<i>value</i> is glob pattern for node description (node name).
<code>nodepat:value1:port:value2</code>	<i>value1</i> is glob pattern for node description (node name); <i>value2</i> is port number.
<code>nodetype:value</code>	<i>value</i> is node type (SW, FI, or RT).
<code>nodetype:value1:port:value2</code>	<i>value1</i> is node type (SW, FI or RT); <i>value2</i> is port number.
<code>rate:value</code>	<i>value</i> is string for rate (25g, 50g, 75g, 100g).



<code>mtucap:value</code>	<code>value</code> is MTU size (2048, 4096, 8192, 10240); omits switch mgmt port 0.
<code>labelpat:value</code>	<code>value</code> is glob pattern for cable label.
<code>lengthpat:value</code>	<code>value</code> is glob pattern for cable length.
<code>cabledetpat:value</code>	<code>value</code> is glob pattern for cable details.
<code>linkdetpat:value</code>	<code>value</code> is glob pattern for link details.
<code>portdetpat:value</code>	<code>value</code> is glob pattern for port details to value.

Examples

```
opa2rm -o slurm topology.xml
opa2rm -o slurm -p 'opa-' topology.xml
opa2rm -o slurm -s '-opa' topology.xml
opa2rm -o slurm -F 'nodepat:compute*' -F 'nodepat:opacore1 *' topology.xml
opa2rm -o nodes -F 'nodedetpat:compute*' topology.xml
opa2rm -o hosts topology.xml
```

5.10.2 opaexpandfile

(Linux) Expands a Intel® Omni-Path Fabric Suite FastFabric hosts, chassis, or switches file. This tool expands and filter outs blank and commented lines. This can be useful when building other scripts that may use these files as input.

Syntax

```
opaexpandfile file
```

Options

`--help` Produces full help text.

file Specifies the FastFabric file to be processed.

Example

```
opaexpandfile allhosts
```

5.10.3 opafirmware

Returns firmware information.

Syntax

```
opafirmware [--showVersion | --showType] [firmwareFile]
```



Options

<code>--help</code>	Produces full help text.
<code>--showVersion</code>	Specifies the version of the firmware file.
<code>--showType</code>	Specifies the type of the firmware file.
<i>firmwareFile</i>	Specifies the firmware filename.

Examples

```
# opafirmware --showVersion STL1.q7.10.0.0.0.spkg
10.0.0.0
# opafirmware --showType STL1.q7.10.0.0.0.spkg
Omni_Path_Switch_Products.q7
```

5.10.4 opasorthosts

Sorts its standard input in a typical host name order and sorts to standard output. Hosts are sorted alphabetically (case-insensitively) by any alpha-numeric prefix, and then sorted numerically by any numeric suffix. Host names may end in a numeric field which may optionally have leading zeros. Unlike a pure alphabetic sort, this command results in intuitive sequencing of host names such as: host1, host2, host10.

This command does not remove duplicates; any duplicates are listed in adjacent lines.

Use this command to build `mpi_hosts` input files for applications or cable tests that place hosts in order by name.

Syntax

```
opasorthosts <hostlist> output_file
```

Options

<code>--help</code>	Produces full help text.
<i>hostlist</i>	Specifies the list of host names.
<i>output_file</i>	Specifies the sorted list output.

```
opasorthosts < host.xml > Sorted_host
```

Standard Input

```
opasorthosts
osd04
osd1
compute20
compute3
```



```
mgmt1
mgmt2
login
```

Standard Output

```
compute3
compute20
login
mgmt1
mgmt2
osd1
osd04
```

5.10.5 opaxmlextract

(Linux) Extracts element values from XML input and outputs the data in CSV format. `opaxmlextract` is intended to be used with `opareport`, to parse and filter its XML output, and to allow the filtered output to be imported into other tools such as spreadsheets and customer-written scripts. `opaxmlextract` can also be used with any well-formed XML stream to extract element values into a delimited format.

Five sample scripts are available as prototypes for customized scripts. They combine various calls to `opareport` with a call to `opaxmlextract` with commonly used parameters.

Syntax

```
opaxmlextract [-v] [-H] [-d delimiter] [-e extract_element]
               [-s suppress_element] [-X input_file] [-P param_file]
```

Options

<code>--help</code>	Produces full help text.
<code>-v/--verbose</code>	Produces verbose output. Includes output progress reports during extraction and output prepended wildcard characters on element names in output header record.
<code>-H/--noheader</code>	Does not output element name header record.
<code>-d/--delimit</code> <code><i>delimiter</i></code>	Uses single character or string as the delimiter between element names and element values. Default is semicolon.
<code>-e/--extract</code> <code><i>extract_element</i></code>	<p>Specifies the name of the XML element to extract. Elements can be nested in any order, but are output in the order specified. Elements can be specified multiple times, with a different attribute name or attribute value. An optional attribute (or attribute and value) can also be specified with elements:</p> <ul style="list-style-type: none">• <code>-e <i>element</i></code>• <code>-e <i>element:attrName</i></code>



- `-e element:attrName:attrValue`

- Notes:**
- Elements can be compound values separated by a dot. For example, `Switches.Node` is a `Node` element contained within a `Switches` element.
 - To output the attribute value as opposed to the element value, a specification such as `-e FIs.Node:id` can be used. This will return the value of the `id` attribute of any `Node` elements within `FIs` element.
 - If desired, a specific element can be selected by its attribute value, such as `-e MulticastFDB.Value:LID:0xc000` which will return the value of the `Value` element within `Multicast FDB` element where the `Value` element has an attribute of `LID` with a value of `0xc000`.
 - A given element can be specified multiple times each with a different `AttrName` or `attrValue`.

`-s/--suppress
suppress_element`

Specifies the name of the XML element to suppress extraction. Can be used multiple times (in any order). Supports the same syntax as `-e`.

`-X/--infile
input_file`

Parses XML from `input_file`.

`-P/--pfile
param_file`

Reads command parameters from `param_file`.

Example

Here is an example of `opareport` output filtered by `opaxmlextract`:

```
# opareport -o comps -s -x | opaxmlextract -d \; -e NodeDesc
-e SystemImageGUID -e NumPorts -s Neighbor
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Getting All Port Counters...
Done Getting All Port Counters
NodeDesc;SystemImageGUID;NumPorts
phs1fnivd13u07n4 hfi1_0;0x00117501016033c7;1
phs1fnivd13u07n2 hfi1_0;0x00117501016033ef;1
phs1fnivd13u07n1 hfi1_0;0x001175010160347a;1
phs1fnivd13u07n3 hfi1_0;0x0011750101603593;1
phs1swivd13u21;0x00117501ff6a5619;48
phs1fnivd13u07n1 hfi1_0;;
```

Details

`opaxmlextract` is a flexible and powerful tool to process an XML stream. The tool:



- Requires no specific element names to be present in the XML.
- Assumes no hierarchical relationship between elements.
- Allows extracted element values to be output in any order.
- Allows an element's value to be extracted only in the context of another specified element.
- Allows extraction to be suppressed during the scope of specified elements.

`opaxmlextract` takes the XML input stream from either stdin or a specified input file. `opaxmlextract` does not use or require a connection to a fabric.

`opaxmlextract` works from two lists of elements supplied as command line or input parameters. The first is a list of elements whose values are to be extracted, called extraction elements. The second is a list of elements for which extraction is to be suppressed, called suppression elements. When an extraction element is encountered and extraction is not suppressed, the value of the element is extracted for later output in an extraction record. An extraction record contains a value for all extraction elements, including those which have a null value.

When a suppression element is encountered, then no extraction is performed during the extent of that element, from start through end. Suppression is maintained for elements specified inside the suppression element, including elements which may happen to match extraction elements. Suppression can be used to prevent extraction in sections of XML that are present, but not of current interest. For example, `NodeDesc` or `NodeGUID` inside a `Neighbor` specification of `opareport`.

`opaxmlextract` attempts to generate extraction records with data values that are valid at the same time. Specifying extraction elements that are valid in the same scope produces a single record for each group of extraction elements. However, mixing extraction elements from different scopes (including different XML levels) may cause `opaxmlextract` to produce multiple records.

`opaxmlextract` outputs an extraction record under the following conditions:

- One or more extraction elements containing a non-null value go out of scope (that is, the element containing the extraction elements is ended) and a record containing the element values has not already been output.
- A new and different value is specified for an extraction element and an extraction record containing the previous value has not already been output.

Element names (extraction or suppression) can be made context-sensitive with an enclosing element name using the syntax `element1.element2`. In this case, `element2` is extracted (or extraction is suppressed) only when `element2` is enclosed by `element1`.

The syntax also allows '*' to be specified as a wildcard. In this case, `*.element3` specifies `element3` enclosed by any element or sequence of elements (for example, `element1.element3` or `element1.element2.element3`). Similarly, `element1.*.element3` specifies `element3` enclosed by `element1` with any number of (but at least 1) intermediate elements.

`opaxmlextract` prepends any entered element name not containing a '*' (anywhere) with '*. ', matching the element regardless of the enclosing elements.



Note: Any element names that include a wildcard should be quoted to the shell attempting to wildcard match against filenames.

At the beginning of operation, `opaxmlextract`, by default, outputs a delimited header record containing the names of the extraction elements. The order of the names is the same as specified on the command line and is the same order as that of the extraction record. Output of the header record can be disabled with the `-H` option. By default, element names are shown as they were entered on the command line. The `-v` option causes element names to be output as they are used during extraction, with any prepended wildcard characters.

Options (parameters) to `opaxmlextract` can be specified on the command line, with a parameter file, or using both methods. A parameter file is specified with `-P param_file`. When a parameter file specification is encountered on the command line, option processing on the command line is suspended, the parameter file is read and processed entirely, and then command line processing is resumed.

Option syntax within a parameter file is the same as on the command line. Multiple parameter file specifications can be made, on the command line or within other parameter files. At each point that a parameter file is specified, current option processing is suspended while the parameter file is processed, then resumed. Options are processed in the order they are encountered on the command line or in parameter files. A parameter file can be up to 8192 bytes in size and may contain up to 512 parameters.

5.10.6 `opaxmlfilter`

Processes an XML file and removes all specified XML tags. The remaining tags are output and indentation can also be reformatted. `opaxmlfilter` is the opposite of `opaxmlextract`.

Syntax

```
opaxmlfilter [-t|-k] [-l] [-i indent] [-s element] [input_file]
```

Options

- | | |
|-------------------------|---|
| <code>--help</code> | Produces full help text. |
| <code>-t</code> | Trims leading and trailing whitespace in tag contents. |
| <code>-k</code> | In tags with purely whitespace that contain newlines, keeps newlines as-is. Default is to format as an empty list. |
| <code>-l</code> | Adds comments with line numbers after each end tag. This can make comparison of resulting files easier since original line numbers are available. |
| <code>-i indent</code> | Sets indentation to use per level. Default is 4. |
| <code>-s element</code> | Specifies the name of the XML element to suppress. Can be used multiple times (maximum of 100) in any order. |



input_file Specifies the XML file to read. Default is `stdin`.

5.10.7 opaxmlindent

(Linux) Takes well-formed XML as input, filters out comments, and generates a uniformly-indented equivalent XML file. Use `opaxmlindent` to reformat files for easier reading and review, also to reformat a file for easy comparison with `diff`.

Syntax

```
opaxmlindent [-t|-k] [-i indent] [input_file]
```

Options

- `--help` Produces full help text.
- `-t` Trims leading and trailing whitespace in tag contents.
- `-k` In tags with purely whitespace that contain newlines, keeps newlines as-is. Default is to format as an empty list.
- `-i indent` Sets indentation to use per level. Default is 4.
- input_file* Specifies the XML file to read. Default is `stdin`.

5.10.8 opaxmlgenerate

(Linux) Takes comma-separated-values (CSV) data as input and generates sequences of XML containing user-specified element names and element values within start and end tag specifications. Use this tool to create an XML representation of fabric data from its CSV form.

Syntax

```
opaxmlgenerate [-v] [-d delimiter] [-i number] [-g element]  
[-h element] [-e element] [-X input_file] [-P param_file]
```

Options

- `--help` Produces full help text.
- `-g/--generate element` Generates value for *element* using value in next field from the input file. Can be used multiple times on the command line. Values are assigned to elements in order.
- `-h/--header element` Name of the XML element that is the enclosing header start tag.



<code>-e/--end element</code>	Name of the XML element that is the enclosing header end tag.
<code>-d/--delimit delimiter</code>	Specifies the delimiter character that separates values in the input file. Default is semicolon.
<code>-i/--indent number</code>	Number of spaces to indent each level of XML output. Default is 0.
<code>-X/--infile input_file</code>	Generates XML from CSV in <code>input_file</code> . One record per line with fields in each record separated by the specified delimiter.
<code>-P/--pfile param_file</code>	Uses input command line options (parameters) from <code>param_file</code> .
<code>-v/--verbose</code>	Produces verbose output. Includes output progress reports during extraction.

Details

`opaxmlgenerate` takes the CSV data from an input file. It generates fragments of XML, and in combination with a script, can be used to generate complete XML sequences. `opaxmlgenerate` does not use nor require a connection to an Intel® Omni-Path Fabric.

`opaxmlgenerate` reads CSV element values and applies element (tag) names to those values. The element names are supplied as command line options to the tool and constitute a template that is applied to the input.

Element names on the command line are of three (3) types, distinguished by their command line option - `Generate`, `Header`, and `Header_End`. The `Header` and `Header_End` types together constitute enclosing element types. Enclosing elements do not contain a value, but serve to separate and organize `Generate` elements.

`Generate` elements, along with a value from the CSV input file, cause XML in the form of `<element_name>value</element_name>` to be generated. `Generate` elements are normally the majority of the XML output since they specify elements containing the input values. `Header` elements cause an XML header start tag of the form: `<element_name>` to be generated. `Header_End` elements cause an XML header end tag of the form `</element_name>` to be generated. Output of enclosing elements is controlled entirely by the placement of those element types on the command line. `opaxmlgenerate` does **not** check for matching start and end tags or proper nesting of tags.

Options (parameters) to `opaxmlgenerate` can be specified on the command line, with a parameter file, or both. A parameter file is specified with `-P param_file`. When a parameter file specification is encountered on the command line, option processing on the command line is suspended, the parameter file is read and processed entirely, and then command line processing is resumed. Option syntax within a parameter file is the same as on the command line. Multiple parameter file specifications can be made, on the command line or within other parameter files. At each point that a parameter file is specified, current option processing is suspended

while the parameter file is processed, then resumed. Options are processed in the order they are encountered on the command line or in parameter files. A parameter file can be up to 8192 bytes in size and may contain up to 512 parameters.

Using opaxmlgenerate to Create Topology Input Files

opaxmlgenerate can be used to create scripts to translate from user-specific format into the opareport topology_input file format. opaxmlgenerate itself works against a CSV style file with one line per record. Given such a file it can produce hierarchical XML output of arbitrary complexity and depth.

The typical flow for a script which translates from a user-specific format into opareport topology_input would be:

- As needed, reorganize the data into link and node data CSV files, in a sequencing similar to that used by opareport topology_input. One link record per line in one temporary file, one node record per line in another temporary file and one SM per line in a third temporary file.
- The script must directly output the boilerplate for XML version, etc.
- opaxmlgenerate can be used to output the Link section of the topology_input, using the link record temporary file.
- opaxmlgenerate can be used to output the Node sections of the topology_input using the node record temporary file. If desired, there could be separate node record temporary files for HFIs, Switches, and Routers.
- opaxmlgenerate can be used to output the SM section of the topology_input, if desired.
- The script must directly output the closing XML tags to complete the topology_input file.

5.10.9 opacheckload

Returns load information on hosts in the fabric.

Syntax

```
opacheckload [-f hostfile] [-h 'hosts'] [-r] [-a|-n numprocs] [-d uploaddir]
```

Options

--help	Produces full help text.
-f <i>hostfile</i>	Specifies the file with hosts to check. Default = /etc/opa/hosts
-h <i>hosts</i>	Specifies the list of hosts to check.
-r	Reverses output to show the least busy hosts. Default is busiest hosts.
-n <i>numprocs</i>	Shows the specified number of top <i>numprocs</i> hosts. Default is 10.
-a	Shows all hosts. Default is 10.



`-d upload_dir` Specifies the target directory to upload loadavg. Default is uploads.

Examples

```
opacheckload
opacheckload -h 'arwen elrond'
HOSTS='arwen elrond' opacheckload
```

Environment Variables

The following environment variables are also used by this command:

HOSTS	List of hosts, used if <code>-h</code> option not supplied.
HOSTS_FILE	File containing list of hosts, used in absence of <code>-f</code> and <code>-h</code> .
UPLOADS_DIR	Directory to upload loadavg, used in absence of <code>-d</code> .
FF_MAX_PARALLEL	Maximum concurrent operations.



6.0 Performance Monitoring

FastFabric provides tools and methods for monitoring the performance of the fabric.

This section describes:

- The TUI menus used to gather Fabric Performance data
- What to do with the data you have gathered
- An overview of port counters and categories

6.1 Monitoring Fabric Performance

Both the FastFabric OPA Fabric Monitoring menu and `opatop` CLI allow you to start up the Fabric Performance Monitoring TUI so that you can monitor the performance of the fabric.

The Fabric Performance Monitor TUI displays performance, congestion, and statistics information about a fabric. Fabric information is divided into two main starting points for analyzing fabric traffic:

- **Performance** (bandwidth utilization): Can identify over-utilized areas (bottle necks) and under-utilized areas (potentially mis-configured).
- **Statistics**: Can identify problems in fabric hardware or configuration, as well as congestion and other performance situations.

6.1.1 Viewing the Fabric Performance Monitoring Summary Screen

The top-level Summary screen shows the basic fabric configuration information as well as performance and statistics information. This is the initial screen you see when you start up the TUI.

After looking at the Summary screen you can decide which area of the fabric (performance or statistics) and which port group or virtual fabric most warrants investigation, and can then drill down into that area.

To view the Fabric Performance Monitoring Summary screen, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter `opatop`.

The Summary screen is displayed.

```
opatop: Img: 10s @ Wed Sep 14 11:29:52 2016, Live
Summary: SW:      0 Ports: SW:      0 HFI:      2      Link:      1
          SM:      1 Node NRsp:      0 Skip:      0 Port NRsp:      0 Skip:      0
          AvgMBps  MinMBps  MaxMBps  AvgKpps  MinKpps  MaxKpps
0 All      Int      0        0        0        0        0        0
          Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min
1 HFIs     Int      0        0        0        0        0        0
          Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min
```




```

2 SWs          No ports in group

Master-SM: LID: 0x0001 Port: 1   Priority: 0   State: Master
          Name: phcppriv10 hfi1_0
          PortGUID: 0x0011750101575300
Secondary-SM: none

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help |
sS Pmcfg Imginfo View 0-n:

```

3. To change to the Virtual Fabrics (VF) Summary screen, type **v**.

The VF Summary screen is shown as in the example below.

```

opatop: Img: 10s @ Thu Sep 22 15:20:07 2016, Live
Summary:  SW:      0 Ports: SW:      0 HFI:      2      Link:      1
          SM:      1 Node NRsp:      0 Skip:      0 Port NRsp:      0 Skip:      0
          AvgMBps   MinMBps   MaxMBps   AvgKPps   MinKPps   MaxKPps
0 Admin      Int      0         0         0         0         0         0
          Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min

Master-SM: LID: 0x0001 Port: 1   Priority: 0   State: Master
          Name: phcppriv10 hfi1_0
          PortGUID: 0x0011750101575300
Secondary-SM: none

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help |
sS Pmcfg Imginfo View 0-n:

```

Summary Screen Field Descriptions

The table below describes the Summary screen field descriptions.

Table 15. Summary Screen Field Descriptions

Field	Description
Fabric Configuration Information	<p>Fabric configuration information includes</p> <ul style="list-style-type: none"> Numbers of links Numbers of switches (SW) Numbers of SMs Numbers of ports Master SM details Secondary SM details (if present)
Performance and Statistics for Each Port Group	<p>Fabric performance and statistics are presented based on port groupings and virtual fabrics grouping:</p> <p>For Port Groups:</p> <ul style="list-style-type: none"> All <ul style="list-style-type: none"> In the All group, all ports are Internal because, by definition, the neighbor port must be in the All group. HFIs

continued...



Field	Description
	<p>In the HFIs groups, all neighbor ports are outside the group, so statistics are contained in the Send and Receive subgroups.</p> <ul style="list-style-type: none">• SWs <p>In the SWs group, neighbor ports are either outside the group (HFI) or inside the group (another switch), so statistics are contained in all three subgroups. A special case for a switch port is the special switch port 0, which is always considered internal to the SWs group.</p> <p>For Virtual Fabrics Group:</p> <ul style="list-style-type: none">• Admin• Default <p>These groups provide a natural subdivision of the ports in a fabric for analysis. For more information about Groups and the operation of the PM, refer to the <i>Intel® Omni-Path Fabric Suite Fabric Manager User Guide</i>.</p> <p>For each group, the following statistics are reported:</p> <ul style="list-style-type: none">• Average MBps (megabytes per second)• Minimum MBps• Maximum MBps• Average KPPs (kilopackets per second)• Minimum KPPs• Maximum KPPs• Status indicator
Performance Utilization	<p>Performance Utilization for each port group is divided into up to three subgroups based on whether a port's neighbor port is in its group:</p> <ul style="list-style-type: none">• Internal <p>If a port's neighbor port is in its group, all performance statistics are contained in the Internal subgroup.</p> <ul style="list-style-type: none">• Send <p>If a port's neighbor is not in its group, statistics for data leaving the port (group) are contained in the Send subgroup</p> <ul style="list-style-type: none">• Receive <p>If a port's neighbor is not in its group, statistics for data entering the port are contained in the Receive subgroup.</p>
Statistics Categories	<p>The statistics categories are:</p> <ul style="list-style-type: none">• Integ – Integrity• Congst – Congestion• Bubble – Idles due to congestion• SmaCong – SMA Congestion• Secure – Security• Routing – Routing <p>Statistics categories are each based on one or more port counters. Each statistics category's status indicator is shown at one of five values/colors based on the category value as compared to a threshold value:</p> <ul style="list-style-type: none">• Minimum – green• Low – blue• Moderate – cyan• Warning – yellow• OVER – red

6.1.2 Viewing the PM Configuration

The PM Configuration screen displays information as provided by the PM.

Notes:

- The PM Configuration screen is the same for VF and non-VF.
- The PM Configuration screen has no screen-specific input commands.



To view PM Configuration, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opatop**.

The Summary screen is displayed.

3. Type **p**.

The PM Configuration screen is displayed as shown in the example below.

```

opatop: Img: 10s @ Thu Sep 22 15:23:17 2016, Live
PM Config:
  Sweep Interval: 10 sec  PM Flags(0x33):
    ProcessHFCntrs=On ProcessVLCntrs=On ClrDataCntrs=Off Clr64bitErrCntrs=Off
    Clr32bitErrCntrs=On Clr8bitErrCntrs=On
  Max Clients: 3
  Total Images: 10  Freeze Images: 5  Freeze Lease: 60 seconds
  Ctg Thresholds: Integrity: 100  Congestion: 100
                   SmaCongest: 100  Bubble: 100
                   Security: 10  Routing: 100
  Integrity Wts: Link Qual: 40  Uncorrectable: 100
                  Link Downed: 25  Rcv Errors: 100
                  Excs Bfr Ovrn: 100  FM Config Err: 100
                  Link Err Reco: 100  Loc Link Integ: 0
                  Lnk Wdth Dngd: 100
  Congest Wts: Cong Discards: 100  Rcv FECN: 5
                  Rcv BECN: 1  Mark FECN: 25
                  Xmit Time Cong 25  Xmit Wait: 10
  PM Memory Size: 169 MB (169295080 bytes)
  PMA MADs: MaxAttempts: 3  MinRespTimeout: 35  RespTimeout: 250
  Sweep: MaxParallelNodes: 10  PmaBatchSize: 2  ErrorClear: 7

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help |

```

4. Type **u** (lowercase) to return to the Summary Screen.

PM Configuration Screen Field Descriptions

For more information on field descriptions, refer to the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide*.

The table below describes the PM Configuration screen field descriptions.

Table 16. PM Configuration Field Descriptions

Field	Description
Sweep Interval	The time over the image data is relevant. Default is 10 seconds NOTE: Normally, the opatop interval should be set to a value \geq Sweep Interval.
PM Flags	Shows whether PM Flags are On or Off for: <ul style="list-style-type: none"> • ProcessHFCntrs • ProcessVLCntrs • ClrDataCntrs • Clr64bitErrCntrs • Clr32bitErrCntrs • Clr8bitErrCntrs
Max Clients	Maximum clients
Total Images	<ul style="list-style-type: none"> • Freeze Images • Freeze Lease time
<i>continued...</i>	



Field	Description
Ctg Thresholds	Category thresholds <ul style="list-style-type: none">• Integrity - Integrity• Congestion - Congestion• Bubble - Idles due to congestion• SmaCongest - SMA Congestion• Security - Security• Routing - Routing
Integrity Wts	Integrity weights <ul style="list-style-type: none">• Link Qual• Uncorrectable• Link Downed• Rcv Errors• Excs Bfr Ovrn• FM Config Err• Link Err Reco• Loc Link Integ• Lnk Wdth Dngd
Congest Wts	Congestion weights <ul style="list-style-type: none">• Cong Discards• Rcv FECN• Rcv BECN• Mark FECN• Xmit Time Cong• Xmit Wait
PM Memory Size	Size of the PM memory footprint in MB and bytes
PMA MADs	PMA MADs retry/timeout <ul style="list-style-type: none">• MaxAttempts• MinRespTimeout• RespTimeout
Sweep	Sweep information <ul style="list-style-type: none">• MaxParallelNodes• PmaBatchSize• ErrorClear

6.1.3 Viewing Image Information

The Image Information screen show the image information as provided by the PM.

Notes:

- The Image Information screen is the same for VF and non-VF.
- The PM Configuration screen has no screen-specific input commands.

To view Image Information, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opatop**.
The Summary screen is displayed.
3. Type **I**.



The Image Info screen is displayed as shown in the example below.

```

opatop: Img: IIs @ Day Month Date HR:MIN:SEC YYYY, Live
Image Inopatop: Img: 10s @ Thu Sep 22 16:51:58 2016, Live
Image Info:
Sweep Start: Thu Sep 22 16:51:58 2016
Sweep Duration: 0.001 Seconds
Image Interval: 10 Seconds

Num SW-Ports:      0  HFI-Ports:      2
Num SWs:           0  Num Links:      1  Num SMs:           1

Num NRsp Nodes:    0  Ports:          0  Unexpected Clear Ports: 0
Num Skip Nodes:    0  Ports:          0

      Master-SM: LID: 0x0001 Port: 1  Priority: 0  State: Master
                  Name: phcppriv10 hfi1_0
                  PortGUID: 0x0011750101575300
      Secondary-SM: none

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help |

```

4. Type **u** (lowercase) to return to the Summary Screen.

Image Information Screen Field Descriptions

The table below describes the Image Information screen field descriptions.

Table 17. Image Information Field Descriptions

Field	Description
Sweep Start	Timestamp for the start of the sweep
Sweep Duration	Length of time for the sweep
Image Interval	The time over the image data is relevant. Default is 10 seconds
Num [Ports]	Number of ports in each group: <ul style="list-style-type: none"> SW-Ports HFI-Ports
Num SWs	Number of switches
Node Information	Node information including: <ul style="list-style-type: none"> No response nodes Skipped nodes
Port Information	Port information including: <ul style="list-style-type: none"> No response ports Skipped ports Unexpected clear ports
SM Information	Master and secondary SM details <ul style="list-style-type: none"> LID Port Priority State Name PortGUID



6.1.4 Viewing Bandwidth Utilization

For each valid performance data subgroup, the Bandwidth Utilization screen displays the total, average, minimum, and maximum MBps and Kpps. For each subgroup, ten performance 'buckets' count the number of ports whose 'MBps compared to link rate' value corresponds to that bucket. This provides an indication of how the data rate of the group compares to its potential.

To view bandwidth utilization, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opatop**.
The Summary screen is displayed.
3. Determine which set of statistics you want to view:
 - To view Group information, continue to the next step.
 - To view VF information, type **v**.

4. Type the number for the specific group statistics that you want to view:

For Port Group:

- 0 – All
- 1 – HFIs
- 2 – SWs

For VF Group:

- 0 – Default
- 1 – Admin

The Info Select screen is displayed as shown in the example below.

```
opatop: Img: 10s @ Fri Sep 23 09:44:49 2016, Live
Group Info Sel: HFIs
Int NumPorts: 2 Rate Min: 100g Max: 100g
Ext NumPorts: 0
  Group Performance (P)
  Group Statistics (S)
  Group Config (C)

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | P S C:
```

5. Type **P**.

The Bandwidth (BW) Util screen is displayed as shown in the example below.

```
opatop: Img: 10s @ Fri Sep 23 09:46:09 2016, Live
Group BW Util: HFIs Criteria: Util-High Number: 10
Int: TotMBps AvgMBps MinMBps MaxMBps TotKpps AvgKpps MinKpps MaxKpps
      0      0      0      0      0      0      0      0
Buckt 0+% 10+% 20+% 30+% 40+% 50+% 60+% 70+% 80+% 90+%
      2      0      0      0      0      0      0      0      0
NoResp Int Ports: PMA: 0 Topo: 0

Int Congestion      Max      0+% 25+% 50+% 75+% 100+%
                  0      2      0      0      0      0
```



Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | cC N0-n Detail:

6. To set the BW stats Criteria for the focus query, type **c** (lowercase) to scroll forward or **C** (uppercase) to scroll in reverse to select one of the following choices:
 - Util-High – Bandwidth Utilization (highest first)
 - UtilPkt-Hi – Packet Utilization (highest first)
 - Util-Low – Bandwidth Utilization (lowest first)
7. To change the Number of entries in the BW stats list, type **N** and enter the target number of entries; then press **Enter**.
8. Type **D** to initiate the group focus query and access the detailed Group Focus screen (refer to [Viewing Focus Information](#) on page 309.)
9. Type **u** (lowercase) for each screen you've accessed until you are back to the screen you want.

Bandwidth Statistics Screen Field Descriptions

The table below describes the bandwidth screen field descriptions.

Table 18. Bandwidth Statistics Field Descriptions

Field	Description
Group Name	<p>Name of the group examined</p> <p>For Port Groups:</p> <ul style="list-style-type: none"> All In the All group, all ports are Internal because, by definition, the neighbor port must be in the All group. HFI's In the HFI's groups, all neighbor ports are outside the group, so statistics are contained in the Send and Receive subgroups. SWs In the SWs group, neighbor ports are either outside the group (HFI) or inside the group (another switch), so statistics are contained in all three subgroups. A special case for a switch port is the special switch port 0, which is always considered internal to the SWs group. <p>For Virtual Fabrics Group:</p> <ul style="list-style-type: none"> Admin Default
Criteria	<p>Focus criterion for Group Focus screen:</p> <ul style="list-style-type: none"> Util-High – Bandwidth Utilization (highest first) UtilPkt-Hi – Packet Utilization (highest first) Util-Low – Bandwidth Utilization (lowest first)
Number	Number of ports for a group focus query
Performance Data Subgroup	<p>Performance statistics for each port group are further divided into up to three subgroups based on whether a port's neighbor port is in its group:</p> <ul style="list-style-type: none"> Internal If a port's neighbor port is in its group, all performance statistics are contained in the Internal subgroup. Send

continued...



Field	Description
	<p>If a port's neighbor is not in its group, statistics for data leaving the port (group) are contained in the Send subgroup</p> <ul style="list-style-type: none">Receive <p>If a port's neighbor is not in its group, statistics for data entering the port are contained in the Receive subgroup.</p>
Statistics	<p>For each group, the following statistics are reported:</p> <ul style="list-style-type: none">Average MBpsMinimum MBpsMaximum MBpsAverage KppsMinimum KppsMaximum KppsStatus indicator
Performance Buckets	<p>Count the number of ports whose 'MBps compared to link rate' value corresponds to that bucket. This provides an indication of how the data rate of the group compares to its potential.</p> <p>Ten buckets from 0+% to 90+%, in 10% increments</p>
NoResp Ports	<p>No Response Ports per subgroup:</p> <ul style="list-style-type: none">PMA PMA failures are port counter query failures during the PM Sweep.Topo Topology errors are failures caused by encountering missing neighbor information in the topology.
Congestion buckets	<p>Provides context (from the Statistics Screen)</p> <ul style="list-style-type: none">Max0+%25+%50+%75+%100+%

6.1.5 Viewing Statistics Category

The Statistics Category screen displays statistics for a port group.

To view statistics category, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opatop**.
The Summary screen is displayed.
3. Determine which set of statistics you want to view:
 - To view Group information, continue to the next step.
 - To view VF information, type **v**.
4. Type the number for the specific group statistics that you want to view:
For Port Group:
 - 0 – All
 - 1 – HFIs
 - 2 – SWs



For VF Group:

- 0 – Default
- 1 – Admin

The Info Select screen is displayed as shown in the example below.

```
opatop: Img: 10s @ Fri Sep 23 09:44:49 2016, Live
Group Info Sel: HFIs
Int NumPorts: 2 Rate Min: 100g Max: 100g
Ext NumPorts: 0
  Group Performance (P)
  Group Statistics (S)
  Group Config (C)

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | P S C:
```

5. Type s.

The Category (Ctg) Stats screen is displayed as shown in the example below.

```
opatop: Img: 10s @ Fri Sep 23 11:55:09 2016, Live
Group Ctg Stats: HFIs Criteria: Integ Number: 10
Int      Max      0+%      25+%      50+%      75+%      100+%
Integrity 0         2         0         0         0         0
Congestion 0         2         0         0         0         0
SmaCongest 0         2         0         0         0         0
Bubble     0         2         0         0         0         0
Security    0         2         0         0         0         0
Routing     0         2         0         0         0         0
Utilization: 0.0% Discards: 0.0%

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | cC N0-n Detail:
```

- To set the category stats Criteria for the focus query, type **c** (lowercase) to scroll forward or **C** (uppercase) to scroll in reverse to select one of the following choices:
 - Integrity category (highest first)
 - Congestion category (highest first)
 - SmaCongestion category (highest first)
 - Bubble category (highest first)
 - Security category (highest first)
 - Routing category (highest first)
- To change the Number of entries in the Err Stats list, type **N** and enter the target number of entries; then press **Enter**.
- Type **D** to initiate the group focus query and access the detailed Group Focus screen (refer to [Viewing Focus Information](#) on page 309.)
- Type **u** (lowercase) for each screen you've accessed until you are back to the screen you want.



Statistics Screen Field Descriptions

The table below describes the bandwidth screen field descriptions.

Table 19. Statistics Field Descriptions

Field	Description
Group Name	<p>Name of the group examined</p> <p>For Port Groups:</p> <ul style="list-style-type: none"> All In the All group, all ports are Internal because, by definition, the neighbor port must be in the All group. All ports are Internal HFI In the HFIs groups, all neighbor ports are outside the group, so statistics are contained in the Send and Receive subgroups. All ports are External SWs In the SWs group, neighbor ports are either outside the group (HFI) or inside the group (another switch), so statistics are contained in all three subgroups. A special case for a switch port is the special switch port 0, which is always considered internal to the SWs group. Ports are Internal and External. <p>For Virtual Fabrics Group:</p> <ul style="list-style-type: none"> Admin Default
Criteria (Statistics Categories)	<p>Focus criteria/statistics categories:</p> <ul style="list-style-type: none"> Integrity <ul style="list-style-type: none"> Link Quality Indicator Link Width Downgrade Local Link Integrity Errors Port Receive Errors Excessive Buffer Overrun Errors (neighbor port) Link Error Recovery Link Downed Uncorrectable Errors FM Config Errors Congestion <ul style="list-style-type: none"> Port Transmit Wait Switch Port Congestion Port Receive FECN (neighbor port) Port Receive BECN (only from FIs) Port Transmit Time Congestion Port Mark FECN SmaCongestion The counters included in the SMA Congestion category are the VL 15 counters equivalent to the port counters in the Congestion category. Bubble <ul style="list-style-type: none"> Port Transmit Wasted Bandwidth Port Transmit Wait Data Port Receive Bubble (neighbor port) Security <ul style="list-style-type: none"> Port Receive Constraint Errors (neighbor port) Port Transmit Constraint Errors Routing <ul style="list-style-type: none"> Port Receive Switch Relay Errors

continued...



Field	Description
	The integrity and congestion error values are calculated by using a weighted sum. The weights for each and the threshold value for each error category can be seen in the PM Configuration screen (PM Configuration Screen Field Descriptions on page 299). For more details about how the values for each error category is composed, refer to the <i>Intel® Omni-Path Fabric Suite Fabric Manager User Guide</i> .
Number	Number of entries for a group focus query
Performance Data Subgroup	Performance statistics for each port group are further divided into up to three subgroups based on whether a port's neighbor port is in its group: <ul style="list-style-type: none"> Internal If a port's neighbor port is in its group, all performance statistics are contained in the Internal subgroup. Send If a port's neighbor is not in its group, statistics for data leaving the port (group) are contained in the Send subgroup Receive If a port's neighbor is not in its group, statistics for data entering the port are contained in the Receive subgroup.
Int or Ext	Location of the port in relation to the group. <ul style="list-style-type: none"> Int – The port's neighbor port is in its group (internal). Ext – The port's neighbor port is not in its group (external).
Category buckets	For each subgroup within a category, there are five histogram buckets. Each bucket has a width of 25% (0+%, 25+%, etc.) with the last bucket width for beyond the threshold (100+%). A bucket is used to measure the number of ports whose category value, when compared to the threshold, falls within the range of the bucket. This provides an indication of how counter rates compare to their thresholds. <ul style="list-style-type: none"> Max 0+% 25+% 50+% 75+% 100+%
Utilization	Percent of error utilization; aids congestion analysis.
Discards	Percent of errors discarded; aids congestion analysis.

6.1.6 Viewing Configuration Information

The Configuration screen displays a list of the ports in a group, including the LID, port number, port GUID, and NodeDesc for each.

To view configuration information, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opatop**.
The Summary screen is displayed.
3. Determine which set of statistics you want to view:
 - To view Group information, continue to the next step.
 - To view VF information, type **v**.
4. Type the number for the specific group statistics that you want to view:
For Port Group:



- 0 – All
- 1 – HFIs
- 2 – SWs

For VF Group:

- 0 – Default
- 1 – Admin

The Info Select screen is displayed as shown in the example below.

```
opatop: Img: 10s @ Fri Sep 23 09:44:49 2016, Live
Group Info Sel: HFIs
Int NumPorts: 2 Rate Min: 100g Max: 100g
Ext NumPorts: 0
  Group Performance (P)
  Group Statistics (S)
  Group Config (C)

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | P S C:
```

5. Type **C**.

The Config screen is displayed as shown in the example below.

```
opatop: Img: 10s @ Fri Sep 23 12:07:29 2016, Live
Group Config: HFIs NumPorts: 2
Ix LIDx Port Node GUID 0x NodeDesc
0 0001 1 0011750101575300 phcppriv10 hfil_0
1 0002 1 001175010157E443 phcppriv11 hfil_0

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | sS P0-n:
```

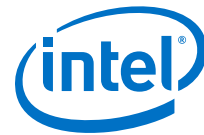
6. Type **s** (lowercase) to scroll forward or **S** (uppercase) to scroll backward through multiple screens of a long port list.
7. Type **P** and enter the target Ix number; then press **Enter** to view the Port Stats screen for the specified Ix (refer to [Viewing Port Statistics](#) on page 312).
8. Type **u** (lowercase) for each screen you've accessed until you are back to the screen you want.

Configuration Information Screen Field Descriptions

The table below describes the Configuration screen field descriptions.

Table 20. Configuration Information Field Descriptions

Field	Description
Group Name	Name of the group examined For Port Groups:
continued...	



Field	Description
	<ul style="list-style-type: none"> All In the All group, all ports are Internal because, by definition, the neighbor port must be in the All group. HFI In the HFIs groups, all neighbor ports are outside the group, so statistics are contained in the Send and Receive subgroups. SWs In the SWs group, neighbor ports are either outside the group (HFI) or inside the group (another switch), so statistics are contained in all three subgroups. A special case for a switch port is the special switch port 0, which is always considered internal to the SWs group. <p>For Virtual Fabrics Group:</p> <ul style="list-style-type: none"> Admin Default
NumPorts	Number of ports returned in the group configuration query
Ix	An index value that is used to select a port to view in the Port Stats screen.
LIDx	LID information
Port	Port Index
Node GUID 0x	Global Unique Identifier (GUID) for the Node
NodeDesc	Description of the node

6.1.7 Viewing Focus Information

The Focus information screen displays a list of the ports within a group, including the LID, port number, focus criterion, port GUID and NodeDesc of each. If the port has a neighbor port, the same information is displayed for the neighbor.

Note: The Focus information screen is the same for VF and non-VF.

To view focus information, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opatop**.
The Summary screen is displayed.
3. Determine which set of statistics you want to view:
 - To view Group information, continue to the next step.
 - To view VF information, type **V**.
4. Type the number for the specific group statistics that you want to view:

For Port Group:

 - 0 – All
 - 1 – HFIs
 - 2 – SWs

For VF Group:

 - 0 – Default
 - 1 – Admin



The Info Select screen is displayed.

5. Determine the Information Select menu to access:
 - To view the Focus information screen for BW Summary, type **B**.
 - To view the Focus information screen for Err Summary, type **S**.
6. Determine the Criteria for the focus query:
 - To set the BW stats Criteria for the focus query, type **c** (lowercase) to scroll forward or **C** (uppercase) to scroll in reverse to select one of the following choices:
 - Util-High – Bandwidth Utilization (highest first)
 - UtilPkt-Hi – Packet Utilization (highest first)
 - Util-Low – Bandwidth Utilization (lowest first)
 - To set the category stats Criteria for the focus query, type **c** (lowercase) to scroll forward or **C** (uppercase) to scroll in reverse to select one of the following choices:
 - Integrity category (highest first)
 - Congestion category (highest first)
 - SmaCongestion category (highest first)
 - Bubble category (highest first)
 - Security category (highest first)
 - Routing category (highest first)
7. Type **D**.

The Focus information screen is displayed as shown in the example below.

```
opatop: Img: 10s @ Fri Sep 23 13:03:09 2016, Live
Group Focus: HFIs GrpNumPorts: 2 NumPorts: 1 Number: 10
Ix Util-High LIDx Port Node GUID 0x NodeDesc
  0      0.0 0001   1 0011750101575300 phcppriv10 hfil_0
<->      0.0 0002   1 001175010157E443 phcppriv11 hfil_0
```

```
Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | sS cC NO-n PO-n:
```

8. To change the criteria after accessing this screen, type **c** (lowercase) to scroll forward or **C** (uppercase) to scroll in reverse to select one of the following choices:
 - Util-High – Bandwidth Utilization (highest first)
 - UtilPkt-Hi – Packet Utilization (highest first)
 - Util-Low – Bandwidth Utilization (lowest first)
 - Integrity category (highest first)
 - Congestion category (highest first)
 - SmaCongestion category (highest first)
 - Bubble category (highest first)



- Security category (highest first)
 - Routing category (highest first)
9. To change the Number of entries in the focus list, type **N** and enter the target number of entries; then press **Enter**.
 10. Type **s** (lowercase) to scroll forward or **S** (uppercase) to scroll backward through multiple screens of a long port list.
 11. Type **P** and enter the target Ix number; then press **Enter** to view the detailed Port Stats screen (refer to [Viewing Port Statistics](#) on page 312).
 12. Type **u** (lowercase) for each screen you've accessed until you are back to the screen you want.

Focus Information Screen Field Descriptions

The table below describes the Focus screen field descriptions.

Table 21. Focus Information Field Descriptions

Field	Description
Group Name	Name of the group examined For Port Groups: <ul style="list-style-type: none"> • All In the All group, all ports are Internal because, by definition, the neighbor port must be in the All group. • HFIs In the HFIs groups, all neighbor ports are outside the group, so statistics are contained in the Send and Receive subgroups. • SWs In the SWs group, neighbor ports are either outside the group (HFI) or inside the group (another switch), so statistics are contained in all three subgroups. A special case for a switch port is the special switch port 0, which is always considered internal to the SWs group. For Virtual Fabrics Group: <ul style="list-style-type: none"> • Admin • Default
GrpNumPorts	Number of ports selected, as determined by the combination of group, criteria, and requested ports
NumPorts	Number of ports returned in the group configuration query
Number	Number of ports for a group focus query
Ix	An index value that is used to select a port to view in the Port Stats screen.
Criteria	Limits the focus to specific port statistics For BW stats: <ul style="list-style-type: none"> • Util-High – Bandwidth Utilization (highest first) • UtilPkt-Hi – Packet Utilization (highest first) • Util-Low – Bandwidth Utilization (lowest first) For Err stats: <ul style="list-style-type: none"> • Integrity category (highest first) • Congestion category (highest first) • SmaCongestion category (highest first) • Bubble category (highest first) • Security category (highest first) • Routing category (highest first)
<i>continued...</i>	



Field	Description
LIDx	LID information
Port	Port Index NOTE: A symbol may be present on the first character of each line related to a port. This symbol is used to indicate a non-ideal condition was observed when calculating the relevant port's data. The possible conditions are, the PM was told to ignore this port ('~'), the PM failed to query this port ('!'), and the PM topology does not know this port's identity ('?').
Node GUID 0x	Global Unique Identifier (GUID) for the Node
NodeDesc	Description of the node

6.1.8 Viewing Port Statistics

The Port Statistics screen displays a specific port and LID's performance and statistics counters.

Note: The Port Statistics screen is the same for VF and non-VF.

To view port statistics, perform the following steps:

1. Log in to the server as root.
2. At the command prompt, enter **opatop**.
The Summary screen is displayed.
3. Determine which set of statistics you want to view:
 - To view Group information, continue to the next step.
 - To view VF information, type **V**.
4. Type the number for the specific group statistics that you want to view:
For Port Group:
 - 0 – All
 - 1 – HFIs
 - 2 – SWsFor VF Group:
 - 0 – Default
 - 1 – AdminThe Info Select screen is displayed.
5. Determine the Information Select menu to access:
 - To view the Port Stats screen for BW Summary, type **P**.
 - To view the Port Stats screen for Err Summary, type **S**.
 - To view the Port Stats screen for Configuration information, type **C**.
If you are accessing the Port Stats screen from the Configuration information screen, skip to Step 7.
6. Determine the Criteria for the focus query as described in [Viewing Bandwidth Utilization](#) on page 302 or [Viewing Statistics Category](#) on page 304.



7. Type **D** to access the Focus information screen.

To make changes to the Focus information prior to accessing the Port Stats screen, refer to [Viewing Focus Information](#) on page 309.

8. Type **P** and enter the target Ix number; then press **Enter** to view the detailed Port Stats screen.

The Port Stats screen is displayed.

Note: Neighbor port and link information is available only when access through the Focus Information screen. It is not available through the Configuration information screen.

```

opatop: Img: 10s @ Fri Sep 23 14:07:40 2016, Live
Port Stats: HFIs LID: 0x2 PortNum: 1 Rate: 100g MTU: 4096
NodeDesc: phcppriv11 hfil_0 NodeGUID: 0x001175010157E443
Neighbor: phcppriv10 hfil_0 LID: 0x1 PortNum: 1
Xmit: Data: 0 MB ( 63 Flits) Pkts: 1
Recv: Data: 0 MB ( 10 Flits) Pkts: 1
Multicast: Xmit Pkts: 0 Recv Pkts: 0
Integrity: | Congestion:
Link Quality: 5 | Cong Discards: 0
Uncorrectable: 0 | Rcv FECN*: 0
Link Downed: 0 | Rcv BECN: 0
Lanes Down: 0 | Mark FECN: 0
Rcv Errors: 0 | Xmit Time Cong: 0
Excs Bfr Ovrn*: 0 | Xmit Wait: 0
FM Conf Err: 0 | Routing and Others:
Lnk Err Recov: 0 | Rcv Sw Relay: 0
Loc Lnk Integ: 0 | Xmit Discards: 0
Security: | Bubble:
Xmit Constrain: 0 | Xmit Wasted BW: 0
Rcv Constrain*: 0 | Xmit Wait Data: 0
SmaCongestion (VL15): | Rcv Bubble*: 0
Cong Discards: 0
Xmit Wait: 0
Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help | Neighbor |

```

9. Type **N** to switch between statistics for the port and its neighbor port.
10. Type **u** (lowercase) for each screen you've accessed until you are back to the screen you want.

Port Statistics Screen Field Descriptions

The table below describes the Port Statistics screen field descriptions.

Table 22. Port Statistics Field Descriptions

Field	Description
Group Name	<p>Name of the group examined</p> <p>For Port Groups:</p> <ul style="list-style-type: none"> All <p>In the All group, all ports are Internal because, by definition, the neighbor port must be in the All group.</p> HFIs <p>In the HFIs groups, all neighbor ports are outside the group, so statistics are contained in the Send and Receive subgroups.</p> SWs

continued...



Field	Description
	<p>In the SWs group, neighbor ports are either outside the group (HFI) or inside the group (another switch), so statistics are contained in all three subgroups. A special case for a switch port is the special switch port 0, which is always considered internal to the SWs group.</p> <p>For Virtual Fabrics Group:</p> <ul style="list-style-type: none"> Admin Default
LIDx	LID information for the node
PortNum	Port number of the node
Rate	link rate
MTU	MTU, if available
NodeDesc	Description of the node
NodeGUID	Global Unique Identifier (GUID) for the Node
Neighbor	Description of the neighboring node
Xmit Data	Size of the data transmitted in MB and Flits and the number of packets
Recv Data	Size of the data received in MB and Flits and the number of packets
Multicast: Xmit Pkts	Number of multicast packets transmitted
Multicast: Recv Pkts	Number of multicast packets received
Statistics Counters	<ul style="list-style-type: none"> Integrity <ul style="list-style-type: none"> Link Quality Uncorrectable Link Downed Lanes Down Receive Errors Excessive Buffer Overrun* FM Config Errors Link Error Recovery Local Link Integrity Security <ul style="list-style-type: none"> Transmit Constraint Receive Constraint* SmaCongestion <p>The counters included in the SMA Congestion category are the VL 15 counters equivalent to the port counters in the Congestion category.</p> <ul style="list-style-type: none"> Cong Discards Xmit Wait Congestion <ul style="list-style-type: none"> Cong Discards Receive FECN* Receive BECN* Mark FECN Transmit Time Congestion Transmit Wait Routing and Others <ul style="list-style-type: none"> Receive Sw Relay Transmit Discards Bubble <ul style="list-style-type: none"> Transmit Wasted Bandwidth Transmit Wait Data
continued...	



Field	Description
	<p>— Receive Bubble*</p> <p>A trailing asterisk (*) on the counter name indicates the count will be used in computing Statistics Category information for the neighbor port.</p>

6.1.9 Navigating PM Sweeps

The Fabric Performance Monitoring TUI allows you to access statistics from sequential PM sweeps (the PM keeps a history of previous sweep images) and queries the PM at a user-specified interval (10 seconds by default). Sweeps are accessed from the short term history database being recorded by the PM. This allows access to statistics from up to 24 hours in the past.

When the Fabric Performance Monitoring TUI queries for statistics for the most recent PM sweep, it is in "Live" mode. In Live mode, the data will change, at the `opato` interval rate, as `opato` queries new PM sweeps. At each screen (summary or detail), the data being displayed is refreshed for the current PM sweep.

A PM sweep can be in "frozen" mode. The data in a frozen sweep will not change, allowing the statistics to be examined in summary and detail screens.

The Fabric Performance Monitoring TUI allows you to navigate the focus to another sweep within the history of sweeps maintained by the PM. For the duration of focus on such a sweep, it will remain frozen. You cannot navigate to any other screen while in "Historic" mode. Navigation can be performed for the screen in focus backward or forward, 1 or 5 sweeps at a time.

To navigate the historical PM sweeps, perform the following steps:

1. Navigate to the screen that you want to analyze historically.
2. Type `r` (lowercase) to go back one sweep at a time.

The date stamp below shows the time of the freeze (highlighted in bold) and the current on-going time (highlighted in italics).

```

opato: Img: 10s @ Fri Sep 23 17:32:32 2016, Hist Now: Fri Sep 23 17:33:08 2016
Port Stats: HFI: LID: 0x1 PortNum: 1 Rate: 100g MTU: 4096
NodeDesc: phcppriv10 hfil_0 NodeGUID: 0x0011750101575300
Neighbor: phcppriv11 hfil_0 LID: 0x2 PortNum: 1
Xmit: Data: 0 MB ( 10 Flits) Pkts: 1
Rcv: Data: 0 MB ( 63 Flits) Pkts: 1
Multicast: Xmit Pkts: 0 Rcv Pkts: 0
Integrity: | Congestion:
Link Quality: 5 | Cong Discards: 0
Uncorrectable: 0 | Rcv FECN*: 0
Link Downed: 0 | Rcv BECN: 0
Lanes Down: 0 | Mark FECN: 0
Rcv Errors: 0 | Xmit Time Cong: 0
Excs Bfr Ovrn*: 0 | Xmit Wait: 0
FM Conf Err: 0 | Routing and Others:
Lnk Err Recov: 0 | Rcv Sw Relay: 0
Loc Lnk Integ: 0 | Xmit Discards: 0
Security: | Bubble:
Xmit Constrain: 0 | Xmit Wasted BW: 0
Rcv Constrain*: 0 | Xmit Wait Data: 0
SmaCongestion (VL15): | Rcv Bubble*: 0
Cong Discards: 0
Xmit Wait: 0
Quit up Live/rRev/fFwd/bookmrkd Bookmrk Unbookmrk ?help | Neighbor |

```



3. Type **R** (uppercase) to go back five sweeps at a time.
4. Type **r** (lowercase) to move ahead one sweep at a time.
5. Type **R** (uppercase) to move ahead five sweeps at a time.
6. Type **L** to return to the Live data.

6.1.10 Bookmarking a Sweep

The Fabric Performance Monitoring TUI allows you to bookmark a sweep to review the information. For the duration of the Bookmark, all information is frozen. You can navigate through the various screens to review the frozen information. The sweep will remain frozen until you explicitly "Unbookmark" it.

Adding a Bookmark

Note: opatop allows only one sweep at a time to be bookmarked.

To bookmark a PM sweep, perform the following steps:

1. Navigate to the screen you want to capture and analyze.
2. Type **B** (uppercase) to bookmark the screen.

In the Image Identification line (line 1), the Live image changes to Bkmk (bookmark) as highlighted in bold in the example screen below.

```
opatop: Img: 10s @ Fri Sep 23 16:44:42 2016, Bkmk Now: Fri Sep 23 16:44:53 2016
Summary: SW:      0 Ports: SW:      0 HFI:      2      Link:      1
          SM:      1 Node NRsp:      0 Skip:      0 Port NRsp:      0 Skip:      0
          AvgMbps   MinMbps   MaxMbps   AvgKPps   MinKPps   MaxKPps
0 All          Int          0          0          0          0          0          0
  Integ:min Congst:min  SmaCong:min  Bubble:min  Secure:min  Routing:min
1 HFIs          Int          0          0          0          0          0          0
  Integ:min Congst:min  SmaCong:min  Bubble:min  Secure:min  Routing:min
2 SWs          No ports in group

Master-SM: LID: 0x0001 Port: 1  Priority: 0  State: Master
          Name: phcppriv10 hfil_0
          PortGUID: 0x0011750101575300
Secondary-SM: none

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help |
sS Pmcfg Imginfo View 0-n:
```

The bookmark will remain until you explicitly remove it.

3. Type **L** to return to the Live data.
4. Type **b** (lowercase) to return to the bookmarked image.

Removing a Bookmark

To remove a bookmark from a PM sweep, perform the following steps:

1. Type **b** (lowercase) to return to the bookmarked image.



2. Type **U** (uppercase).

In the Image Identification line (line 1), the Bkmk image changes back to Live (bookmark) as highlighted in bold in the example screen below.

```
opatop: Img: 10s @ Fri Sep 23 16:49:52 2016, Live
Summary: SW:      0 Ports: SW:      0 HFI:      2      Link:      1
         SM:      1 Node NRsp:      0 Skip:      0 Port NRsp:      0 Skip:      0
         AvgMBps  MinMBps  MaxMBps  AvgKpps  MinKpps  MaxKpps
0 All      Int      0      0      0      0      0      0
  Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min
1 HFIs     Int      0      0      0      0      0      0
  Integ:min Congst:min SmaCong:min Bubble:min Secure:min Routing:min
2 SWs      No ports in group

Master-SM: LID: 0x0001 Port: 1 Priority: 0 State: Master
          Name: phcppriv10 hfi1_0
          PortGUID: 0x0011750101575300
Secondary-SM: none

Quit up Live/rRev/fFwd/bookmrked Bookmrk Unbookmrk ?help |
sS Pmcfg Imginfo View 0-n:
```

6.1.11 Using the opatop Command Line Options

While opatop starts the Fabric Performance Monitoring TUI, you can use the command line options as shown below:

Syntax

```
opatop [-v] [-q] [-h hfi] [-p port] [-i seconds]
```

Options

<code>--help</code>	Produces full help text.
<code>-v/--verbose level</code>	Specifies the verbose output level. Value is additive and includes: <ul style="list-style-type: none"> 1 Screen 4 STDERR opatop 16 STDERR PaClient
<code>-q/--quiet</code>	Disables progress reports.
<code>-h/--hfi hfi</code>	Specifies the HFI, numbered 1..n. Using 0 specifies that the <code>-p port</code> port is a system-wide port number. (Default is 0.)



<code>-p/--port port</code>	Specifies the port, numbered 1..n. Using 0 specifies the first active port. (Default is 0.)
<code>-i/--interval seconds</code>	Interval in <i>seconds</i> at which PA queries are performed to refresh to the latest PA image. Default = 10 seconds.

-h and -p options permit a variety of selections:

- `-h 0` First active port in system (default).
- `-h 0 -p 0` First active port in system.
- `-h x` First active port on HFI x.
- `-h x -p 0` First active port on HFI x.
- `-h 0 -p y` Port y within system (no matter which ports are active).
- `-h x -p y` HFI x, port y.

6.2 Using Fabric Performance Data

This section provides information for what to do after you have gathered Fabric performance data.

6.2.1 Top Level Data

Top level data refers to the high-level perspective of the possible data you gathered from PA attributes. You can "drill down" to get information that is more specific, such as a list of ports.

6.2.1.1 Fabric Configuration and PM Image Information

From the PA, you can access fabric configuration and PM Image information. This data shows general information about the PM Image, including an unique 64-bit ID that you can use to access all the data collected for this PM Image.

The `ImageInfo` query can provide additional information such as:

- **Basic topology information** includes the number of HFI, Switch Nodes, and Ports. Also, you can view topology information about the Primary (Master) SM and the Secondary (Standby) SM, if present.
- **PM sweep data** includes the start time, sweep duration, and the time over which this Image is valid, as well as the number of ports and nodes that had failures and for which data was not gathered.

If You See No Response Nodes and No Response Ports

If you see No Response Nodes and No Response Ports, review the FM's Log to find out what failures are causing these ports to be unsuccessful.



Likely reasons for No Response Nodes and No Response Ports are timeouts due to port bounces or reboots. This happens because the PM sweep may already be underway and is using the most recently completed SM Sweep's topology data, which may not include the bounced port's new port state. If this only happens one time, it should be okay to ignore; but, if this is a transient or reoccurring issue, you will likely see that port or its neighbor appear to have integrity issues and should drill down and get more data on the offending ports.

If You See Unexpected Clears

If you see Unexpected Clears, review the FM's Log to find out what ports and what counters are being unexpectedly cleared.

CLI Tools such as `opapmaquery` and `opareport` can clear PMA counters and can trigger this, so check with other users first. Additionally, a reboot of the node may also reset the counters.

6.2.1.2 PM Port Group's Performance Utilization and Statistical Data

From the PA, you can access PM Port Group performance utilization and statistical data that provides conglomerated data of all the links within a PM Port Group.

A port's Performance data will fall within one of three subgroups based upon whether both (Internal subgroup), only itself (Send subgroup), or just its neighbor (Receive subgroup) is within the PM Port Group. The performance subgroup data has three subsections: the ten-bucket utilization percentage histogram, the performance statistics, and the no response ports counters.

The Statistical data available is divided into two subgroups: Internal and External. Each subgroup has the following subsections: a five-bucket histogram and a maximum value field for each of the six PA Categories.

If You See Ports in the Higher Percentage Buckets

If you see ports in the higher percentage buckets, it means that those ports are experiencing high values of that Category. The values for each bucket represent the number of ports that are "binned" within that percentage range (bucket).

If you see ports in the higher buckets for the Integrity category, then you will need to drill down further to find out what ports are experiencing Integrity issues. Note that reboots and general fabric maintenance (such as moving systems, replacing cables, etc.) can create false positives. You may want to verify if this issue recurs after a planned interruption is over before continuing to drill down to gather more data.

If you see ports in the Congestion higher percentage buckets, then you should check whether a node is being overloaded by the jobs running or by lack of allocated resources. Also, make sure you are using an appropriate Routing algorithm for your fabric. In a more serious situation, you may have to investigate the traffic pattern of the application, ISL resources, or over-subscription in the fabric. You can drill down to find out what ports are having this congestion and identify what resources are perhaps being over utilized and need to be redistributed.

If there are ports in the SMA Congestion higher percentage buckets, then you should check the SM and verify the configuration. SMA congestion is congestion specific to SM-only traffic and should happen only under extreme conditions.

If You See PMA or Topology No Response Ports

PMA No Response Ports are the same No Response Ports from the fabric configuration data, only limited in scope to the specific PM Port Group. PMA No Response ports are usually one-offs and follow the same steps as no response ports from fabric configuration (refer to section [If You See No Response Nodes and No Response Ports](#) on page 318).

Topology Incomplete ports are extremely rare. These are ports that should have had an active neighbor (all but Switch Port Zero), but do not. This is usually an indication that the SM has an inaccurate topology. Forcing an SM re-sweeps may clear this error if no other errors are occurring. Otherwise, you will have to drill down, find the Neighbor Port information, and manually bounce the link.

6.2.2 Mid-Tier Data

Mid-tier data contains the statistical information for the link-level perspective. The data provides a sorted list of links that you can use to drill down further to get each port's exact Port Counter values (if available). Links are sorted based upon criteria such as PA Category values and utilization metrics.

6.2.2.1 Sorted Lists of Links based upon Statistical Criteria

After choosing a criterion (Utilization or PA Category), a sorted list of links is formed, which is ordered by the value of the criterion for both ports of the link. Switch Port Zero evaluates as a Link with no neighbor.

Depending on the type of criterion, the number of offending ports, and their location in the fabric, several conditions are possible. While not always needed, more specific port counter values can be gathered at the per-port level when drilling down to the bottom tier.

If You See One Link With A Non-Zero Integrity Value

Having just one link with an integrity issue may have several causes.

One of the more common and benign causes is that the other side of the link bounced during the PM Sweep and appeared to be down (`LinkQualityIndicator = 0 [Down]`). This can also be seen in the No Response Ports values described in the previous sections, [If You See No Response Nodes and No Response Ports](#) on page 318 and [If You See PMA or Topology No Response Ports](#) on page 320. As this is usually a one-off, it can be ignored.

If this issue is not planned and is reoccurring, the port may be experiencing a Signal Integrity issue. Degradation of Signal Integrity may have several possible causes. A quick check of the cable's connections or the quality of the cable itself may be the likely solution.

Several FastFabric tools, such as `opalinkanalysis`, can help you identify links that may be misconfigured or operating at slower speeds. If one end of the link is attached to an HFI, you can try to access the node's syslog and see if the HFI is reporting any errors. In addition, you can "drill down" further to view the individual Port Counter values.



If You See Multiple Links With A Non-Zero Integrity Value

When multiple links have a non-zero value, you will need to determine their location in the fabric and group the links by proximity and purpose (compute, storage, etc.). For any links that cannot be grouped, you can follow the same process as described in [If You See One Link With A Non-Zero Integrity Value](#) on page 320. However, links that are in close proximity or share a similar purpose may have related causes and may require a slight change to the manner in which the shared issue is debugged.

An example would be if the group of links was attached to the same switch, then there may be an issue with the connections going to and from the switch or, more likely, the environment around the switch is contributing in some way (power, heating, etc.). In this case, you should first verify the switch's physical state is as expected. You can use a tool such as `opaswitchadmin` or `opachassisadmin` to gather data such as power supply status and temperature. If the group was all storage nodes, then perhaps the issue is related to the storage devices or software.

If You See One Link with a High Congestion Value

If you see a single link with a high congestion value and the link is an ISL, then you may need to investigate the application's design, configuration, and placement in the fabric. Next, you should check the over-subscription ratio and verify the topology of the fabric. Tools such as `opareport`, `opaextractmissinglinks`, and `opaxlattopology` can be used to verify the current topology against a predefined topology configuration file. If the link is attached to an HFI, you may need to alter what jobs are being run on that node and see if it may be overloaded. The Congestion value for a port is adjusted based upon the Utilization of the link to give a more accurate display of Congestion on the port.

However, if the link does not have a high Utilization but still has high Congestion, the link may be experiencing a more serious issue. You can "drill down" further to view the individual Port Counter values to better identify the issue.

6.2.3 Lowest Tier Data

PA Port Counters are the lowest tier of data available through the PA User interface.

6.2.3.1 Individual Port Counter Data

The lowest level in the PA is the Port Counters Data. Depending on the options of the request, response values can be a delta between itself and the previous Image or the RAW counter values obtained during a PM sweep. From these values, you can see the individual Port Counters that were used to compute the PA Categories at the higher levels.

Refer to the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide* that provides an appendix describing all of the counters, their categories, and how they are computed into each category. This document also includes the rationale behind the computation and inclusion of the counters in their respective categories.

6.3 Port Counters Overview

Each port in an Intel® Omni-Path Fabric maintains a set of port counters to indicate both traffic and error counts. These counters can be grouped into the categories described in this section. Each port stops incrementing when the max value is reached, irrespective of counter size. Most of the counters are 64-bits in size. Exceptions are noted.

6.3.1 Utilization

These counters reflect the normal utilization of the port and Virtual Lane when present.

Several of these counters are used during the calculation of Congestion, SMA Congestion, and the Bubble Categories. The Utilization metrics provide a way of giving some of the other counters context by comparing them to the amount of data or packets that were transmitted or received.

6.3.1.1 PortXmitData (TxD) and PortVLXmitData[n]

These counters indicate the total number of fabric packet flits transmitted. This does not include idle nor other LF command flits.

6.3.1.2 PortRcvData (RxD) and PortVLRcvData[n]

These counters indicate the total number of fabric packet flits received.

6.3.1.3 PortMulticastXmitPkts (MTxP)

This counter indicates the number of multicast and collective packets transmitted.

6.3.1.4 PortMulticastRcvPkts (MRxP)

This counter indicates the number of multicast and collective packets received.

6.3.2 Link Integrity

These counters reflect errors in the Physical (PHY) and Link Layers, as well as errors in firmware. In some cases, these errors are benign and can be ignored. However in other cases, excessive link integrity errors can indicate a hardware problem such as a poor connection, marginal cable, incorrect length/model cable for signal rate, or damaged/broken hardware, such as bad connectors.

When a bad packet is detected, one of these counters is incremented and the Link Layer may either discard or replay the packet.

During the link training sequence, assorted errors may be observed. This is a normal part of the link training and clock synchronization process. Hence, errors observed as part of rebooting nodes or moving cables should not be considered a problem.

The category is calculated as a weighted sum of the counters in the group. With the exception of ExcessiveBufferOverflowErrors, the counters in this group report on the receive side of the link. However, the counter can indicate a problem on either side of the link.



6.3.2.1 Link Quality Indicator (LQI)

This is a status indicator, similar to the signal strength bar display on a mobile phone, that enumerates link quality as a range of 0-5, with 5 being very good. Values in the lower part of the range may indicate hardware problems with components such as ports and cables that surface as signal integrity issues, leading to performance and other problems. The LQI gives you an instantaneous view of a link's quality on every hardware port.

Table 23. Link Quality Values and Description

Link Quality Value	Description
5	Working at or above preferred link quality, no action needed.
3	Working on low end of acceptable link quality, recommended corrective action on next maintenance window.
2	Working below acceptable link quality, recommend timely corrective action.
1	Working far below acceptable link quality, recommend immediate corrective action.
0	Link down

For more information on the counters used in determining a LQI value, refer to [Accessing Link Quality Indicator Values](#) on page 328 and the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide*.

6.3.2.2 LocalLinkIntegrityErrors (LLI) Counter

This counter indicates the number of retries initiated by a link transfer layer receiver.

The retry rate is represented by the Link Quality Indicator. A link that is meeting performance requirements has a Link Quality of 5, which corresponds to 1000 or fewer replays per second.

6.3.2.3 PortRcvErrors (RxE) Counter

This counter indicates the total number of packets containing an error that were received by the port, including Link Layer protocol violations and malformed packets. It indicates possible misconfiguration of a port, either by the SM or by user intervention. It can also indicate hardware issues or extremely poor link signal integrity.

6.3.2.4 ExcessiveBufferOverrunErrors (EBO) Counter

This counter, associated with credit management, indicates an input buffer overrun. It indicates possible misconfiguration of a port, either by the SM or by user intervention. It can also indicate hardware issues or extremely poor link signal integrity.

6.3.2.5 LinkErrorRecovery (LER) Counter

This counter indicates the number of times the link has successfully completed the link error recovery process.

Link Quality Indicator is the primary indicator for link quality to use. This counter is factored into the value reported for Link Quality Indicator. This counter may be non-zero for a properly functioning link.

6.3.2.6 LinkDowned (LD) Counter

This counter indicates the total number of times the port has failed the link error recovery process and downed the link. These events can cause disruptions to fabric traffic.

6.3.2.7 UncorrectableErrors (Unc) Counter

This counter indicates the number of unrecoverable device errors. This may indicate a defect in the reporting device.

6.3.2.8 FMConfigErrors Counter (FMC)

This counter reports inconsistent configurations of the low-level Subnet Management Agent (SMA) on either side of the link. It indicates possible misconfiguration of a port, either by the SM or by user intervention.

6.3.3 Congestion

These counters reflect possible errors that indicate traffic congestion in the fabric.

When congestion or a packet that has seen congestion is detected, one of these counters is incremented and then depending on the issue reported, the packet must wait. In an extreme case, the packet may time out and be dropped.

The category is calculated as a weighted sum of the counters in the context of the utilization counters. With the exception of PortRcvFECN, the counters are all reported on the transmit side of the link. In addition, PortRcvBECN is only taken if the local node is an HFI. However, the counter could indicate a problem on either side of the link.

6.3.3.1 CongDiscards (CD) Counter

Note: Formerly known as "SwPortCongestion".

This switch-only counter indicates the number of packets that were discarded as unable to transmit due to timeouts.

6.3.3.2 PortRcvFECN (RxF) Counter

When a device receives a packet with the Forward Explicit Congestion Notification (FECN) bit set to one, this counter is incremented.

6.3.3.3 PortRcvBECN (RxB) Counter

When a device receives a packet with the Backward Explicit Congestion Notification (BECN) bit set to one, this counter is incremented.

6.3.3.4 PortMarkFECN (MkF) Counter

This counter indicates the total number of packets that were marked Forward Explicit Congestion Notification (FECN) by the transmitter due to congestion.



6.3.3.5 PortXmitTimeCong (TxTC) Counter

This counter indicates the total number of *flit times* that the port was in a congested state for any data VL.

6.3.3.6 PortXmitWait (TxW) Counter

This counter indicates the amount of time (in *flit times*) any virtual lane had data but was unable to transmit due to no credits available.

6.3.4 SMA Congestion

These counters reflect congestion in the fabric specific to communication between the Subnet Manager and Subnet Manager Agents using the management VL (VL 15).

The category is calculated exactly as the Congestion category using the same weights and the correct VL15 utilization counters.

6.3.4.1 PortVLXmitWait[15] (VLTxW[15]) Counter

This counter behaves the same as PortXmitWait, but it is restricted to VL 15, which carries only SM traffic.

6.3.4.2 VLCongDiscards[15] (VLCD[15]) Counter

Note: Formerly known as "SwPortVLCongestion".

This counter behaves the same as CongDiscards, but it is restricted to VL 15, which carries only SM traffic.

6.3.4.3 PortVLRcvFECN[15] (VLRxF[15]) Counter

This counter behaves the same as PortRcvFECN, but it is restricted to VL 15, which carries only SM traffic.

6.3.4.4 PortVLRcvBECN[15] (VLRxB[15]) Counter

This counter behaves the same as PortRcvBECN, but it is restricted to VL 15, which carries only SM traffic.

6.3.4.5 PortVLXmitTimeCong[15] (VLTxTC[15]) Counter

This counter behaves the same as PortXmitTimeCong, but it is restricted to VL 15, which carries only SM traffic.

6.3.4.6 PortVLMarkFECN[15] (VLMkF[15]) Counter

This counter behaves the same as PortMarkFECN, but it is restricted to VL 15, which carries only SM traffic.

6.3.5 Bubble

These counters occur when an unexpected idle flit is transmitted or received.



The transmit port sends idle flits until it can continue sending the rest of the packet. The category is calculated as follows:

1. The maximum value between the sum of the XmitWastedBW and XmitWaitData or the neighbor's PortRcvBubble.
2. Then divide the previous value by the port's utilization to provide context.

6.3.5.1 PortXmitWastedBW (WBW) Counter

This counter indicates the number of *flit times* where one or more packets have been started but the transmitters are forced to send idles due to bubbles in the ingress stream. Also, the VLs that have data to be sent are not permitted to preempt the currently transmitting VL.

6.3.5.2 PortXmitWaitData (TxWD) Counter

This counter indicates the number of *flit times* where one or more packets have been started but interrupted due to bubbles in the ingress stream.

6.3.5.3 PortRcvBubble (RxBb) Counter

This counter indicates the total number of *flit times* where one or more packets have started to be received, but the receiver received idle flits from the wire.

6.3.6 Security

These counters reflect possible security problems in the fabric.

Security problems can occur if a PKey or SLID violation occurs at the port during the ingress or egress of a packet.

The category is calculated as the sum of the neighbor's PortRcvConstraintErrors and the local port's PortXmitConstraintErrors.

6.3.6.1 PortRcvConstraintErrors (RxCE)

This counter is incremented when partition key or source LID violations are detected in a received packet, indicating a possible security issue or misconfiguration of device security settings.

6.3.6.2 PortXmitConstraintErrors (TxCE)

This counter is incremented when partition key violations are detected in a packet attempting to be transmitted, indicating a possible security issue or misconfiguration of device security settings.

6.3.7 Routing

These counters reflect possible routing issues. When a routing issue occurs, the offending packet is dropped.

A typical cause of this error is the routing to a wrong egress port or an improper Service Channel (SC) mapping. These errors can be a side effect of a port or device going down while traffic was still in flight to or through the given port or device.



6.3.7.1 PortRcvSwitchRelayErrors (RxSR)

This counter indicates the number of packets that were dropped due to internal routing errors. It indicates possible misconfiguration of a switch by the SM.

6.3.8 Other

These counters do not fit into any of the previous categories.

6.3.8.1 PortRcvRemotePhysicalErrors (RxRP)

This counter indicates the number of downstream effects of signal integrity (SI) problems. It indicates an SI issue in the upstream path.

This counter was not included as it does not directly indicate the link that had the issue, so it can be misleading.

6.3.8.2 PortXmitDiscards (TxDc)

This counter indicates the number of packets dropped due to several reasons including timeouts and improper packet lengths.

Note: This counter is a super set that includes Congestion Discards counter.



7.0 FastFabric Diagnostics Capabilities

7.1 Overview

Many features are built into the Intel® Omni-Path Software to help diagnose fabric issues. For example, a link quality indicator gives you a quick way to monitor any link in the fabric, using the familiar signal strength bar display found on most mobile phones. Many tools found in the FastFabric focus on topology verification, which gives you a way to compare the current fabric configuration against the expected fabric configuration. Fabric anomalies can be analyzed to resolve issues. The tools can help you debug cabling, HFIs, switches, or configuration issues.

The ability to see various port-level information is included in FastFabric.

- You can view information such as vendor, model, length, technology, and date of the cables.
- Each port gives you a link down reason as well as the link down reason for the neighboring port. This includes providing LinkInit reasons as well as offline disabled reasons for each port in the fabric. This information gives more insight into the current state of a link and provides a reason the port is in this state.

For example, this information can be used to determine why an FM refused to bring up a certain link or why a link went down in the first place.

7.2 Accessing Link Quality Indicator Values

You can access the LQI values using several methods, including through the FastFabric OPA Fabric Monitoring tool, CLI commands, or Fabric Manager GUI. For more information on LQI values, refer to [Link Quality Indicator \(LQI\)](#) on page 323.

FastFabric Fabric Performance Monitoring TUI

You can view detailed information on LQI values using the FastFabric Fabric Performance Monitoring TUI.

Log into the FastFabric Fabric Performance Monitoring TUI using the instructions found in [Accessing the Fabric Performance Monitor](#) on page 32.

At the top-level Summary screen of `opatop`, you can monitor the overall fabric health. If errors are seen here, you can drill down into the sub-screens to see port-level information, such as the link quality, to gain insight into the problems.

A typical workflow might be:

- From the TUI Summary screen, you notice that there are integrity errors in the HFIs group.
- To investigate further, type `1` at the prompt.
- On the next screen, type `S` to view the group statistics.



- From this screen, if you notice Integrity errors that you want to investigate further, you can type **D** to view the detailed statistical information.
- From here, you can type **C** to sort the ports by different categories (such as, Integrity, Congestion, SmaCongestion) to find the ports that are showing issues.
- When sorted by Integrity, you can then type **P** and **X** (where X is the target Ix port number) to show more detailed port-level information, including the link quality indicator for that given port.

For additional information on each TUI screen, refer to [Monitoring Fabric Performance](#) on page 296.

CLI Command

You can see the LQI values using the commands:

- `opainfo`
- `opalinkanalysis errors`
- `opaextracterror`
- `opareport -o errors`

Fabric Manager GUI

You can view the LQI using the Fabric Manager GUI.

1. Launch the Fabric Manager GUI.
2. Go to the Performance tab.
LQI will display a 5-bar indicator.

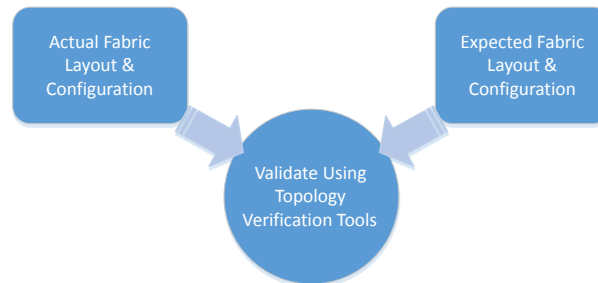
Refer to the *Intel® Omni-Path Fabric Suite Fabric Manager GUI User Guide* for more information.

7.3 Topology Verification

FastFabric provides several ways to assist you in validating a running fabric's configuration and layout against a predefined/expected topology. This verification process can help you to identify issues including detecting missing cables, hosts, or switches; or verifying that cables are in the correct places. You can run topology verification tools during the initial fabric startup or at any other time after a fabric is configured. You can also set up the Fabric Manager to verify the topology every time a sweep of the fabric is done (using the Fabric Manager's predefined topology verification feature).

Creating the Expected Fabric Layout File

The expected fabric layout is defined by an input file in XML format. Topology verification tools use this file along with the actual fabric configuration to analyze a fabric.



You use the `opaxlattopology` tool to translate a user-friendly, CSV-formatted file into the required XML format. To create the `topology.xml` file, perform the following:

1. Edit the sample `topology.xlsx` found in `/usr/share/opa/samples` folder to depict the expected layout of the fabric.
Refer to [topology.xlsx Overview](#) for more details.
2. Save the `topology.xlsx` as CSV format.
3. Run the `opaxlattopology` tool against the CSV file to get the expected fabric layout in XML format.

Refer to [opaxlattopology](#) for more details.

You can now use the `topology.xml` file to verify or validate against the actual fabric layout.

Validating a Topology Against an Actual Fabric Layout

`opareport` is one of the main tools provided by FastFabric used in the topology verification process. It provides various options to verify specific parts of the fabric. All these options require the expected `topology.xml` file as an input.

The table below summarizes the `opareport` options as well as the specific area of the fabric that they are used to verify.

Command	Verifies
<code>opareport -o verifyfis -T topology.xml</code>	HFI
<code>opareport -o verifyfsws -T topology.xml</code>	Switches
<code>opareport -o verifyfnodes -T topology.xml</code>	Nodes
<code>opareport -o verifyfsm -T topology.xml</code>	Nodes running SM
<code>opareport -o verifylinks -T topology.xml</code>	Links
<code>opareport -o verifyextlinks -T topology.xml</code>	Links External to a System
<code>opareport -o verifyfilinks -T topology.xml</code>	Links to HFI
<code>opareport -o verifyislink -T topology.xml</code>	Inter Switch Links
<code>opareport -o verifyextislink -T topology.xml</code>	Inter Switch Links, External to System
<code>opareport -o verifyall -T topology.xml</code>	All the Parameters



Topology Verification provides a detailed explanation to help interpret the `opareport` output for some of the output types above.

Refer to the following for additional verification details on:

- Using `opareport` for topology verification
- Other topology verification tools: `opaextractbadlinks`, `opaextractlink`, `opaextractmissinglinks`, `opaextractsellinks`, `opaextractstat2`, and `opalinkanalysis`
- `opafabricanalysis`: Used for fabric verification

Note: Fabric Manager also has a feature called "predefined topology verification" that can be used to detect dynamic changes that can occur in a fabric while the Fabric Manager is running. This feature also relies on an input XML topology file that describes nodes and links using a combination of Node GUIDs, Node Description, and Port number. This can be created by using `opaxlattopology` as explained above. When enabled, the predefined topology verification is done on every Fabric Manager sweep of the fabric. The Fabric Manager can be customized to define which fields are checked and the behavior that takes place when a topology mismatch is detected (that is, quarantine the offending node). Refer to the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide* for more details.

7.3.1 Interpreting Output of Topology Verification Tools

You can use the command `opareport -o verifyall -T topology.xml` to verify HFI, switches, SMs and links.

No Errors Detected

If no errors are detected in the topology, an output is shown similar to the example below:

```
[root@node057 topology]$ opareport -o verifyall -T topology.xml
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Done Getting vFabric Records
Parsing topology.xml...
FIs Topology Verification

FIs Found with incorrect configuration:
4 of 4 Fabric FIs Checked

FIs Expected but Missing or Duplicate in input:
4 of 4 Input FIs Checked

Total of 0 Incorrect FIs found
0 Missing, 0 Unexpected, 0 Duplicate, 0 Different
-----
SWs Topology Verification

SWs Found with incorrect configuration:
1 of 1 Fabric SWs Checked

SWs Expected but Missing or Duplicate in input:
1 of 1 Input SWs Checked

Total of 0 Incorrect SWs found
```



```
0 Missing, 0 Unexpected, 0 Duplicate, 0 Different
-----
SMs Topology Verification

SMs Found with incorrect configuration:
1 of 1 Fabric SMs Checked

SMs Expected but Missing or Duplicate in input:
1 of 1 Input SMs Checked

Total of 0 Incorrect SMs found
0 Missing, 0 Unexpected, 0 Duplicate, 0 Different
-----
Links Topology Verification

Links Found with incorrect configuration:
4 of 4 Fabric Links Checked

Links Expected but Missing, Duplicate in input or Incorrect:
4 of 4 Input Links Checked

Total of 0 Incorrect Links found
0 Missing, 0 Unexpected, 0 Misconnected, 0 Duplicate, 0 Different
-----
```

For each verification type run by the tool, the following attributes are reported:

- Missing
- Unexpected
- Misconnected
- Duplicate
- Different

A description for each of these attributes can be found in [Interpreting Health Check .changes Files](#) on page 151.

Possible Errors Detected

The following two examples show possible error conditions as well as an interpretation of the output that the `opareport` tool will report.

The example below shows an output of a verification where only one side of the expected links matches a port in fabric. No link exists in the topology file for "node059 hfi1_0". For this error condition, the tool can confidently match the other side of the link and can indicate that one side of the link is incorrect. Two issues are reported: one unexpected and one misconnected link.

```
[root@node057 topology]$ opareport -o verifyall -T topology.xml
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Done Getting vFabric Records
Parsing topology.xml...
FIs Topology Verification

FIs Found with incorrect configuration:
4 of 4 Fabric FIs Checked

FIs Expected but Missing or Duplicate in input:
4 of 4 Input FIs Checked
```



```

Total of 0 Incorrect FIs found
0 Missing, 0 Unexpected, 0 Duplicate, 0 Different
-----
SWs Topology Verification

SWs Found with incorrect configuration:
1 of 1 Fabric SWs Checked

SWs Expected but Missing or Duplicate in input:
1 of 1 Input SWs Checked

Total of 0 Incorrect SWs found
0 Missing, 0 Unexpected, 0 Duplicate, 0 Different
-----
SMs Topology Verification

SMs Found with incorrect configuration:
1 of 1 Fabric SMs Checked

SMs Expected but Missing or Duplicate in input:
1 of 1 Input SMs Checked

Total of 0 Incorrect SMs found
0 Missing, 0 Unexpected, 0 Duplicate, 0 Different
-----
Links Topology Verification

Links Found with incorrect configuration:
Rate NodeGUID      Port Type Name
100g 0x001175010265baf7 46 SW  Switchbaf7
<-> 0x0011758101660683 1 FI  node059 hf1l_0
Unexpected Link

4 of 4 Fabric Links Checked

Links Expected but Missing, Duplicate in input or Incorrect:
Rate MTU  NodeGUID      Port or PortGUID  Type Name
100g 8192      1
<->          46          FI login1 hf1l_0
Incorrect Link, 1st port found to be:
      0x0011758101660683 1 FI  node059 hf1l_0

4 of 4 Input Links Checked

Total of 2 Incorrect Links found
0 Missing, 1 Unexpected, 1 Misconnected, 0 Duplicate, 0 Different
-----

```

The example below shows a situation where both sides of an expected link match the same port in fabric. In this output, the tool indicates that either the input topology file has an issue (duplicate occurrence of the port) or the link is incorrectly cabled.

```

[root@node057 topology]$ opareport -o verifyall -T topology.xml
Getting All Node Records...
Done Getting All Node Records
Done Getting All Link Records
Done Getting All Cable Info Records
Done Getting All SM Info Records
Done Getting vFabric Records
Parsing topology.xml...
FIs Topology Verification

FIs Found with incorrect configuration:
4 of 4 Fabric FIs Checked

FIs Expected but Missing or Duplicate in input:
4 of 4 Input FIs Checked

```



```
Total of 0 Incorrect FIs found
0 Missing, 0 Unexpected, 0 Duplicate, 0 Different
-----
SWs Topology Verification

SWs Found with incorrect configuration:
1 of 1 Fabric SWs Checked

SWs Expected but Missing or Duplicate in input:
1 of 1 Input SWs Checked

Total of 0 Incorrect SWs found
0 Missing, 0 Unexpected, 0 Duplicate, 0 Different
-----
SMs Topology Verification

SMs Found with incorrect configuration:
1 of 1 Fabric SMs Checked

SMs Expected but Missing or Duplicate in input:
1 of 1 Input SMs Checked

Total of 0 Incorrect SMs found
0 Missing, 0 Unexpected, 0 Duplicate, 0 Different
-----
Links Topology Verification

Links Found with incorrect configuration:
Rate NodeGUID      Port Type Name
100g 0x001175010160357f  1 FI  node060 hfil_0
<-> 0x001175010265baf7 37 SW  Switchbaf7
Unexpected Link

4 of 4 Fabric Links Checked

Links Expected but Missing, Duplicate in input or Incorrect:
Rate MTU  NodeGUID      Port or PortGUID  Type Name
100g 8192      1                  FI node059 hfil_0
Duplicate Port in input or incorrectly cabled
<->          37                  SW Switchbaf7
Duplicate Port in input or incorrectly cabled
Duplicate Port in input or incorrectly cabled

4 of 4 Input Links Checked

Total of 2 Incorrect Links found
0 Missing, 1 Unexpected, 0 Misconnected, 1 Duplicate, 0 Different
-----
```

Note: The `opareport` verification tool reports all issues from multiple perspectives. Therefore, it may output incorrect links more than once. The tool does not try to match up issues or differentiate the source of duplicate errors. For example, the same error could be reported twice, once appearing as a mismatched port on an expected link, and again as an error looking like a cabling mistake.

7.4 Port Type Information

Ports contain data that describe their types and the properties of the cables connected to them. This includes physical information as well as metadata about the port/cable.

Ports have one of the following types:



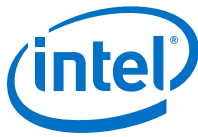
Port Type	Description
Disconnected	Part of design but physically unused
QSFP	Standard cable port
Fixed	Hardwired unchangeable port
Variable	Hardwired changeable port

Cable Information

A description of Cable info is shown in the table below. The **bold** entries are of most use for diagnostics, while the remaining entries may aid in debugging. The amount of information shown, that is, the number of fields, can be controlled by a command line option to the tools.

Note: Different output levels may show the information with slightly altered labels.

Cable Info Item	Description
Identifier	Identifier type of cable
Power Class	Power Consumption Class: 0 --- Power Class 1 (1.5 W max) 1 --- Power Class 2 (2.0 W max) 2 --- Power Class 3 (2.5 W max) 3 --- Power Class 4 (3.5 W max)
TXCDRSupported	TX CDR present
RxCDRSupported	RX CDR present
Connector	Connector type code
NominalBR	Nominal supported bit rate
OM2Length	Supported OM2 fiber length
OM3Length	Supported OM3 fiber length
OM4Length	Supported OM4 fiber length
DeviceTech	Cable type description
VendorName	Vendor name
VendorOUI	Vendor OUI
VendorPN	Vendor part number
VendorRev	Vendor revision
MaxCaseTemp	Max case temperature in degrees Celsius
CC_BASE	Checksum
TxInpEqAutoAdp	TX input EQ auto-adaptive capable
TxInpEqFixProg	TX input EQ fixed programmable capable
RxOutpEmphFixProg	RX output emphasis fixed programmable capable
RxOutpAmpFixProg	RX Output amplitude fixed programmable capable
TxCDRonOffCtrl	TX CDR on/off control implemented
RxCDRonOffCtrl	RX CDR on/off control implemented
<i>continued...</i>	



Cable Info Item	Description
TxSquelchImp	TX Squelch implemented
MemPage02Provided	Memory page 02 provided
MemPage01Provided	Memory page 01 provided
VendorSN	Vendor serial number
DateCode	Vendor manufacturing date code
CC_EXT	Checksum
CertCableFlag	OPA certified cable
ReachClass	Vendor specific field
CertDataRates	OPA certified data rate

Port and Cable Verification Tools

In addition to `opareport`, several lower-level tools are useful for verifying port information and settings. These tools also allow you to view the properties for a cable in a link. These tools include:

- `opaportinfo`
- `opainfo`
- `opasaquery`
- `opasmaquery`

In some cases, the tools overlap with each other.

More specific information on each tool can be found in the *Intel® Omni-Path Fabric Host Software User Guide*.

`opasaquery` allows you to query the Fabric Manager for its internal information. `opaportinfo`, `opainfo`, and `opasmaquery` can be used to query a port directly. `opainfo` provides information about local ports, whereas the other tools provide information about any port in the fabric.

For example:

- `opainfo -d4`: Provides basic port information along with detailed cable information for local port
- `opaportinfo -l2`: Provides port info for port with lid 2
- `opasmaquery -o portinfo`: Provides local port information
- `opasmaquery -o cableinfo -d4`: Provides local port cable information in detail
- `opasaquery -o portinfo`: Provides port information for all ports in fabric
- `opasaquery -o cableinfo`: Provides cable info

Note: Because `opareport` shows port names instead of LIDs, it may be more useful than using `opasaquery`.



For more information on Node Performance, refer to the *Intel® Omni-Path Fabric Suite Fabric Manager User Guide*.

7.5 Link Down Reason

In order for two link partners to communicate reliably, a set of link states is defined to identify the ability of the link to move management or data traffic. Two indicators provide information about the reason a link went down:

- **LinkDownReason:** The reason the local port initiated a *LinkDown* from either the *LinkInit*, *LinkArmed*, or *LinkActive* state. It only captures the first reason for why the link is down (if more than one reason before the indicator is cleared).
- **NeighborLinkDownReason:** The value received from the neighbor.

Note: The SM is in charge of clearing both values to permit subsequent reasons to be recorded. The SM clears both these values as part of bringing the link to Armed.

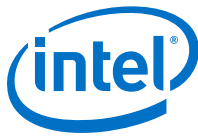
The `opasaquery` tool can be used to show both the *LinkDownReason* and the *NeighborLinkDownReason*: `opasaquery -o portinfo`.

You can also look at the *LinkDownErrorLog*, which stores the last eight historical reasons for why the port went down, using: `opasaquery -o portinfo -vvv`.

Other tools showing the *LinkDownReason* are `opaportinfo` and `opasmaquery`.

The table below describes the *LinkDownReason* values:

Value	Description
0: None	
<i>Corresponding to locally initiated link bounce due to PortErrorAction</i>	
2: Bad Packet Length	Illegal packet length in header
3: Packet Too Long	Packet longer than length
4: Packet Too Short	Packet shorter than length with normal tail
5: Bad source LID	Illegal SLID (0, using multicast as SLID. Does not include security validation of SLID)
6: Bad destination LID	Illegal DLID (0, does not match HFI, multicast DLID on SC15)
7: Bad L2	Illegal L2 opcode
8: Bad SC	Unconfigured SC
10: Bad Mid Tail	Body/Tail received without a corresponding Head flit
12: Preempt Error	Preempting with same VL
13: Preempt VL15	Preempting a VL15 packet
14: Bad VL Marker	
17: Bad Head Distance	Distance violation between two head flits
18: Bad Tail Distance	Distance violation between two tail flits
19: Bad Control Distance	Distance violation between two credit LF command flits
20: Bad Credit Ack	Credits return for unsupported VL
<i>continued...</i>	



Value	Description
21: Unsupported VL Marker	
22: Bad Preempt	Exceeding the interleaving level
23: Bad Control Flit	Unknown or reserved control flit received—deprecated
24: Exceed Multicast Limit	
32: Excessive Buffer Overrun	
<i>Corresponding to local initiated intentional link down</i>	
33: Unknown	
35: Reboot	Reboot or service reset
36: Neighbor Unknown	Link down was not locally initiated but no <i>LinkGoingDown</i> idle flit was received
39: FM Bounce	FM initiated bounce by transitioning from <i>LinkUp</i> to <i>Polling</i> .
40: Speed Policy	Link outside link policy
41: Width Policy	Link downgrade outside policy
<i>Corresponding to local initiated intentional link down via transition to Offline or Disabled</i>	
49: Disconnected	Link can never reach <i>LinkUp</i>
50: No Local Media Installed	Module is not installed in local port connector
51: Not Installed	Internal link not installed, due to absence of link partner FRU o backplane
52: Chassis Config	Chassis management forcing port <i>Offline</i> due to incompatible or absent link partner FRU or backplane
54: End to End not Installed	Silicon photonics mid-board module installed, but unable to detect link partner silicon photonics, due to absence of some part of the optical interconnect or absence of the remote module
56: Power Policy	Unable to enable port without exceeding power policy
57: Link Speed Policy	Link Speed Enabled policy Is not able to be met due to a persistent cause
58: Link Width Policy	Link Width Enabled policy is not able to be met due to a persistent cause such as board design having insufficient lanes. Does not include dynamic reasons such as failed link negotiation or <i>LinkWidthDowngrade</i> below policy
60: Switch Management	User disabled via switch management interface (CLI, SNMP, GUI, Config file, etc.)
61: SMA Disabled	User disabled via SMA packet changing Physics Port State to <i>Disabled</i>
63: Transient	Port recently entered <i>Offline</i> and is waiting for a Timeout to ensure synchronization with link partner Physics Port State machine.

If the link is currently down, the *LinkDownReason* and *LinkDownErrorLog* will not be available. It will be populated when the link comes back up.

In many cases, the *LinkDownReason* will be the same as the neighbor ports value of *NeighborLinkDownReason* and vice versa. However, there are exceptions. The following table shows the *LinkDownReasons* that are only applicable to *LinkDownReason* and will not be used for *NeighborLinkDownReason*.



36	Neighbor Unknown
49	Disconnected
50	No Local Media Installed
51	Not Installed
54	End to End not Installed

Below are some sample combinations of values for a configuration with Device A connected to Device B:

- For a link down initiated by device A and device A is able to send the reason:
A.LinkDownReason=X, A.NeighborLinkDownReason=0; B.LinkDownReason=0, B.NeighborLinkDownReason=X
- For a link down initiated by device A and B concurrently, where one of the devices is able to send the reason to the other device:
A.LinkDownReason=X, A.NeighborLinkDownReason=Y; B.LinkDownReason=Y, B.NeighborLinkDownReason=X
- For a link down initiated by device A and device A is unable to send reason to device B:
A.LinkDownReason=X, A.NeighborLinkDownReason=0; B.LinkDownReason=36 (Neighbor Unknown), B.NeighborLinkDownReason=0
- For an unexplained link down and device A or B is unable to send a reason code for the link going down (for example, cable failure):
A.LinkDownReason=36 (Neighbor Unknown), A.NeighborLinkDownReason=0; B.LinkDownReason=36 (Neighbor Unknown), B.NeighborLinkDownReason=0
- For a link down initiated by device A due to hard failure (for example, power loss, hard reset, ASIC fault, FW/driver crash, etc) and device A is unable to send reason to device B:
A.reason codes are inaccessible, powers back up as 0,0; B.LinkDownReason=36 (Neighbor Unknown), B.NeighborLinkDownReason=0

8.0 MPI Sample Applications

As part of a Intel® Omni-Path Fabric Suite FastFabric Toolset installation, sample MPI applications and benchmarks are installed in `/usr/src/opa/mpi_apps`. The sample applications can be used to perform basic tests and performance analysis of MPI, the servers, and the fabric.

The sample applications provided in the package include:

- Latency/bandwidth deviation test
- OSU latency (3 versions)
- OSU bandwidth (3 versions)
- OSU bidirectional bandwidth
- MPI stress test
- High Performance Linpack (HPL2)
- Intel® MPI Benchmarks (IMB)
- Pallas MPI Benchmarks (PMB)
- MPI fabric stress tests
- MPI batch run_* scripts

To tune the fabric for optimal performance, refer to *Intel® Omni-Path Fabric Performance Tuning User Guide*.

8.1 Building and Running Sample Applications

8.1.1 Building MPI Sample Applications

Perform the following procedure to build the applications:

1. Type `export MPICH_PREFIX=/usr/mpi/X/Y`

where:

- X is a compiler such as `gcc`
- Y is an MPI variation such as `openmpi-1.2.5`

Alternately, if you use the `mpi-selector` package to define which MPI you use, you can use the `get_selected_mpi.sh` script to do this for you by typing: `./usr/src/opa/mpi_apps/get_selected_mpi.sh`

This will show you the currently selected MPI and set the `MPICH_PREFIX` variable to match.

2. Type `cd /usr/src/opa/mpi_apps`
3. Type `make clean`



4. Type `make full` which builds all of the sample applications.

Note: The MPI used does not have to be in the `/usr/mpi` directory. The default MPIS installed with the Intel® OP Software are located here, however, you can also export `MPICH_PREFIX` to point to any location where you have another third party MPI installed.

The Intel® Omni-Path Fabric Suite FastFabric TUI can assist with building the MPI sample applications by providing a simple way to select the MPI to use for the build.

Alternatives include:

- `opa-base` - Builds applications in core RPM: Deviation, group stress, and `mpi_check`.
- `quick` - Builds everything in `opa-base`, plus OSU1 Latency, OSU1 Bandwidth, OSU2, OSU3.8, Intel® MPI Benchmarks (IMB), Deviation, HPL2, Group Stress
- `full` - Builds everything from `quick`.
- `all` - Builds everything from `full`.

8.1.2 Running MPI Sample Applications

To run the applications, an `mpi_hosts` file must be created in `/usr/src/opa/mpi_apps` that provides the names of the hosts on which processes should be run. Either IPoIB or Ethernet names can be specified. Typically, use of IPoIB names provides faster job startup, especially on larger clusters. These run scripts allow the `mpi_hosts` filename to be specified through the environment variable `MPI_HOSTS`. If this variable is not defined, the default `mpi_hosts` is used.

If a host has more than one real CPU, its name may appear in the MPI hosts file once per CPU.

Note: Intel® Xeon® Processors support hyper-threading; however, it significantly impacts performance for floating point intensive MPI applications, such as HPL2. For this reason, Intel recommends that you disable hyper-threading.

Note: When running the applications, all hosts listed in `MPI_HOSTS` must have a copy of the applications compiled for the same value of `MPICH_PREFIX`, for example, the same variation and version of MPI.

When the `run_*` scripts are used to execute the applications, the variation of MPI used to build the applications is detected and the proper `mpirun` is used to start the application.

To determine which variation of MPI the applications have been built, use the command:

```
cat /usr/src/opa/mpi_apps/.prefix
```

Note: Some variations of MPI may require that the MPD daemon be started prior to running applications. Consult the documentation on the specific variation of MPI for more information on how to start the MPD daemon.

When MPI applications are run with the `run_*` scripts provided, the results of the run are logged to a file in `/usr/src/opa/mapi_apps/logs`. The file name includes the date and time of the run for uniqueness.

The `run_*` scripts automatically use the `ofed.openmpi.params` or `ofed.mvapich2.params` files to set up parameters for `mpirun`. These files have various samples of setting parameters such as vFabric selection, dispersive routing, etc. These parameter files can also set the `MPI_CMD_ARGS` variable to provide additional arguments to `mpirun`.

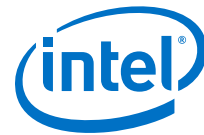
The current `run_*` scripts include:

- `run_allhfilatency` Checks the latencies of every pair of HFIs in the fabric.
- `run_alltoall3` Runs the OSU3 all-to-all benchmark.
- `run_batch_cabletest` Runs the `run_cabletest` script on every node in the fabric, but runs them in batches to reduce the load on the fabric.
- `run_bcast2` Runs OSU2 broadcast test.
- `run_bcast3` Runs OSU3 broadcast test.
- `run_bibw3` Runs OSU3 bidirectional bandwidth test.
- `run_bw` Runs OSU1 bandwidth test.
- `run_bw2` Runs OSU2 bandwidth test.
- `run_bw3` Runs OSU3 bandwidth test.
- `run_cabletest` Stresses groups of nodes in the fabric to discover possible bad cables.
- `run_deviation` Runs the deviation test.
- `run_hpl2` Runs HPL V2.
- `run_imb` Runs Intel® MPI Benchmarks (IMB).
- `run_lat` Runs OSU1 latency test.
- `run_lat2` Runs OSU2 latency test.
- `run_lat3` Runs OSU3 latency test.
- `run_mbw_mr3` Runs OSU3 `mbw_mr` test (multibandwidth message rate test).
- `run_mpicheck` Simple test to validate that MPI is passing data correctly.
- `run_multi_lat3` Runs OSU3 multi-latency test.
- `run_multibw` Runs OSU1 multi-bandwidth test.

8.2 Latency/Bandwidth Deviation Test

This is an analysis/diagnostic tool to perform assorted pairwise bandwidth and latency tests and report pairs outside an acceptable tolerance range. The tool identifies specific nodes that have problems and provides a concise summary of results.

This tool is also used by the Intel® Omni-Path Fabric Suite FastFabric Toolset Check MPI performance TUI menu item. It can also be invoked using `opahost mpipefdeviation`.



Perform the following procedure to use the script provided to run this application:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_deviation NP`

where:

NP is the number of processes to run or `all`, such as:

```
./run_deviation 4
```

This runs a quick latency and bandwidth test against pairs of the hosts specified in `mpi_hosts`. By default, each host is run against a single reference host and the results are analyzed. Pairs that have 20% less bandwidth or 50% more latency than the average pair are reported as failures.

Note: For this test, the `mpi_hosts` file should not list a given host more than once, regardless of how many CPUs the host has.

The tool can be run in a sequential or a concurrent mode. Sequential mode is the default and it runs each host against a reference host. By default, the reference host is selected based on the best performance from a quick test of the first 40 hosts.

In concurrent mode, hosts are paired up and all pairs are run concurrently. Since there may be fabric contention during such a run, any poor performing pairs are then rerun sequentially against the reference host.

Concurrent mode runs the tests in the shortest amount of time, however, the results could be slightly less accurate due to switch contention. In heavily oversubscribed fabric designs, if concurrent mode is producing unexpectedly low performance, try sequential mode.

`run_deviation` supports a number of parameters that allow for more precise control over the mode, benchmark and pass/fail criteria.

```
'ff'      When specified, the configured FF_DEVIATION_ARGS will be used
bwtol     Percent of bandwidth degradation allowed below Avg value
lattol    Percent of latency degradation allowed above Avg value

Other deviation arguments:
    [bwbidir] [bwunidir] [-bwdelta MBs] [-bwthres MBs] [-bwloop count]
[-bwsiz size] [-latdelta usec] [-latthres usec] [-latloop count] [-latsiz size]
[-c] [-b] [-v] [-vv] [-h reference_host]
-bwbidir  Perform a bidirectional bandwidth test
-bwunidir Perform a unidirectional bandwidth test (default)
-bwdelta  Limit in MB/s of bandwidth degradation allowed below Avg value
  -bwthres Lower Limit in MB/s of bandwidth allowed below Avg value
  -bwloop  Number of loops to execute each bandwidth test
  -bwsiz   Size of message to use for bandwidth test
  -latdelta Limit in usec of latency degradation allowed above Avg value
  -latthres Upper Limit in usec of latency allowed
  -latloop Number of loops to execute each latency test
  -latsiz  Size of message to use for latency test
  -c       Run test pairs concurrently instead of the default of sequential
  -b       When comparing results against tolerance and delta use best
           instead of Avg
  -v       verbose output
  -vv      Very verbose output
  -h       Baseline host to use for sequential pairing
Both bwtol and bwdelta must be exceeded to fail bandwidth test
When bwthres is supplied, bwtol and bwdelta are ignored
```



```
Both lattol and latdelta must be exceeded to fail latency test
When latthres is supplied, lattol and latdelta are ignored
```

```
For consistency with OSU benchmarks MB/s is defined as 1000000 bytes/s
```

```
Examples:
```

```
./run_deviation 20 ff
./run_deviation 20 ff -v
./run_deviation 20 20 50 -c
./run_deviation 20 '' '' -c -v -bwthres 1200.5 -latthres 3.5
./run_deviation 20 20 50 -c -h compute0001
./run_deviation 20 0 0 -bwdelta 200 -latdelta 0.5
```

```
Example of 4 hosts with both 20% bandwidth and latency tolerances running in
concurrent mode using the verbose option with a specified baseline host.
```

```
./run_deviation 4 20 20 -c -v -h hostname
```

8.3 OSU Tests

8.3.1 OSU Latency

This is a simple benchmark of end-to-end latency for various MPI message sizes. The values reported are one-direction latency.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_lat`

This runs assorted latencies from 0 to 256 bytes. To run a different set of message sizes, an optional argument specifying the maximum message size can be provided.

This benchmark only uses the first two nodes listed in `MPI_HOSTS`.

8.3.2 OSU Latency2

This is a simple performance test of end-to-end latency for various MPI message sizes. The values reported are one-direction latency.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_lat2`

This runs assorted latencies from 0 to 4 Megabytes.

This benchmark only uses the first two nodes listed in `MPI_HOSTS`.

8.3.3 OSU Latency 3

This is a simple performance test of end-to-end latency for various MPI message sizes. The values reported are one-direction latency.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`



2. Type `./run_lat3`

This runs assorted latencies from 0 to 4 Megabytes.

This benchmark only uses the first two nodes listed in `MPI_HOSTS`.

8.3.4 OSU Multi Latency3

This is a simple performance test of end-to-end latency for multiple concurrent pairs of hosts for various MPI message sizes. The values reported are average one-direction latency.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_multi_lat3 NP`

where:

`NP` is the number of processes to run or `all`, such as:

```
./run_multi_lat3 4
```

This runs assorted latencies from 0 to 4 Megabytes.

This benchmark only uses the first `NP` nodes listed in `MPI_HOSTS`.

8.3.5 OSU Bandwidth

This is a simple benchmark of maximum unidirectional bandwidth.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_bw`

This runs assorted bandwidths from 4K to 4Mbytes. To run a different set of message sizes, an optional argument specifying the maximum message size can be provided.

This benchmark only uses the first two nodes listed in `MPI_HOSTS`.

8.3.6 OSU Bandwidth2

This is a simple benchmark of maximum unidirectional bandwidth.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_bw2`

This runs assorted bandwidths from 1 byte to 4Mbytes.

This benchmark only uses the first two nodes listed in `MPI_HOSTS`.

8.3.7 OSU Bandwidth3

This is a simple benchmark of maximum unidirectional bandwidth.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_bw3`

This runs assorted bandwidths from 1 byte to 4Mbytes.

This benchmark only uses the first two nodes listed in `MPI_HOSTS`.

8.3.8 OSU Multi Bandwidth3

This is a simple benchmark of aggregate unidirectional bandwidth and messaging rate for multiple concurrent pairs of nodes.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_mbw_mr3 NP`

where:

`NP` is the number of processes to run or `all`, such as:

```
./run_mbw_mr3 4
```

This runs assorted messaging rates from 1 byte to 4Mbytes.

8.3.9 OSU Bidirectional Bandwidth

This is a simple benchmark of maximum bidirectional bandwidth.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_bibw2`

This runs assorted bandwidths from 1 byte to 4Mbytes.

This benchmark only uses the first two nodes listed in `MPI_HOSTS`.

8.3.10 OSU Bidirectional Bandwidth3

This is a simple benchmark of maximum bidirectional bandwidth.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_bibw3`

This runs assorted bandwidths from 1 byte to 4Mbytes.

This benchmark only uses the first two nodes listed in `MPI_HOSTS`.



8.3.11 OSU All to All 3

This is a simple benchmark of AllToAll latency.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_alltoall3 NP`

where:

NP is the number of processes to run or `all`, such as:

```
./run_alltoall3 4
```

This runs assorted latencies from 1 byte to 1Mbytes.

8.3.12 OSU Broadcast 3

This is a simple benchmark of Broadcast latency.

Perform the following steps:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_bcast3 NP`

where:

NP is the number of processes to run or `all`, such as:

```
./run_bcast3 4
```

This runs assorted latencies from 1 byte to 16K bytes.

8.3.13 OSU Multiple Bandwidth/Message Rate

The Multiple Bandwidth / Message Rate Test (`osu_mbw_mr`) is intended to be used with block assigned ranks. This means that all processes on the same machine are assigned ranks sequentially.

Note: All benchmarks are run using two processes, except for `osu_bcast` and `osu_mbw_mr` which can use more than two processes.

If you're using `mpd` with `MVAPICH2`, you must specify the number of processes on each host in the host file, otherwise `mpd` assigns ranks in a cyclic fashion. Refer to the following table for rank assignments.

Table 24. Rank Assignment

Rank	Block	Cyclic
0	host1	host1
1	host1	host2
2	host1	host1
<i>continued...</i>		



Rank	Block	Cyclic
3	host1	host2
4	host2	host1
5	host2	host2
6	host2	host1
7	host2	host2

Here is an example of MPD HOSTFILE:

```
host1:4
host2:4

MPI-1
-----
osu_bcast          - Broadcast Latency Test
osu_bibw           - Bidirectional Bandwidth Test
osu_bw            - Bandwidth Test
osu_latency        - Latency Test
osu_mbw_mr         - Multiple Bandwidth / Message Rate Test
osu_multi_lat      - Multi-pair Latency Test

MPI-2
-----
osu_acc_latency    - Accumulate Latency Test
osu_get_bw         - One-Sided Get Bandwidth Test
osu_get_latency    - One-Sided Get Latency Test
osu_latency_mt     - Multi-threaded Latency Test
osu_put_bibw       - One-Sided Put Bidirectional Test
osu_put_bw         - One-Sided Put Bandwidth Test
osu_put_latency    - One-Sided Put Latency Test
```

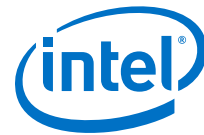
8.4 Latency Tests

The latency tests are carried out in a ping-pong fashion. The sender sends a message with a certain data size to the receiver and waits for a reply from the receiver. The receiver receives the message from the sender and sends back a reply with the same data size. Many iterations of this ping-pong test are carried out and average one-way latency numbers are obtained. Blocking version of MPI functions (MPI_Send and MPI_Recv) are used in the tests.

8.4.1 Multi-Threaded Latency Test

The multi-threaded latency test performs a ping-pong test with a single sender process and multiple threads on the receiving process. In this test, the sending process sends a message of a given data size to the receiver and waits for a reply from the receiver process. The receiving process has a variable number of receiving threads (set by default to 2), where each thread calls MPI_Recv and upon receiving a message sends back a response of equal size. Many iterations are performed and the average one-way latency numbers are reported.

Note: This test is only applicable for MVAPICH2 with threading support enabled.



8.4.2 Multi-Pair Latency Test

This test is very similar to the latency test, except that multiple pairs are performing the same test simultaneously. In order to perform the test across just two nodes, the hostnames must be specified in block fashion.

8.4.3 Broadcast Latency Test

This test is carried out in the following manner.

After doing an MPI_Bcast, the root node waits for an acknowledgment from the last receiver. This acknowledgment is in the form of a zero byte message from the receiver to the root. This test is carried out for a large number (1000) of iterations. The Broadcast latency is obtained by subtracting the time taken for the acknowledgment from the total time. The acknowledgment time is computed by doing a ping-pong test.

8.4.4 One-Sided Put Latency Test

The sender (origin process) calls MPI_Put (ping) to directly place a message of certain data size in the receiver window. The receiver (target process) calls MPI_Win_wait to make sure the message has been received. Then the receiver initiates a MPI_Put (pong) of the same data size to the sender, which is now waiting on a synchronization call. Several iterations of this test are carried out, and the average put latency is obtained.

Note: This test is only applicable for MVAPICH2.

8.4.5 One-Sided Get Latency Test

The origin process calls MPI_Get (ping) to directly fetch a message of certain data size from the target process window to its local window. It then waits on a synchronization call (MPI_Win_complete) for local completion. After the synchronization call, the target and origin processes are switched for the pong message. Several iterations of this test are carried out and the average get latency is obtained.

Note: This test is only applicable for MVAPICH2.

8.4.6 One-Sided Accumulate Latency Test

The origin process calls MPI_Accumulate to combine the data moved to the target process window with the data that resides at the remote window. The combining operation used in the test is MPI_SUM. The origin process then waits on a synchronization call (MPI_Win_complete) for local completion. After the synchronization call, the target and origin process are switched for the pong message. Several iterations of this test are carried out, and the average accumulate latency number is obtained.

Note: This test is only applicable for MVAPICH2.

8.5 Bandwidth Tests

The bandwidth tests are carried out by having the sender sending out a fixed number (equal to the window size) of back-to-back messages to the receiver and then waiting for a reply from the receiver. The receiver sends the reply only after receiving all

these messages. This process is repeated for several iterations and the bandwidth is calculated based on the elapsed time (from the time sender sends the first message until the time it receives the reply back from the receiver) and the number of bytes sent by the sender. The objective of these bandwidth tests is to determine the maximum sustained data rate that can be achieved at the network level. Non-blocking versions of MPI functions (MPI_Isend and MPI_Irecv) are used in the test.

8.5.1 Bidirectional Bandwidth Test

The bidirectional bandwidth test is similar to the bandwidth test, except that both nodes send out a fixed number of back-to-back messages and wait for the reply. This test measures the maximum sustainable aggregate bandwidth by two nodes.

8.5.2 Multiple Bandwidth / Message Rate Test

The multi-pair bandwidth and message rate test evaluates the aggregate uni-directional bandwidth and message rate between multiple pairs of processes. Each of the sending processes sends a fixed number of messages (the window size) back-to-back to the paired receiving process before waiting for a reply from the receiver. This process is repeated for several iterations. The objective of this benchmark is to determine the achieved bandwidth and message rate from one node to another node with a configurable number of processes running on each node.

8.5.3 One-Sided Put Bandwidth Test

The bandwidth tests are carried out by the origin process calling a fixed number of back-to-back Puts, and then waiting on a synchronization call (MPI_Win_complete) for completion. This process is repeated for several iterations, then the bandwidth is calculated, based on the elapsed time and the number of bytes sent by the origin process.

Note: This test is only applicable for MVAPICH2.

8.5.4 One-Sided Get Bandwidth Test

The bandwidth tests are carried out by an origin process calling a fixed number of back-to-back Gets, and then waiting on a synchronization call (MPI_Win_complete) for completion. This process is repeated for several iterations, then the bandwidth is calculated based on the elapsed time and the number of bytes sent by the origin process.

Note: This test is only applicable for MVAPICH2.

8.5.5 One-Sided Put Bidirectional Bandwidth Test

The bidirectional bandwidth test is similar to the bandwidth test, except that both nodes send out a fixed number of back-to-back Put messages and wait for their completion. This test measures the maximum sustainable aggregate bandwidth by two nodes.

Note: This test is only applicable for MVAPICH2.



8.6 mpi_stress Test

This test can be used to place stress on the interconnect as part of verifying stability. The `run_mpi_stress` script can be used to run this application.

This MPI stress test program is designed to load an MPI interconnect with point-to-point messages while optionally checking for data integrity. By default, it runs with all-to-all traffic patterns, optionally including you and your local peers. It can also be set up with multi-dimensional grid traffic patterns, and can be parameterized to run rings, open 2D grids, closed 2D grids, cubic lattices, hypercubes, and so forth. Optionally, the message data can be randomized and checked using CRC checksums (strong but slow), or XOR checksums (weak but fast). The communication kernel is built out of non-blocking point-to-point calls to load the interconnect. The program is not designed to exhaustively test different MPI primitives. Performance metrics are displayed, but may not be entirely accurate.

Usage

```
run_mpi_stress [number_processes] [mpi_stress arguments]
```

Options

mpi_stress arguments

- `-a INT` – desired alignment for buffers (must be power of 2)
- `-b BYTE` – byte value to initialize non-random send buffers (otherwise 0)
- `-c` – enable CRC checksums
- `-D INT` – set max data amount per msg size (default 1073741824)
- `-d` – enable data checksums (otherwise headers only)
- `-e` – exercise the interconnect with random length messages
- `-g INT` – use INT-dimensional grid connectivity (non-periodic)
- `-G INT` – use INT-dimensional grid connectivity (periodic) (default is to use all-to-all connectivity)
- `-h` – display this help page
- `-i` – include local ranks as destinations (only for all-to-all)
- `-I INT` – set msg size increment (default power of 2)
- `-l INT` – set min msg size (default 0)
- `-L INT` – set min msg count (default 100)
- `-m INT` – set max msg size (default 4194304)
- `-M INT` – set max msg count (default 10000)
- `-n INT` – number of times to repeat (default 1)
- `-O` – show options and parameters used for the run.
- `-p` – show progress
- `-P` – poison receive buffers at init and after each receive
- `-q` – quiet mode (don't show error details)



- `-r` – fill send buffers with random data (else 0 or `-b byte`)
- `-R` – round robin destinations (default is random selection)
- `-s` – include self as a destination (only for all-to-all)
- `-S` – use non-blocking synchronous sends (`MPI_Issend`)
- `-t INT` – run for INT minutes (implicitly adds `-n BIGNUM`)
- `-u` – uni-directional traffic (only for grid)
- `-v` – enable verbose mode (more `-v` for more verbose)
- `-w INT` – number of send/recv in window (default 20)
- `-x` – enable XOR checksums
- `-z` – enable typical options for data integrity (`-drx`) (for stronger integrity checking try using `-drc` instead)
- `-Z` – zero receive buffers at init and after each receive

8.7 High Performance Linpack (HPL2)

This test is a standard benchmark for Floating Point Linear Algebra performance. Version 2.0 is provided, which includes the Dr K. Goto Linear Algebra library. If desired, you can modify the HPL2 `makefiles` to use alternate libraries. Atlas source code and the open source math library are also provided in `/usr/src/opa/mpi_apps/ATLAS`. On RHEL systems, HPL2 attempts to use the Atlas package provided in the distribution.

Note: The Linear Algebra Library is highly optimized for a given CPU model. When running in a fabric with mixed CPU models, the HPL2 application must be rebuilt for each CPU model and that version must be used on all CPUs of the given type. Attempting to run a CPU with a library that is not optimized for the given CPU results in less than optimal performance. In some cases (such as trying to run an AMD CPU-optimized library on an Intel CPU), HPL2 may fail or produce incorrect results.

HPL2 is known to scale very well, and is the benchmark of choice for identifying a systems ranking in the Top 500 supercomputers (<http://www.top500.org>).

Prior to running this application, an `HPL.dat` file must be installed in `/usr/src/opa/mpi_apps/hpl2/bin/ICS.${ARCH}.${CC}` on all nodes. The `config_hpl2` script and some sample configurations are included.

The `config_hpl2` script can select from one of the assorted `HPL.dat` files in `opt/opa/src/mpi_apps/hpl-config`. These files are a good starting point for most clusters, and should get within 10-20% of the optimal performance for the cluster. The problem sizes used assume a cluster with 1GB of physical memory per processor. For each cluster size, 4 files are provided:

- `t` – A very small test run (5000 problem size)
- `s` – A small problem size on the low end of optimal problem sizes
- `m` – A medium problem size
- `l` – A large problem size



These can be selected using `config_hpl2`. The following command displays the pre-configured problem sizes available:

```
./config_hpl2
```

For example, to quickly confirm that HPL2 runs on the 16 nodes in the `/usr/src/opa/mpi_apps/mpi_hosts` file:

1. Type `./config_hpl2 16t`.

This command edits the `HPL.dat` file on the local host for a 16 host “very small” test, and copies that file to all hosts in the `mpi_hosts` file.

2. Once the `HPL.dat` has been configured and copied, HPL2 can be run using the script.

Type `cd /usr/src/opa/mpi_apps`

3. Type `./run_hpl2 NP`

where:

`NP` is the number of processors for the run, or `all`. For example:

```
./run_hpl2 16
```

For more information about HPL2, refer to the `README`, `TUNING`, and assorted HTML files in the `/usr/src/opa/mpi_apps/hpl2` directory.

8.8 Intel® MPI Benchmarks (IMB)

Use the `run_imb` sample script in `/usr/src/opa/mpi_apps` to run the Intel® MPI Benchmarks (IMB).

1. Type `cd /usr/src/opa/mpi_apps`

2. Type `./run_imb NP`

where:

`NP` is the number of processes to run, or `all`. A minimum of two processes is required. For example:

```
./run_imb 4
```

8.9 Pallas MPI Benchmark (PMB)

The Pallas MPI benchmark performs exhaustive benchmarking of latency and bandwidth for assorted message sizes for many MPI primitives. This benchmark is a good tool for evaluating and tuning small clusters, or a subset of a large cluster.

PMB has known scalability limitations, particularly in its **AllToAll** phase. This phase can simultaneously perform up to 4 MB transfers to and from all nodes at once. However, there is a downside in that a system must have approximately $10 \times NP$ MB of memory available per process for Pallas data to run this benchmark. Therefore, for a small cluster (approximately 16 processors or less), the memory requirement is modest at 160 MB. However, for a larger cluster (approximately 256 processors or greater), the memory requirement is rather large at 2.5 GB.



Intel recommends that you use PMB for smaller runs (2-32 processes), since the benchmark is likely to fail at larger process counts. Depending upon the amount of memory in the system and the numbers of processes to run, the `VIADEV_MEM_REG_MAX` parameter in `/usr/src/opa/mpi_apps/mpi.param.pallas` may need to be edited.

To run the benchmark:

1. Type `cd /usr/src/opa/mpi_apps`
2. Type `./run_pmb NP`

where:

`NP` is the number of processes to run, or `all`. For example:

```
./run_pmb 4
```

8.10 MPI Fabric Stress Tests

These sample applications are designed to stress parts of a cluster to help ensure that the fabric is working properly. Although they report measurement data similar to other bandwidth applications, they are not intended to be benchmarking tools. Instead, they should be used to identify potential performance issues in the fabric, such as bad cables.

8.10.1 All HFI Latency

The All HFI Latency test is a specialized stress test for large fabrics. It iterates through every possible pairing of the HFIs in the fabric, and performs a latency test on each pair. At the end of each combination, the test reports the fastest and slowest pairs. This test has no real value as a performance benchmark, but is extremely useful for checking for cabling problems in the fabric. A script is provided to run this application. It requires no arguments, but can take several options if needed. To run with no arguments, follow these steps:

1. Change directory to `/usr/src/opa/mpi_apps`.
`cd /usr/src/opa/mpi_apps`
2. Run the All HFI Latency test
`./run_allhfilatency`

This test runs a 60 second test on the first two nodes listed in the `mpi_hosts` file.

To change the default behavior, specify up to three optional arguments, for example:

```
./run_allhfilatency NP MN SS
```

where:

`NP` is the number of processes to run, or `all`.

`MN` is the number of minutes the test should run.

`SS` is the size of the messages to use when testing (between 1 byte and 4 megabytes).



For example, to run a 30 minute test on 64 nodes with 4 kilobyte messages, the following command would be used from the `/usr/src/opa/mpi_apps` directory:

```
./run_allhfilatency 64 30 4096
```

Once 30 minutes has elapsed, the test completes as soon as the current round of testing has completed.

If you want the tests to repeat indefinitely, use the duration `infinite` as shown in the following CLI command:

```
./run_allhfilatency 64 infinite 4096
```

There are three options, `-c`, `-h`, and `-v` available:

- `-h / --help` Provides some help text, then terminates.
- `-c / --csv` Prints all raw test results in CSV file format, into the application logfile. Useful for analyzing the raw results with a spreadsheet application.
- `-v / --verbose` Runs the test in a verbose mode that shows more information.

To use the results of this test, look for nodes that are often listed as the slowest at the end of the round. One of those nodes may have a cabling problem, or there may be a congested interswitch link causing those nodes to experience degraded performance.

8.10.2 run_cabletest

The `run_cabletest` tool is a specialized stress test for large fabrics. It groups MPI ranks into sets that are tested against other members of the set. This test has no real value as a performance benchmark, but is extremely useful for checking for cabling problems in the fabric.

`./run_cabletest` requires no arguments, but does require you to generate a group hosts file. This is done with the `gen_group_hosts` script. The name of the group hosts file is specified by the `$MPI_GROUP_HOSTS` variable, and defaults to `mpi_group_hosts`. For more information on `gen_group_hosts`, refer to [gen_group_hosts](#) on page 358.

By default, `run_cabletest` runs for 60 minutes and uses 4-megabyte messages. These settings can be changed by using the three optional arguments: duration, smallest message size, and largest message size. The arguments are specified in order:

1. Change directory to `/usr/src/opa/mpi_apps`.

```
cd /usr/src/opa/mpi_apps
```
2. Run the `run_cabletest` test including the duration in minutes, the smallest message size, and the largest message size.

```
./run_cabletest dd ss ll
```

where:

- `dd` is the duration in minutes.
- `ss` is the smallest message size.

- `11` is the largest message size.

For example, to run a one minute test with 4-megabyte messages, enter the following CLI command:

```
./run_cabletest 1
```

Once one minute has elapsed, the test completes when the current round of testing completes.

If you want the tests to repeat indefinitely, use `infinite` as the duration, as shown in the following CLI command:

```
./run_cabletest infinite
```

In addition to the duration, you can specify the smallest and largest messages to send. The messages must be between 16384 and 4194304 (4 megabytes). The following example tests message sizes between 1 and 4 megabytes, and runs for 24 hours:

```
./run_cabletest 1440 1048576 4194304
```

There are two options available, `-h` and `-v`:

- `-h / --help` – provides this help text.
- `-v / --verbose` – runs the test in a verbose mode that shows you how the nodes were grouped.

8.10.3 run_batch_cabletest

The `run_batch_cabletest` in `/usr/src/opa/mpi_apps` makes it easier to run the `run_cabletest` stress test (see [run_cabletest](#) on page 355). The `run_batch_cabletest` script runs separate jobs for each `BATCH_SIZE` hosts, and can generate the `mpi_group_hosts` files needed using a single `mpi_hosts` file, which lists each host to be tested once, in topology order. For many clusters, `opasorthosts` may help put a list of hosts in topology order, or `opafindgood` may be used to identify candidate hosts. By using many small jobs, the impact of any individual host issues (host crash, hang, etc) during the test is limited to one batch of hosts.

Note: When using `run_batch_cabletest`, the log files are separated. Each individual job gets its own log file, with a suffix to the log filename indicating the run number within the set of batches. For example: `cabletest.04Jan12165901.1` `cabletest.04Jan12165901.2` This avoids any intermingling of output from multiple runs in a single log file.

By default, `run_batch_cabletest` runs for 60 minutes and uses 4-megabyte messages. These settings can be changed by using the three optional arguments: duration, smallest message size, and largest message size. The arguments are specified in order:

1. Change directory to `/usr/src/opa/mpi_apps`.



```
cd /usr/src/opa/mpi_apps
```

2. Run the `run_batch_cabletest` test including the duration in minutes, the smallest message size, and the largest message size.

```
./run_batch_cabletest [duration [minmsg [maxmsg]]]
```

where:

- `duration` is the duration in minutes and can be `infinite`
- `minmsg` is the smallest message size. Must be between 16384 and 4194304.
- `maxmsg` is the largest message size. Must be between 16384 and 4194304.

This builds a set of `mpi_hosts.#` and `mpi_group_hosts.#` files, with no more than `BATCH_SIZE` hosts each. If an odd number of hosts appears in `mpi_hosts`, the last one is skipped.

For example, to run a one minute batch test, with 4-megabyte messages, enter the following CLI command:

```
./run_batch_cabletest 1
```

Once one minute has elapsed, the batch test completes when the current round of testing completes.

If you want the tests to repeat indefinitely, use `infinite` as the duration, as shown in the following CLI command:

```
./run_batch_cabletest infinite
```

In addition to the duration, you can specify the smallest and largest messages to send. This example batches test message sizes between 1 and 4 megabytes, and runs for 24 hours:

```
./run_batch_cabletest 1440 1048576 4194304
```

The following options are available:

- `-h / --help` – provides this help text.
- `-v / --verbose` – runs the test in a verbose mode that shows you how the nodes were grouped.
- `-n` – specifies the number of processes to run per host.
- `duration` – how many minutes to run. Default is 60.
- `minmsg` – smallest message to use. Must be between 16384 and 4194304.
- `maxmsg` – largest message to use. Must be between 16384 and 4194304.

Default `minmsg` and `maxmsg` is 4 Megabytes.

Each `run_cabletest` MPI job has its output saved to a corresponding `/tmp/` `nohup.#.out` file.



Environment Variables

- `MPI_HOSTS` - `mpi_hosts` file to use. The default is `mpi_hosts`. This file lists the hosts in topology order, one entry per host. The hosts are paired sequentially (first and second, third and fourth, and so on).
- `BATCH_SIZE` - The maximum hosts per MPI job. The default is 18, and the number must be even.

Examples

```
./run_batch_cabletest  
MPI_HOSTS=good ./run_batch_cabletest 1440  
BATCH_SIZE=16 MPI_HOSTS=good ./run_batch_cabletest infinite
```

8.10.4 gen_group_hosts

This tool generates an `mpi_group_test` file for use with `run_cabletest`. The `gen_group_hosts` tool asks three questions that need to be answered in order for it to generate the `mpi_group_hosts` file.

The first question asks for the name of your hosts file. The hosts must be listed in this file in group order, with one host per line. The hosts cannot be listed more than once and must be listed in their physical order. The default hosts file is `/usr/src/opa/mpi_apps/mpi_hosts`.

The second question asks how big your groups are. For example, if you want to test each node against the node next to it, use 2 as the group size. If you want to test the nodes connected to one leaf switch against the nodes on another leaf switch, and you have 16 nodes per leaf, use 32 as the group size. The default group size is 2.

The third question asks how many processes you want to run per node. The higher the number, the higher the link utilization. The number must be between 1 and the number of processors per node. The default number of processes per node is 3. Using more processes than needed to saturate the link does not improve testing.

After all questions are answered, the `/usr/src/opa/mpi_apps/mpi_group_hosts` file is generated.

If the number of the hosts is not a multiple of the group size, a warning is shown.

8.10.5 run_multibw

`run_multibw` runs `mpi_multibw`, which performs a multi-core pairwise bandwidth test. `mpi_multibw` is based on OSU bw and multi-lat.

1. Change directory to `/usr/src/opa/mpi_apps`.

```
cd /usr/src/opa/mpi_apps
```
2. Run the `run_multibw` test including the number of processes on which to run the test.

```
./run_multibw processes
```

where: *processes* is the number of processes on which to run the test. All indicates the test should be run for every process in the `mpi_hosts` file.



8.10.6 run_nxnlatbw

run_nxnlatbw runs mpi_nxnlatbw, which is an NxN latency bandwidth test.

1. Change directory to /usr/src/opa/mpi_apps.
2. Run the run_nxnlatbw test, including the number of processes on which to run the test.

```
cd /usr/src/opa/mpi_apps
```

```
./run_nxnlatbw processes
```

where: *processes* is the number of processes on which to run the test. All indicates the test should be run for every process in the mpi_hosts file.

8.11 MPI Batch run_* Scripts

The run_batch_script makes it easier to run other run_* scripts as many smaller jobs. This script is located in /usr/src/opa/mpi_apps and runs separate jobs for each BATCH_SIZE host. By using many small jobs, the impact of any individual host issues (host crash, hang, etc.) during the test is limited to one batch of hosts.

Note: When using run_batch_script, the log files are separated. Each individual job gets its own log file with a suffix to the log filename indicating the run number within the set of batches. For example, mpi_groupstress.04Jan12165901.1
mpi_groupstress.04Jan12165901.2 This scheme avoids any intermingling of output from multiple runs in a single log file.

Usage

```
./run_batch_script [-e] run_script [args]
```

or

```
./run_batch_script --help
```

Options

- -e – Force an even number of hosts in the final batch by skipping the last one.
- run_script – A run_* script from this directory
- args – Arguments for run_script. If the first argument is NP, it is replaced with the process count.

This builds a set of mpi_hosts.# files with no more than BATCH_SIZE hosts each. If -e is specified and an odd number of hosts appear in mpi_hosts, the last one is skipped. Each run_script MPI job has its output saved to a corresponding/tmp/nohup.#.out file

This script is only used for scripts that use MPI_HOSTS.

To run run_cabletest, use run_batch_cabletest.

Environment Variables

- MPI_HOSTS – mpi_hosts file to use. Default is mpi_hosts.



- `BATCH_SIZE` – Maximum hosts per MPI job. The default is 18. If `-e` is used, the number must be even.
- `MIN_BATCH_SIZE` – Minimum hosts per MPI job. The default is 2. If `-e` is used, the number must be even.

The following environment variables are supported in individual `run_*` scripts:

- `SHOW_MPI_HOSTS` – Set to `y` if `MPI_HOSTS` contents should be output prior to starting job.
- `SHOW_MPI_HOSTS_LINES` – Set to the maximum number of lines in hosts file.

Examples

```
./run_batch_script run_deviation NP ff
BATCH_SIZE=2 MPI_HOSTS=good ./run_batch_script run_lat2
BATCH_SIZE=16 MPI_HOSTS=good ./run_batch_script run_deviation ff
MIN_BATCH_SIZE=16 BATCH_SIZE=16 ./run_batch_script run_hpl2 16
```

8.11.1 SHMEM Batch `run_*` scripts

Scripts for various SHMEM benchmarks included with SHMEM are contained in `/usr/src/opa/shmem_apps`. The behavior of these scripts is very similar to those in `mpi_apps`.

Each SHMEM application/benchmark has an accompanying `run_*` script, which assumes the existence of a local `mpi_hosts` file. The provided `run_*` scripts include the following:

- `run_alltoall`
- `run_barrier`
- `run_reduce`
- `run_get[put]_bw`
- `run_get[put]_bibw`



9.0 FastFabric Troubleshooting

This chapter provides instructions and tips for troubleshooting common issues when operating FastFabric tools.

9.1 Switching to the Intel P-State Driver to Run Certain FastFabric Tools

Some Intel-provided tools require the use of the Intel P-State driver rather than the `acpi_cpufreq` driver. For example, the `hostverify.sh` tool fails with RHEL* 6.7 due to the Intel P-State driver not being the default `cpufreq` driver.

If you are using the `acpi_cpufreq` driver, perform one of the following methods to switch to Intel P-State driver in order to use the target tool.

Temporary Switch to Intel P-State Driver

To temporarily switch to the Intel P-state driver, perform the following steps:

1. Make sure `cpupowerutils` package is installed.

```
# yum install cpupowerutils
```

2. Check if any other `cpufreq` kernel driver is active.

```
# cpupower frequency-info -d
```

3. Unload another `cpufreq` kernel driver (if any).

```
# rmmod acpi_cpufreq
```

4. Load `intel_pstate` driver.

```
# modprobe intel_pstate
```

5. Set `cpufreq` governor to "performance".

```
# cpupower -c all frequency-set -g performance
```

6. After using `hostverify.sh` or other tools that needed the Intel P-state set, you may reboot to return to the `acpi_cpufreq` driver.

Load Intel P-State Driver at Boot Time

To load the Intel P-state driver at boot time, perform the following steps:



1. Create a script file `/etc/sysconfig/modules/intel_pstate.modules` and add below text to it.

```
#!/bin/sh
/sbin/modprobe intel_pstate >/dev/null 2>&1
```

2. Add executable permissions for above file.

```
# chmod +x /etc/sysconfig/modules/intel_pstate.modules
```

3. Reboot the system for the changes to take effect.
4. Verify that the Intel P-state driver is loaded.
5. Install the `cpupowerutils` package, if not already installed.

```
# yum install cpupowerutils
```

6. Set `cpufreq` governor to 'performance'.

```
# cpupower -c all frequency-set -g performance
```

To re-enable the `acpi_cpufreq` driver, perform the following:

1. Disable `intel_pstate` in the kernel command line:

Edit `/etc/default/grub` by adding `intel_pstate=disable` to `GRUB_CMDLINE_LINUX`.

For example:

```
GRUB_CMDLINE_LINUX=vconsole.keymap=us console=tty0
vconsole.font=latarcyrheb-sun16 crashkernel=256M
console=ttyS0,115200 intel_pstate=disable
```

2. Apply the change using:

```
if [ -e /boot/efi/EFI/redhat/grub.cfg ]; then
  GRUB_CFG=/boot/efi/EFI/redhat/grub.cfg
else if [ -e /boot/grub2/grub.cfg ]; then
  GRUB_CFG=/boot/grub2/grub.cfg
grub2-mkconfig -o $GRUB_CFG
```

3. Reboot.

When the system comes back up with `intel_pstate` disabled, the `acpi_cpufreq` driver is loaded.



Appendix A Map of Intel® Omni-Path Architecture Commands

The following table maps certain InfiniBand* and Intel® True Scale commands to corresponding Intel® Omni-Path Architecture commands. It is not a complete list of commands.

Table 25. Map of InfiniBand*, Intel® True Scale, and Intel® OPA Commands

InfiniBand*	Intel® True Scale	Intel® OPA
ibstat	ibstat	opainfo ibstat <i>Note:</i> Use opainfo for more complete information.
ibv_devinfo	ibv_devinfo	ibv_devinfo <i>Note:</i> MTU >=4K reports as 4K.
ibstatus	ibstatus	opainfo
ibportstate	ibportstate iba_portconfig	opaportconfig
ibdiagnet	iba_report -o all	opareport -o all
iblinkinfo	iba_report -o links	opareport -o links
ibnetdiscover	iba_report -o links	opareport -o links
ibnodes	iba_report -d 1	opareport -d 1
ibhosts	iba_report -d 1	opareport -d 1
ibswitches	iba_report -d 1	opareport -d 1
sminfo	fabric_info	opafabricinfo opareport -d 1
ibqueryerrors -r	iba_report -o errors	opareport -o errors
ibqueryerrors -k	iba_report -o none -C	opareport -o none -C
ibtracert	iba_report -o route -S node:x -D node:y	opareport -o route -S node:x -D node:y
ibroute	iba_report -o linear [-o mcast]	opareport -o linear [-o mcast]
perfquery ibclearerrors	iba_extract_stat iba_extract_stat2 iba_report -o nodes -s -d 10	opaextractstat opaextractstat2 opareport -o nodes -s -d 10
ibping	ibping	ibping
ibcheckwidth	iba_report -o slowlinks	opareport -o slowlinks