

Sun Datacenter InfiniBand Switch 72

Command Reference



Copyright © 2009, 2010 Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related software documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Copyright © 2009, 2010, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. UNIX est une marque déposée concédée sous licence par X/Open Company, Ltd.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation.



Adobe PostScript

Contents

Using This Documentation xv

Understanding the Commands 1

Understanding Hardware Commands 1

checkboot Command 2

Syntax 2

Description 2

Example 3

checkpower Command 3

Syntax 3

Description 3

Example 3

checkvoltages Command 4

Syntax 4

Description 4

Example 4

connector Command 5

Syntax 5

Description 5

Options 6

Example 7

disablesm Command 7

Syntax	7
Description	7
Example	8
disableswitchport Command	8
Syntax	8
Description	8
Example	8
enablesm Command	9
Syntax	9
Description	9
Example	9
enableswitchport Command	10
Syntax	10
Description	10
Example	10
env_test Command	11
Syntax	11
Description	11
Example	11
getfanspeed Command	12
Syntax	13
Description	13
Example	13
getportstatus Command	13
Syntax	13
Description	14
Example	14
i4reset Command	14

Syntax	14
Description	14
Example	15
listlinkup Command	15
Syntax	15
Description	15
Example	15
managementreset Command	16
Syntax	16
Description	16
Example	17
nm2version Command	17
Syntax	17
Description	17
Example	18
readchfru Command	18
Syntax	18
Description	19
Example	19
setlinkspeed Command	19
Syntax	19
Description	20
Example	20
setsmpriority Command	20
Syntax	20
Description	21
Example	21
showtemps Command	22

Syntax	22
Description	22
Example	22
showunhealthy Command	23
Syntax	23
Description	23
Example	23
Understanding InfiniBand Commands	23
ibaddr Command	26
Syntax	26
Description	26
Options	26
Example	27
ibcheckerrors Command	27
Syntax	27
Description	28
Options	28
Example	28
ibchecknet Command	29
Syntax	29
Description	29
Options	30
Example	30
ibchecknode Command	31
Syntax	31
Description	31
Options	32
Example	32

ibcheckport Command	32
Syntax	32
Description	33
Options	33
Example	34
ibcheckportstate Command	34
Syntax	34
Description	34
Options	35
Example	35
ibcheckportwidth Command	35
Syntax	36
Description	36
Options	36
Example	37
ibcheckstate Command	37
Syntax	37
Description	37
Options	38
Example	38
ibcheckwidth Command	39
Syntax	39
Description	39
Options	39
Example	40
ibclearcounters Command	40
Syntax	40
Description	41

Options	41
Example	41
ibclearerrors Command	42
Syntax	42
Description	42
Options	42
Example	43
ibdatacounters Command	43
Syntax	43
Description	43
Options	44
Example	44
ibdatacounts Command	45
Syntax	45
Description	45
Options	45
Example	46
ibdiagnet Command	46
Syntax	46
Description	47
Options	48
Example	49
ibdiagpath Command	51
Syntax	51
Description	51
Options	52
Example	53
ibhosts Command	54

Syntax	55
Description	55
Options	55
Example	55
ibnetdiscover Command	56
Syntax	56
Description	56
Options	57
Example	58
ibnodes Command	59
Syntax	59
Description	60
Options	60
Example	60
ibportstate Command	61
Syntax	61
Description	61
Options	62
Example	62
ibroute Command	63
Syntax	63
Description	64
Options	64
Example	65
ibstatus Command	65
Syntax	65
Description	66
Options	66

Example	66
ibswitches Command	67
Syntax	67
Description	67
Options	67
Example	68
ibsysstat Command	68
Syntax	68
Description	69
Options	69
Example	70
ibtracert Command	70
Syntax	71
Description	71
Options	71
Example	72
opensm Command	72
Syntax	72
Description	73
Options	74
Example	77
opensmd Daemon	78
Syntax	78
Description	78
Options	81
Example	82
osmtest Command	82
Syntax	82

Description	82
Options	83
Example	84
perfquery Command	85
Syntax	85
Description	86
Options	86
Example	87
saquery Command	87
Syntax	88
Description	88
Options	89
Example	90
sminfo Command	90
Syntax	91
Description	91
Options	91
Example	92
smpdump Command	92
Syntax	93
Description	93
Options	93
Example	94
smpquery Command	94
Syntax	94
Description	95
Options	95
Example	96

Using This Documentation

This command reference provides detailed information regarding the commands for administering the Sun Datacenter InfiniBand Switch 72 from Oracle®. This document is written for developers, system administrators, and users who have advanced experience administering InfiniBand fabrics and hardware.

- “Related Documentation” on page xv
- “Documentation, Support, and Training” on page xvi
- “Documentation Feedback” on page xvi

Related Documentation

The documents listed as online are available at:

(<http://docs.sun.com/app/docs/prod/ib.switch.72>)

Application	Title	Part Number	Format	Location
Getting started	<i>Sun Datacenter InfiniBand Switch 72 Getting Started Guide</i>	820-7755	Printed PDF	Shipping kit Online
Last-minute information	<i>Sun Datacenter InfiniBand Switch 72 Product Notes</i>	820-7753	PDF	Online
Installation, administration, and service	<i>Sun Datacenter InfiniBand Switch 72 User's Guide</i>	820-7751	PDF HTML	Online
Command reference	<i>Sun Datacenter InfiniBand Switch 72 Command Reference</i>	820-7752	PDF HTML	Online
Compliance	<i>Sun Datacenter InfiniBand Switch 72 Safety and Compliance Guide</i>	820-7754	PDF	Online

Some commands reference the PICMG 3.0 specification, available at:

(<http://www.picmg.org>)

Some commands reference the IPMI specification, available at:

(<http://www.intel.com/design/servers/ipmi>)

Documentation, Support, and Training

These web sites provide additional resources:

- Documentation (<http://docs.sun.com>)
- Support (<http://www.sun.com/support>)
- Training (<http://www.sun.com/training>)

Documentation Feedback

Submit comments about this document by clicking the Feedback[+] link at (<http://docs.sun.com>). Include the title and part number of your document with your feedback:

Sun Datacenter InfiniBand Switch 72 Command Reference, part number 820-7752-12.

Understanding the Commands

This document provides detailed information regarding the commands which are used to administer the switch.

- [“Understanding Hardware Commands” on page 1](#)
- [“Understanding InfiniBand Commands” on page 23](#)

Related Information

- [Switch User’s Guide](#)
- [Switch Remote Administration](#)

Understanding Hardware Commands

The hardware commands are a simplified way to direct the Komtron management controller, internal to the switch, to perform complex tasks. Only the root user of the management controller can run all of the hardware commands. The format of the hardware commands is as follows:

```
# command [arguments] [arguments] . . .
```

Command Syntax	Links
checkboot	“checkboot Command” on page 2
checkpower	“checkpower Command” on page 3
checkvoltages	“checkvoltages Command” on page 4
connector <i>number</i> [<i>present</i> <i>read reg</i> <i>write reg data</i>]	“connector Command” on page 5
disableesm	“disableesm Command” on page 7
disableswitchport <i>switch_chip port</i>	“disableswitchport Command” on page 8
enableesm	“enableesm Command” on page 9
enableswitchport <i>switch_chip port</i>	“enableswitchport Command” on page 10

Command Syntax	Links
<code>env_test</code>	“env_test Command” on page 11
<code>getfanspeed</code>	“getfanspeed Command” on page 12
<code>getportstatus</code> <i>switch_chip port [-v]</i>	“getportstatus Command” on page 13
<code>i4reset</code> <i>switch_chip</i>	“i4reset Command” on page 14
<code>listlinkup</code>	“listlinkup Command” on page 15
<code>managementreset</code>	“managementreset Command” on page 16
<code>nm2version</code>	“nm2version Command” on page 17
<code>readchfru</code>	“readchfru Command” on page 18
<code>setlinkspeed</code> <i>switch_chip port speed</i>	“setlinkspeed Command” on page 19
<code>setsmpriority</code> <i>priority</i>	“setsmpriority Command” on page 20
<code>showtemps</code>	“showtemps Command” on page 22
<code>showunhealthy</code>	“showunhealthy Command” on page 23

Related Information

- [“Understanding InfiniBand Commands” on page 23](#)

checkboot Command

Displays switch chip boot status.

Syntax

```
checkboot
```

Description

This hardware command checks the boot status of the switch chip. Output is a simplified OK.

Example

The following example shows how to check the boot status of the switch chips with the `checkboot` command.

```
# checkboot  
I4-A OK  
I4-B OK  
I4-C OK  
I4-D OK  
I4-E OK  
I4-F OK  
#
```

Related Information

- [“i4reset Command” on page 14](#)
- [“env_test Command” on page 11](#)

checkpower Command

Displays power supply status.

Syntax

```
checkpower
```

Description

This hardware command checks the status of the power supplies. Output is a simplified OK.

Example

The following example shows how to check the power supply status with the `checkpower` command.

```
# checkpower
```

```
PSU 0 present status: OK
PSU 1 present status: OK
#
```

Related Information

- “[checkvoltages Command](#)” on page 4
- “[env_test Command](#)” on page 11

checkvoltages Command

Verifies voltages.

Syntax

```
checkvoltages
```

Description

This hardware command displays the internal voltages for the main board. The left column displays the expected voltage, and the right column displays the measured voltage. The command also provides a summary of the voltage conditions.

Example

The following example shows how to check that voltages are at nominal values with the `checkvoltages` command.

```
# checkvoltages
Voltage ECB OK
Measured 3.3V Main = 3.28 V
Measured 3.3V Standby = 3.37 V
Measured 12V = 12.00 V
Measured 5V = 5.10 V
Measured VBAT = 3.08 V
Measured 1.8V = 1.77 V
Measured 1.2V Standby = 1.21 V
Measured 1.8V Standby = 1.80 V
Measured 2.5VA = 2.51 V
Measured 2.5VB = 2.51 V
```

```
Measured 1.2VA = 1.22 V
Measured 1.2VB = 1.21 V
Measured 1.2VC = 1.21 V
Measured 1.2VD = 1.22 V
Measured 1.2VE = 1.21 V
Measured 1.2VF = 1.21 V
All voltages OK
#
```

Related Information

- [“checkpower Command” on page 3](#)
- [“env_test Command” on page 11](#)

connector Command

Reads and writes CXP cable registers.

Syntax

`connector number present | read reg | write reg data`

where:

- *number* is the number of the connector (0A–11B).
- *reg* is the data register to act upon.
- *data* is the value to write to the register (in decimal).

Description

This hardware command performs a pass-fail test to verify that an InfiniBand cable is connected to a particular connector. The command can also read the data registers of the cable and write data to those registers.

Note – The values read from and stored in the registers are hexadecimal. When writing to the register, the *data* is automatically converted from decimal to hexadecimal.

The following table provides more commonly read registers and a description of their contents.

Register	Description
22	Temperature. Convert value to decimal = degrees centigrade.
26	Voltage. Convert value to decimal x 0.026 = volts.
130	Connector type: <ul style="list-style-type: none"> • 12 (0Ch) – MPO • 32 (20h) – HSSDC II • 33 (21h) – Copper pigtail • 34 (22h) – RJ-45
140	Bit rate. Value x 100 = Mbps.
144	Length of optical cable supported. Value x 1 = meters.
146	Length of copper cable supported. Value x 1 = meters.
148–163	Vendor Name. 16-byte ASCII.
164	Transceiver code: <ul style="list-style-type: none"> • Bit 5 – IB 4x 850 nm optical • Bit 4 – IB 4x copper active • Bit 3 – IB 4x copper passive • Bit 2 – QDR speed (10.0 Gbps) • Bit 1 – DDR speed (5.0 Gbps) • Bit 0 – SDR speed (2.5 Gbps)
168–183	Part number. 16-byte ASCII.
184–185	Revision. 2-byte ASCII.
196–211	Serial Number. 16-byte ASCII.
212–213	Manufacturing year. 2-byte ASCII (2000 = 00).
214–215	Manufacturing month. 2-byte ASCII (January = 01).
216–217	Manufacturing day. 2-byte ASCII (1st = 01).

Options

The following table describes the options to the `connector` command and their purposes:

Option	Purpose
<code>present</code>	Checks for the presence of connector <i>number</i>

Option	Purpose
read	Returns the data in register <i>reg</i> . Data values are hexadecimal.
write	Writes the value of <i>data</i> into the register <i>reg</i> . Data values are converted from decimal to hexadecimal.

Example

The following example shows how to write a value of 14 (E in hexadecimal) to register 218 of connector 1A with the `connector` command.

```
# connector 1A read 218
Read connector 1A reg 218 value 0xff
# connector 1A write 218 14
Write connector 1A reg 218 value 0E
# connector 1A read 218
Read connector 1A reg 218 value 0x0E
#
```

Related Information

- [“env_test Command” on page 11](#)
- [“listlinkup Command” on page 15](#)

disablesm Command

Disables the Subnet Manager within the management controller.

Syntax

```
disablesm
```

Description

This hardware command disables the OpenSM `opensmd` daemon. You use this command in the event that a Subnet Manager, external to the switch, is preferred.

Example

The following example shows how to disable the `opensmd` daemon with the `disableesm` command.

```
# disableesm
Stopping IB Subnet Manager...-.-.-.-.-+          [ OK ]
#
```

Related Information

- [“enableesm Command” on page 9](#)
- [“setsmpriority Command” on page 20](#)
- [“opensm Command” on page 72](#)

disableswitchport Command

Disables a switch chip port.

Syntax

```
disableswitchport switch_chip port
```

where:

- *switch_chip* is the letter of the switch chip (A–F).
- *port* is the number of the port (1–36).

Description

This hardware command disables a port on the switch chip.

Example

The following example shows how to disable port 1 on switch chip A with the `disableswitchport` command.

```
# disableswitchport A 1
#
```


Related Information

- [“enableswitchport Command” on page 10](#)
- [“ibportstate Command” on page 61](#)

enablesm Command

Enables the Subnet Manager within the management controller.

Syntax

```
enablesm
```

Description

This hardware command enables the OpenSM `opensmd` daemon on the management controller.

Note – Similarly, both the `enablesm` and `opensm` commands invoke an instance of the same Subnet Manager. Conversely, the `enablesm` command merely starts the Subnet Manager while the `opensm` command can configure the Subnet Manager.

Like the OpenSM Subnet Manager discussed in [“opensm Command” on page 72](#), the `enablesm` command initiates an instance of the `opensmd` *daemon*. The daemon reads the `/etc/opensm/opensm.conf` configuration file upon startup, which it uses to configure the OpenSM Subnet Manager. The `enablesm` command is more user-friendly because it is not necessary to provide command-line options and arguments.

Example

The following example shows how to enable the `opensmd` daemon with the `enablesm` command.

```
# enablesm
Starting IB Subnet Manager.           [ OK ]
#
```

Related Information

- [“disablesm Command” on page 7](#)
- [“setsmpriority Command” on page 20](#)
- [“opensm Command” on page 72](#)

enableswitchport Command

Enables a switch chip port.

Syntax

```
enableswitchport switch_chip port
```

where:

- *switch_chip* is the letter of the switch chip (A–F).
- *port* is the number of the port (1–36).

Description

This hardware command enables a port on the switch chip.

Example

The following example shows how to enable port 1 on switch chip A with the `enableswitchport` command.

```
# enableswitchport A 1
#
```

Related Information

- [“disableswitchport Command” on page 8](#)
- [“ibportstate Command” on page 61](#)

env_test Command

Displays environmental status.

Syntax

```
env_test
```

Description

This hardware command performs a series of hardware and environmental tests of the switch. This command is an amalgamation of the following commands:

- checkpower
- checkvoltages
- showtemps
- getfanspeed
- connector
- checkboot

The command output provides voltage and temperature values, pass-fail results, and error messages.

Example

The following example shows how to display the hardware and environmental status of the switch with the `env_test` command.

```
# env_test
NM2 Environment test started:
Starting Voltage test:
Voltage ECB OK
Measured 3.3V Main = 3.28 V
Measured 3.3V Standby = 3.40 V
Measured 12V = 12.06 V
Measured 5V = 5.10 V
Measured VBAT = 3.17 V
Measured 1.8V = 1.78 V
Measured 1.2V Standby = 1.21 V
Measured 1.8V Standby = 1.80 V
Measured 2.5VA = 2.51 V
Measured 2.5VB = 2.51 V
```

```
Measured 1.2VA = 1.22 V
Measured 1.2VB = 1.22 V
Measured 1.2VC = 1.21 V
Measured 1.2VD = 1.21 V
Measured 1.2VB = 1.21 V
Measured 1.2VE = 1.21 V
Measured 1.2VF = 1.21 V
Voltage test returned OK
Starting PSU test:
PSU 0 present
PSU 1 present
PSU test returned OK
Starting Temperature test:
Back temperature 23.00
Front temperature 32.62
ComEx temperature 26.12
I4-A temperature 55, maxtemperature 56
I4-B temperature 48, maxtemperature 49
I4-C temperature 53, maxtemperature 53
I4-D temperature 48, maxtemperature 49
I4-E temperature 53, maxtemperature 54
I4-F temperature 53, maxtemperature 54
Temperature test returned OK
Starting FAN test:
Fan 0 running at rpm 12433
Fan 1 running at rpm 12311
Fan 2 running at rpm 12311
Fan 3 running at rpm 12433
Fan 4 running at rpm 12433
FAN test returned OK
Starting Connector test:
Connector test returned OK
Starting I4 test:
I4-A OK
I4-B OK
I4-C OK
I4-D OK
I4-E OK
I4-F OK
All I4s OK
I4 test returned OK
NM2 Environment test PASSED
#
```

get fanspeed Command

Displays fan speed.

Syntax

```
getfanspeed
```

Description

This hardware command displays the speed of the fans. The command also indicates if the fan is “not present” or has “stopped”.

Example

The following example shows how to display fan speeds with the `getfanspeed` command.

```
# getfanspeed
Fan 0 rpm 17940
Fan 1 rpm 17940
Fan 2 rpm 17687
Fan 3 rpm 17940
Fan 4 rpm 17687
#
```

Related Information

- [“env_test Command” on page 11](#)

getportstatus Command

Displays port status.

Syntax

```
getportstatus switch_chip port [-v]
```

where:

- *switch_chip* is the letter of the switch chip (A–F).
- *port* is the number of the port (1–36).

Description

This hardware command returns the status of the specified *port* of the *switch_chip*. The `-v` option provides verbose output.

Example

The following example shows how to display the status of port 7 on switch chip A with the `getportstatus` command.

```
# getportstatus A 7
Portstate 4
Portphystate 5
LinkWidthActive 2
LinkSpeedActive 4
#
```

Related Information

- [“ibcheckport Command” on page 32](#)
- [“ibcheckportstate Command” on page 34](#)
- [“ibcheckportwidth Command” on page 35](#)

i4reset Command

Resets the switch chip.

Syntax

```
i4reset switch_chip
```

where *switch_chip* is the letter of the switch chip (A–F).

Description

This hardware command resets the switch chip.

Example

The following example shows how to reset switch chip A with the `i4reset` command.

```
# i4reset A
Reset I4-A
#
```

Related Information

- [“checkboot Command” on page 2](#)
- [“ibportstate Command” on page 61](#)

listlinkup Command

Displays links presence.

Syntax

```
listlinkup
```

Description

This hardware command lists the presence of links and the up-down state of the associated ports on the switch chip.

Example

The following example shows how to display link presence and associated ports with the `listlinkup` command.

```
# listlinkup
Connector 0A Present <-> I4-A Ports 22 up 21 up 20 up
Connector 1A Not present
Connector 2A Not present
Connector 3A Not present
Connector 4A Not present
.
.
```

```
Connector 10B Not present
Connector 11B Not present
Link I4-A 01 <-> I4-E 09 up
Link I4-A 02 <-> I4-F 08 up
Link I4-A 03 <-> I4-F 07 up
.
.
.
Link I4-D 18 <-> I4-E 16 up
#
```

Note – The output in the example is just a portion of the full output.

Related Information

- [“setlinkspeed Command” on page 19](#)
- [“ibportstate Command” on page 61](#)

managementreset Command

Resets the management controller.

Syntax

```
managementreset [-r]
```

Description

This hardware command resets the management controller, the CPLD, and the I4 switch chips. The command requests a reboot, and in most situations, this should be done. The `managementreset` command also forces links to retrain upon management controller services startup. The `-r` option bypasses the query for reboot and reboots automatically.

Example

The following example shows how to reset the management controller and InfiniBand fabric services with the `managementreset` command.

```
# managementreset
Stopping Environment daemon, please wait
Resetting CPLD, please wait
Restarting Environment daemon
Reboot needed to reconnect to I4 and enable IB ports
Do you want do reboot now [yes/no]:yes
Broadcast message from root (pts/0) (Fri Nov 20 17:10:27 2009):
The system is going down for reboot NOW!
# Connection to 123.45.67.89 closed by remote host.
Connection to 123.45.67.89 closed.
#
```

Note – By rebooting the management controller, the link to the management console is severed. You must re-access the management controller to regain administrative control.

Related Information

- [“i4reset Command” on page 14](#)
- [“ibportstate Command” on page 61](#)

nm2version Command

Displays switch version.

Syntax

```
nm2version
```

Description

This hardware command shows the hardware and software versions, and date information for the switch and management controller.

Note – The output of the `nm2version` command contains extraneous information relevant only to the manufacturer of the management controller. See the following example.

Example

The following example shows how to display the version information with the `nm2version` command.

```
# nm2version
NM2-72p version: 0.1.0-1
Build time: Aug 24 2009 16:41:03
FPGA version: 0x94
ComExpress info:
Board Name: "NOW1"
Manufacturer Name: "JUMP"
Manufacturing Date: 2009.02.19
Last Repair Date: 1980.01.01
Serial Number: "NCD2S0240"
Hardware Revision: 0x0100
Firmware Revision: 0x0102
Jida Revision: 0x0103
Feature Number: 0x0001
#
```

In this example, only the output that is **bold** is relevant to the administration of the switch.

Related Information

- *Switch Service*, display switch chip firmware version

readchfru Command

Displays chassis FRU information.

Syntax

```
readchfru
```


- *switch_chip* is the letter of the switch chip (A–F).
- *port* is the number of the port (1–36).
- *speed* is the speed of the port 1(x), 2(x), or 4(x).

Description

This hardware command sets the *speed* of a *port* on a *switch_chip*. If too many symbol errors are encountered, try setting for a slower speed (2 or 1).

Example

The following example shows how to set the speed of port 3 on switch chip A to 4x with the `setlinkspeed` command.

```
# setlinkspeed A 3 4
# enableswitchport A 3
#
```

Note – You must reset the port after setting the port speed. You can use the `enableswitchport` command to reset the port.

Related Information

- [“listlinkup Command” on page 15](#)
- [“ibportstate Command” on page 61](#)

setsmpriority Command

Sets the Subnet Manager priority.

Syntax

```
setsmpriority priority
```

where *priority* is a number from 0 (lowest) to 13 (highest).

Description

This hardware command sets the priority of the Subnet Manager within the management controller. You use this command when there are multiple Subnet Managers in the InfiniBand fabric. By setting a Subnet Manager to a higher priority than another Subnet Manager, it becomes the master Subnet Manager. By setting a Subnet Manager to a lower priority than another Subnet Manager, it becomes the slave Subnet Manager.

The `setsmpriority` command writes the value of *priority* to the `sm_priority` parameter of the `/etc/opensm/opensm.conf` file.

Note – You must stop or disable the OpenSM Subnet Manager before issuing the `setsmpriority` command. See [“disableesm Command” on page 7](#) and [“opensmd Daemon” on page 78](#).

Note – Setting Subnet Managers of the same fabric to the same priority can have undesirable results.

Example

The following example shows how to set the priority of the Subnet Manager to 3 using the `setsmpriority` command.

```
# setsmpriority 3
-----
OpenSM 3.2.6_20090717
  Reading Cached Option File: /etc/opensm/opensm.conf
  Loading Cached Option:routing_engine = ftree
  Loading Cached Option:sm_priority = 13
  Loading Cached Option:sminfo_polling_timeout = 1000
  Loading Cached Option:polling_retry_number = 3
Command Line Arguments:
  Priority = 3
  Creating config file template '/tmp/osm.conf'.
  Log File: /var/log/opensm.log
-----
#
```

Note – You *must* restart the Subnet Manager using the `disableesm` and `enableesm` commands after setting the priority.

Related Information

- [“disablem Command” on page 7](#)
- [“enablem Command” on page 9](#)
- [“opensm Command” on page 72](#)

showtemps Command

Displays switch temperatures.

Syntax

```
showtemps
```

Description

This hardware command displays current and maximum achieved internal temperatures for the switch.

Example

The following example shows how to display switch temperatures with the showtemps command.

```
# showtemps
Back temperature 22.88
Front temperature 31.75
Com-Express temperature 26.12
I4-A temperature 55 maxtemperature 55
I4-B temperature 47 maxtemperature 49
I4-C temperature 52 maxtemperature 52
I4-D temperature 48 maxtemperature 49
I4-E temperature 52 maxtemperature 52
I4-F temperature 52 maxtemperature 54
#
```

Related Information

- [“env_test Command” on page 11](#)

showunhealthy Command

Displays problematic components.

Syntax

```
showunhealthy
```

Description

This hardware command shows a list of switch components that appear to have a problem. Unlike the `env_test` command, the `showunhealthy` command *only* displays messages for components that have failed testing.

Example

The following example shows how to display a list of unhealthy components in the switch with the `showunhealthy` command.

```
# showunhealthy  
OK - No unhealthy sensors  
#
```

Related Information

- [“env_test Command” on page 11](#)

Understanding InfiniBand Commands

The InfiniBand commands are a means of monitoring and controlling many aspects of the InfiniBand fabric. These commands are also run from the management controller, which is also the host of a Subnet Manager external to the switch. Only the `root` user of the management controller can run the InfiniBand commands. The format of the InfiniBand commands is typically as follows:

command [*option*] [*option*] ...

Command Syntax	Links
<code>ibaddr [-d] [-D] [-G] [-l] [-g] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>] [-V] [-h] [<i>lid dr_path guid</i>]</code>	“ibaddr Command” on page 26
<code>ibcheckerrors [-h] [-b] [-v] [-N] [<i>topology</i>] [-C <i>ca_name</i> -P <i>ca_port</i> -t <i>timeout</i>]</code>	“ibcheckerrors Command” on page 27
<code>ibchecknet [-h] [-N] [<i>topology</i>] [-C <i>ca_name</i> -P <i>ca_port</i> -t <i>timeout</i>]</code>	“ibchecknet Command” on page 29
<code>ibchecknode [-h] [-v] [-N] [-G] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>] [<i>lid dr_path guid</i>]</code>	“ibchecknode Command” on page 31
<code>ibcheckport [-h] [-v] [-N] [-G] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>] <i>lid guid port</i></code>	“ibcheckport Command” on page 32
<code>ibcheckportstate [-G] [-h] [-N] [-v] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>] [<i>lid guid port</i>]</code>	“ibcheckportstate Command” on page 34
<code>ibcheckportwidth [-G] [-h] [-N] [-v] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>] [<i>lid guid port</i>]</code>	“ibcheckportwidth Command” on page 35
<code>ibcheckstate [-h] [-N] [-v] [<i>topology</i>] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>]</code>	“ibcheckstate Command” on page 37
<code>ibcheckwidth [-h] [-N] [-v] [<i>topology</i>] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>]</code>	“ibcheckwidth Command” on page 39
<code>ibclearcounters [-h] [<i>topology</i>] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>]</code>	“ibclearcounters Command” on page 40
<code>ibclearerrors [-h] [-N] [<i>topology</i>] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>]</code>	“ibclearerrors Command” on page 42
<code>ibdatacounters [-b] [-h] [-N] [-v] [<i>topology</i>] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>]</code>	“ibdatacounters Command” on page 43
<code>ibdatacounts [-b] [-G] [-h] [-N] [-v] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>] <i>lid guid port</i></code>	“ibdatacounts Command” on page 45
<code>ibdiagnet [-c <i>count</i>] [-v] [-r] [-o <i>outputdir</i>] [-t <i>topology</i>] [-s <i>system</i>] [-i <i>device</i>] [-p <i>port</i>] [-wt <i>topology</i>] [-pm] [-pc] [-P <i>PM = value</i>] [-lw 1x 4x 12x] [-ls 2.5 5 10] [-skip <i>checks</i>] [-load_db <i>file</i>] [-h] [-V]</code>	“ibdiagnet Command” on page 46
<code>ibdiagpath -n[<i>src_name, </i>]<i>dst_name</i> -l[<i>src_lid, </i>]<i>dst_lid</i> -d <i>p1,p2,p3,...</i> [-c <i>count</i>] [-v] [-o <i>outputdir</i>] [-t <i>topology</i>] [-s <i>system</i>] [-i <i>device</i>] [-p <i>port</i>] [-wt <i>topology</i>] [-pm] [-pc] [-P <i>PM = value</i>] [-lw 1x 4x 12x] [-ls 2.5 5 10] [-skip <i>checks</i>] [-load_db <i>file</i>] [-h] [-V]</code>	“ibdiagpath Command” on page 51
<code>ibhosts [-h] [<i>topology</i>] [-C <i>ca_name</i>] [-P <i>ca_port</i>] [-t <i>timeout</i>]</code>	“ibhosts Command” on page 54

Command Syntax	Links
<code>ibnetdiscover [-d] [-e] [-v] [-s] [-l] [-g] [-H] [-S] [-R] [-C ca_name] [-P ca_port] [-t timeout] [-V] [-p] [-h] [topology]</code>	“ibnetdiscover Command” on page 56
<code>ibnodes [-h] [topology] [-C ca_name] [-P ca_port] [-t timeout]</code>	“ibnodes Command” on page 59
<code>ibportstate [-d] [-D] [-e] [-G] [-h] [-s smlid] [-v] [-C ca_name] [-P ca_port] [-t timeout] lid dr_path guid port [op]</code>	“ibportstate Command” on page 61
<code>ibroute [-d] [-a] [-n] [-D] [-e] [-G] [-h] [-M] [-s smlid] [-v] [-V] [-C ca_name] [-P ca_port] [-t timeout] [lid dr_path guid [startlid [endlid]]]</code>	“ibroute Command” on page 63
<code>ibstatus [-h] [devname[:ib_port]]...</code>	“ibstatus Command” on page 65
<code>ibswitches [-h] [topology] [-C ca_name] [-P ca_port] [-t timeout]</code>	“ibswitches Command” on page 67
<code>ibsysstat [-d] [-e] [-G] [-h] [-s smlid] [-v] [-V] [-C ca_name] [-P ca_port] [-t timeout] [-o oui] [-S] lid guid [op]</code>	“ibsysstat Command” on page 68
<code>ibtracert [-d] [-D] [-G] [-h] [-m mlid] [-s smlid] [-v] [-V] [-C ca_name] [-P ca_port] [-t timeout] [lid dr_path guid [startlid [endlid]]]</code>	“ibtracert Command” on page 70
<code>opensm [-F filename] [-c filename] [-g guid] [-l lmc] [-p priority] [-smkey SMKey] [-r] [-R engine] [-A] [-z] [-M filename] [-U filename] [-S filename] [-a path] [-u path] [-X path] [-m path] [-o] [-s interval] [-t timeout] [-maxsmpps number] [-console[off local socket loopback]] [-console-port port] [-i filename] [-f path] [-L size] [-e] [-P filename] [-N] [-Q[-Y filename]] [-y] [-B] [-I] [-v] [-V] [-D flags] [-dopt] [-h]</code>	“opensm Command” on page 72
<code>osmtest [-f c a v s e f m q t] [-w time] [-dopt] [-m lid] [-g guid] [-p] [-i filename] [-sopt] [-Mopt] [-t timeout] [-l path] [-v] [-vf flags] [-h]</code>	“osmtest Command” on page 82
<code>perfquery [-d] [-e] [-G] [-h] [-a] [-l] [-r] [-R] [-v] [-V] [-C ca_name] [-P ca_port] [-t timeout] [lid guid [[port] [reset_mask]]]</code>	“perfquery Command” on page 85
<code>saquery [-h] [-d] [-p] [-N] [-D] [-S] [-I] [-L] [-l] [-G] [-O] [-U] [-c] [-s] [-g] [-m] [-x] [-C ca_name] [-P ca_port] [-t timeout] [--src-to-dst source:destination] [--sgid-to-dgid source-destination] [name lid guid]</code>	“saquery Command” on page 87
<code>sminfo [-d] [-e] -s state -p priority -a activity [-D] [-G] [-h] [-v] [-V] [-C ca_name] [-P ca_port] [-t timeout] smlid smdr_path</code>	“sminfo Command” on page 90
<code>smpdump [-s] [-D] [-h] [-v] [-C ca_name] [-P ca_port] [-t timeout] lid dr_path attr [mod]</code>	“smpdump Command” on page 92
<code>smpquery [-d] [-D] [-e] [-G] [-h] [-v] [-V] [-C ca_name] [-P ca_port] [-t timeout] lid dr_path guid [op params]</code>	“smpquery Command” on page 94

Related Information

- “Understanding Hardware Commands” on page 1

ibaddr Command

Queries InfiniBand addresses.

Syntax

```
ibaddr [-d] [-D] [-G] [-l] [-g] [-C ca_name] [-P ca_port] [-t timeout] [-V] [-h] [lid | dr_path | guid]
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *guid* is the global unit identifier.

Description

This InfiniBand software command displays the LID and range as well as the GID address of the port specified. The local port information is provided by default.

Note – This command is also used as a simple address resolver.

Options

The following table describes the options to the `ibaddr` command and their purposes:

Option	Purpose
-d	Sets the debug level. Can be used several times to increase the debug level.
-D	Uses the directed path address. The path is a comma delimited sequence of out ports.

Option	Purpose
-G	Shows the LID range and GUID for port GUID addresses.
-l	Shows the LID range only.
-g	Shows the GUID address only.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.
-V	Displays the version information.
-h	Provides help.

Example

The following example shows how to display the local port's GUID and LID range with the `ibaddr` command.

```
# ibaddr
GID fe80::2:c903:2:cb6a LID start 0x1 end 0x1
#
```

Related Information

- [ibaddr man page](#)
- [“ibroute Command” on page 63](#)
- [“ibtracert Command” on page 70](#)

ibcheckerrors Command

Validates InfiniBand fabric and report errors.

Syntax

```
ibcheckerrors [-h] [-b] [-v] [-N] [topology] -C ca_name -P ca_port -t timeout]
```

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.

- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that uses the topology file created by the `ibnetdiscover` command to scan the InfiniBand fabric to validate the connectivity and report errors from the port counters.

Options

The following table describes the options to the `ibcheckerrors` command and their purposes:

Option	Purpose
-h	Provides help.
-b	Enables brief mode. Reduced output is only if errors are present. Does not identify the errors.
-v	Provides verbose output.
-N	Uses mono mode instead of color mode.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to check error counters for all LIDs in the InfiniBand fabric with the `ibcheckerrors` command.

```
# ibcheckerrors
#warn: counter SymbolErrors = 3121      (threshold 10) lid 25 port 255
#warn: counter RcvSwRelayErrors = 48545  (threshold 100) lid 25 port 255
#warn: counter XmtDiscards = 9789       (threshold 100) lid 25 port 255
Error check on lid 25 (Sun DCS 72 QDR FC switch o4nm2-72p-2) port all:  FAILED
#warn: counter RcvSwRelayErrors = 56839  (threshold 100) lid 25 port 28
Error check on lid 25 (Sun DCS 72 QDR FC switch o4nm2-72p-2) port 28:  FAILED
#warn: counter RcvSwRelayErrors = 56839  (threshold 100) lid 25 port 9
Error check on lid 25 (Sun DCS 72 QDR FC switch o4nm2-72p-2) port 9:  FAILED
#warn: counter XmtDiscards = 9714       (threshold 100) lid 25 port 1
```

```
Error check on lid 25 (Sun DCS 72 QDR FC switch o4nm2-72p-2) port 1: FAILED
.
.
.
## Summary: 6 nodes checked, 0 bad nodes found
##          142 ports checked, 3 ports have errors beyond threshold
#
```

Note – The output in the example is just a portion of the full output.

Related Information

- [ibcheckerrors man page](#)
- [“ibchecknode Command” on page 31](#)
- [“ibcheckport Command” on page 32](#)
- [“ibnetdiscover Command” on page 56](#)

ibchecknet Command

A simplified version of the `ibcheckerrors` command.

Syntax

```
ibchecknet [-h] [-N] [topology] -C ca_name -P ca_port -t timeout]
```

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that uses the topology file created by the `ibnetdiscover` command to scan the InfiniBand fabric to validate the connectivity and report errors from the port counters.

Options

The following table describes the options to the `ibchecknet` command and their purposes:

Option	Purpose
-h	Provides help.
-N	Uses mono mode instead of color mode.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to check error counters for all LIDs in the InfiniBand fabric with the `ibchecknet` command.

```
# ibchecknet
#warn: counter SymbolErrors = 3121      (threshold 10) lid 25 port 255
#warn: counter RcvSwRelayErrors = 48545  (threshold 100) lid 25 port 255
#warn: counter XmtDiscards = 9789      (threshold 100) lid 25 port 255
Error check on lid 25 (Sun DCS 72 QDR FC switch o4nm2-72p-2) port all:  FAILED
#warn: counter RcvSwRelayErrors = 56839  (threshold 100) lid 25 port 28
Error check on lid 25 (Sun DCS 72 QDR FC switch o4nm2-72p-2) port 28:  FAILED
#warn: counter RcvSwRelayErrors = 56839  (threshold 100) lid 25 port 9
Error check on lid 25 (Sun DCS 72 QDR FC switch o4nm2-72p-2) port 9:  FAILED
#warn: counter XmtDiscards = 9714      (threshold 100) lid 25 port 1
Error check on lid 25 (Sun DCS 72 QDR FC switch o4nm2-72p-2) port 1:  FAILED
.
.
.
## Summary: 6 nodes checked, 0 bad nodes found
##          142 ports checked, 0 bad ports found
##          3 ports have errors beyond threshold
#
```

Note – The output in the example is just a portion of the full output.

Related Information

- [ibchecknet man page](#)
- [“ibcheckerrors Command” on page 27](#)
- [“ibchecknode Command” on page 31](#)
- [“ibcheckport Command” on page 32](#)
- [“ibnetdiscover Command” on page 56](#)

ibchecknode Command

Validates InfiniBand nodes and reports errors.

Syntax

```
ibchecknode -v [-h] [-N] [-G] [-C ca_name] [-P ca_port] [-t timeout]  
[lid | dr_path | guid]
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unit identifier.

Description

This InfiniBand command checks node connectivity and performs a simple check to verify the functionality of the specified node.

Note – The port address is a LID, unless the `-G` option is used to specify a GUID address.

Options

The following table describes the options to the `ibchecknode` command and their purposes:

Option	Purpose
-h	Provides help.
-N	Uses mono mode instead of color mode.
-G	Uses the port GUID address.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to check if LID 24 is active with the `ibchecknode` command.

```
# ibchecknode -v 24
Node check lid 24: OK
#
```

Related Information

- [ibchecknode man page](#)
- [“ibaddr Command” on page 26](#)
- [“smpquery Command” on page 94](#)

ibcheckport Command

Validates InfiniBand ports and reports errors.

Syntax

```
ibcheckport [-h] [-v] [-N] [-G] [-C ca_name] [-P ca_port] [-t timeout]  
lid|guid port
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unit identifier.
- *port* is the port being validated.

Description

This InfiniBand command checks port connectivity and performs simple sanity checks for the specified port.

Note – The port address is a LID, unless the `-G` option is used to specify a GUID address.

Options

The following table describes the options to the `ibcheckport` command and their purposes:

Option	Purpose
<code>-h</code>	Provides help.
<code>-v</code>	Provides verbose output.
<code>-N</code>	Uses mono mode instead of color mode.
<code>-G</code>	Uses the port GUID address.
<code>-C</code>	Uses the specified channel adapter name.
<code>-P</code>	Uses the specified channel adapter port.
<code>-t</code>	Overrides the default timeout.

Example

The following example shows how to check port 8 on LID 24 with the `ibcheckport` command.

```
# ibcheckport -v 24 8
Port check lid 24 port 8: OK
#
```

Related Information

- [ibcheckport man page](#)
- [“getportstatus Command” on page 13](#)
- [“ibaddr Command” on page 26](#)
- [“smpquery Command” on page 94](#)

ibcheckportstate Command

Validates an InfiniBand port.

Syntax

```
ibcheckportstate -v [-G] [-h] [-N] [-C ca_name] [-P ca_port] [-t timeout] [lid|guid] port
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unit identifier.
- *port* is the port being validated.

Description

This InfiniBand command checks the specified port for the logical (Active) and physical (LinkUp) states.

Note – The port address is a LID, unless the `-G` option is used to specify a GUID address.

Options

The following table describes the options to the `ibcheckportstate` command and their purposes:

Option	Purpose
<code>-G</code>	Uses the port GUID address.
<code>-h</code>	Provides help.
<code>-N</code>	Uses mono mode instead of color mode.
<code>-C</code>	Uses the specified channel adapter name.
<code>-P</code>	Uses the specified channel adapter port.
<code>-t</code>	Overrides the default timeout.

Example

The following example shows how to check port 8 on LID 24 with the `ibcheckportstate` command.

```
# ibcheckportstate -v 24 8
Port check lid 24 port 8: OK
#
```

Related Information

- [ibcheckportstate man page](#)
- [“getportstatus Command” on page 13](#)
- [“ibaddr Command” on page 26](#)
- [“smpquery Command” on page 94](#)

ibcheckportwidth Command

Validates InfiniBand ports for 1x (or wider) link width.

Syntax

```
ibcheckportwidth -v [-G] [-h] [-N] [-C ca_name] [-P ca_port] [-t  
timeout] [lid|guid] port
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *guid* is the global unit identifier.
- *port* is the port being validated.

Description

This InfiniBand command checks connectivity and if the specified port is at 1x (or wider) link width.

Note – The port address is a LID, unless the `-G` option is used to specify a GUID address.

Options

The following table describes the options to the `ibcheckportwidth` command and their purposes:

Option	Purpose
-G	Uses the port GUID address.
-h	Provides help.
-N	Uses mono mode instead of color mode.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to check the width of port 8 on LID 24 with the `ibcheckportwidth` command.

```
# ibcheckportwidth -v 24 8
Port check lid 24 port 8: OK
#
```

Related Information

- [ibcheckportwidth man page](#)
- [“getportstatus Command” on page 13](#)
- [“ibaddr Command” on page 26](#)
- [“smpquery Command” on page 94](#)

ibcheckstate Command

Displays ports that are LinkUp but not Active.

Syntax

```
ibcheckstate [-h] [-N] [-v] [topology|-C ca_name] [-P ca_port] [-t timeout]
```

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that uses the topology file created by the `ibnetdiscover` command. The script scans the InfiniBand fabric to validate the port logical and physical state, and reports any ports that have a logical state other than Active or a physical state other than LinkUp.

Options

The following table describes the options to the `ibcheckstate` command and their purposes:

Option	Purpose
-h	Provides help.
-N	Uses mono mode instead of color mode.
-v	Provides verbose output.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to check the state of all ports with the `ibcheckstate` command.

```
# ibcheckstate -v
# Checking Switch: nodeguid 0x0021283a8620b0f0
Node check lid 25: OK
Port check lid 25 port 36: OK
Port check lid 25 port 35: OK
.
.
.
Port check lid 24 port 30: OK
Port check lid 24 port 29: OK

## Summary: 6 nodes checked, 0 bad nodes found
##          142 ports checked, 0 ports with bad state found
#
```

Note – The output in the example is just a portion of the full output.

Related Information

- [ibcheckstate man page](#)
- [“ibchecknode Command” on page 31](#)

- “[ibcheckportstate Command](#)” on page 34
- “[ibnetdiscover Command](#)” on page 56

ibcheckwidth Command

Finds 1x (or wider) links in the InfiniBand fabric.

Syntax

```
ibcheckwidth [-h] [-N] [-v] [topology] [-C ca_name] [-P ca_port] [-t timeout]
```

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that uses the topology file created by the `ibnetdiscover` command. The script scans the InfiniBand fabric to validate Active link widths and report those which are 1x (or wider) links.

Options

The following table describes the options to the `ibcheckwidth` command and their purposes:

Option	Purpose
-h	Provides help.
-N	Uses mono mode instead of color mode.
-v	Provides verbose output.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to check the 1x (or wider) links for all ports with the `ibcheckwidth` command.

```
# ibcheckwidth -v
# Checking Switch: nodeguid 0x0021283a8620b0f0
Node check lid 25: OK
Port check lid 25 port 36: OK
Port check lid 25 port 35: OK
Port check lid 25 port 34: OK
.
.
.
Port check lid 24 port 30: OK
Port check lid 24 port 29: OK
## Summary: 6 nodes checked, 0 bad nodes found
##          142 ports checked, 0 ports with 1x width in error found
#
```

Note – The output in the example is just a portion of the full output.

Related Information

- [ibcheckwidth man page](#)
- [“ibchecknode Command” on page 31](#)
- [“ibcheckportwidth Command” on page 35](#)
- [“ibnetdiscover Command” on page 56](#)

ibclearcounters Command

Clears port counters in the InfiniBand fabric.

Syntax

```
ibclearcounters [-h][topology|-C ca_name][-P ca_port][-t timeout]
```

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.

- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that clears the Performance Manager agent port counters by either discovering the InfiniBand fabric topology or using an existing topology file. The counters are:

- XmtData
- RcvData
- XmtPkts
- RcvPkts

Options

The following table describes the options to the `ibclearcounters` command and their purposes:

Option	Purpose
-h	Provides help.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to clear the Performance Manager agent port counters with the `ibclearcounters` command.

```
# ibclearcounters
## Summary: 6 nodes cleared 0 errors
#
```

Related Information

- [ibclearcounters man page](#)
- [“ibnetdiscover Command” on page 56](#)

- [“perfquery Command” on page 85](#)

ibclearerrors Command

Clears error counters in the InfiniBand fabric.

Syntax

```
ibclearerrors [-h] [-N] [topology | -C ca_name] [-P ca_port] [-t timeout]
```

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that clears the Performance Manager agent error counters in `PortCounters` by either discovering the InfiniBand fabric topology or using an existing topology file.

Options

The following table describes the options to the `ibclearerrors` command and their purposes:

Option	Purpose
-h	Provides help.
-N	Uses mono mode instead of color mode.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to clear all error counters with the `ibclearerrors` command.

```
# ibclearerrors
## Summary: 6 nodes cleared 0 errors
#
```

Related Information

- `ibclearerrors` man page
- [“ibnetdiscover Command” on page 56](#)
- [“perfquery Command” on page 85](#)

ibdatacounters Command

Queries the InfiniBand fabric for data counters.

Syntax

```
ibdatacounters [-b] [-h] [-N] [-v] [topology] [-C ca_name] [-P ca_port] [-t timeout]
```

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that uses the topology file created by the `ibnetdiscover` command. The script scans the InfiniBand fabric to validate the connectivity and reports the values of the data counters.

Options

The following table describes the options to the `ibdatacounters` command and their purposes:

Option	Purpose
-b	Enables brief mode. Reduced output is only if errors are present. Does not identify the errors.
-h	Provides help.
-N	Uses mono mode instead of color mode.
-v	Provides verbose output.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display the data counters for all ports with the `ibdatacounters` command.

```
# ibdatacounters

## Summary: 6 nodes checked, 0 bad nodes found
##           142 ports checked
#
```

Note – You might see more output than what is in the example.

Related Information

- [ibdatacounters man page](#)
- [“ibdatacounts Command” on page 45](#)
- [“ibnetdiscover Command” on page 56](#)

ibdatacounts Command

Displays InfiniBand fabric port data counters.

Syntax

```
ibdatacounts [-b] [-G] [-h] [-N] [-v] [-C ca_name] [-P ca_port] [-t timeout]  
lid|guid port
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unit identifier.
- *port* is the port being validated.

Description

This InfiniBand command returns the Performance Manager agent data counters from a specified port or node.

Note – The port address is a LID, unless the `-G` option is used to specify a GUID address.

Options

The following table describes the options to the `ibdatacounts` command and their purposes:

Option	Purpose
-b	Enables brief mode. Reduced output is only if errors are present. Does not identify the errors.
-G	Uses the port GUID address.
-h	Provides help.
-N	Uses mono mode instead of color mode.

Option	Purpose
-v	Provides verbose output.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display the data counters for LID 24, port 8 with the `ibdatacounts` command.

```
# ibdatacounts 24 8
# Port counters: Lid 24 port 8
XmtData:.....43272
RcvData:.....95256
XmtPkts:.....601
RcvPkts:.....1323
#
```

Related Information

- [ibdatacounts man page](#)
- [“ibaddr Command” on page 26](#)
- [“perfquery Command” on page 85](#)

ibdiagnet Command

Performs InfiniBand fabric diagnostic.

Syntax

```
ibdiagnet [-c count] [-v] [-r] [-o outputdir] [-t topology] [-s system] [-i device] [-p port] [-wt topology] [-pm] [-pc] [-P PM = value] [-lw 1x|4x|12x] [-ls 2.5|5|10] [-skip checks] [-load_db file] [-h] [-V]
```

where:

- *count* is the number of packets.
- *outputdir* is the output directory.

- *topology* is the topology file.
- *system* is the local system name.
- *device* is the index of the device connecting to the InfiniBand fabric.
- *port* is the port of the device.
- *PM* is the Performance Manager counter number.
- *value* is the threshold of the Performance Manager counter.
- *checks* is one or more strings that identifies the checks made:
 - `dup_guids`
 - `zero_guids`
 - `pm`
 - `logical_state`
 - `part`
 - `ipoib`
 - `all`
- *file* is the subnet database `.db` file.

Description

This InfiniBand command scans the InfiniBand fabric using directed route packets, extracting all the available information regarding the connectivity and devices. This command produces a set of files in the output directory. By default, the output directory is `/tmp`. The following table describes the files.

File Name	Description
<code>ibdiagnet.log</code>	Dump of all the application reports generated according to the provided flags.
<code>ibdiagnet.lst</code>	List of all the nodes, ports, and links in the fabric.
<code>ibdiagnet.fdb</code>	Dump of the unicast forwarding tables of the fabric switches.
<code>ibdiagnet.mcfdb</code>	Dump of the multicast forwarding tables of the fabric switches.
<code>ibdiagnet.masks</code>	In case of duplicate port/node GUIDs, these file include the map between masked GUIDs and real GUIDs.
<code>ibdiagnet.sm</code>	List of all the Subnet Manager (state and priority) in the fabric.
<code>ibdiagnet.pm</code>	Dump of the Performance Manager counters values, for the fabric links.

File Name	Description
ibdiagnet.pkey	Dump of the existing partitions and their member host ports.
ibdiagnet.mcg	Dump of the multicast groups, their properties, and member host ports.
ibdiagnet.db	Dump of the internal subnet database. This file can be loaded in later runs using the <code>-load_db</code> option.

During the discovery phase, the command also checks for duplicate node/port GUIDs in the InfiniBand fabric. If such an error is detected, it is displayed on the standard output.

After the discovery phase is completed, directed route packets are sent multiple times to detect possible problematic paths on which packets might be lost. A report of suspected bad links is displayed on the standard output.

If requested with the `-r` option, a full report of fabric qualities is displayed, including:

- Subnet Manager report
- Number of nodes and systems
- Hop-count information containing maximal hop-count, an example path, and a hop-count histogram
- All CA-to-CA paths traced
- Credit loop report
- MGID-MLID-HCAs multicast group and report
- Partitions report
- IPoIB report

Note – If the InfiniBand fabric includes only one CA, then CA-to-CA paths are not reported. Additionally, if a topology file is provided, the `ibdiagnet` command uses the names defined in the topology file for the output reports.

Options

The following table describes the options to the `ibdiagnet` command and their purposes:

Option	Purpose
<code>-c</code>	Sets the minimum number of packets sent across each link.
<code>-v</code>	Provides verbose output.

Option	Purpose
-r	Provides a report of fabric qualities.
-t	Specifies the topology file name.
-s	Specifies the local system name.
-i	In the case of multiple devices on the local system, this option specifies the index of the device of the port used to connect to the InfiniBand fabric.
-p	Specifies the local device port number used to connect to the InfiniBand fabric.
-o	Specifies the output directory.
-lw	Specifies the expected link width.
-ls	Specifies the expected link speed.
-pm	Dumps all the fabric link Performance Manager counters into <code>ibdiagnet.pm</code> .
-pc	Resets all the fabric link Performance Manager counters.
-P	Use the Performance Manager counter of PM set to the threshold of value.
-skip	Skips the executions of the selected checks. One or more checks can be specified.
-wt	Writes out the discovered topology into the given file.
-load_db	Loads subnet data from the given <code>.db</code> file and skips the subnet discovery stage. Note - Some checks require actual subnet discovery, and are disabled if <code>load_db</code> is specified. Those checks are duplicate/zero GUIDs, link state, and Subnet Manager status.
-h	Provides help.
-V	Displays the version information.

Example

The following example shows how to test the InfiniBand fabric with the `ibdiagnet` command. The command checks for 4x link width and 10 Gbyte/sec speed, and then dumps the Performance Manager counters and then clears them.

```
# ibdiagnet -lw 4x -ls 10 -pm -pc
Loading IBDIAGNET from: /usr/lib64/ibdiagnet1.2
-W- Topology file is not specified.
  Reports regarding cluster links will use direct routes.
Loading IBDM from: /usr/lib/ibdm1.2
-I- Using port 0 as the local port.
```

```

-I- Discovering ... 24 nodes (23 Switches & 1 CA-s) discovered.
-I-----
-I- Bad Guids/LIDs Info
-I-----
-I- No bad Guids were found
-I-----
-I- Links With Logical State = INIT
-I-----
-I- No bad Links (with logical state = INIT) were found
-I-----
-I- PM Counters Info
-I-----
-I- No illegal PM counters values were found
-I-----
-I- Links With links width != 4x (as set by -lw option)
-I-----
-I- No unmatched Links (with width != 4x) were found
-I-----
-I- Links With links speed != 10 (as set by -ls option)
-I-----
-I- No unmatched Links (with speed != 10) were found
-I-----
-I- Fabric Partitions Report (see ibdiagnet.pkey for a full hosts list)
-I-----
-I-----
-I- IPoIB Subnets Check
-I-----
-I- Subnet: IPv4 PKey:0x7fff QKey:0x00000b1b MTU:2048Byte rate:10Gbps SL:0x00
-W- No members found for group
-I-----
-I- Bad Links Info
-I- No bad link were found
-I-----
-I-----
-I- Stages Status Report:
  STAGE                Errors Warnings
  Bad GUIDS/LIDs Check          0      0
  Link State Active Check       0      0
  Performance Counters Report   0      0
  Specific Link Width Check     0      0
  Specific Link Speed Check     0      0
  Partitions Check              0      0
  IPoIB Subnets Check          0      1
Please see /tmp/ibdiagnet.log for complete log
-I-----
-I- Done. Run time was 68 seconds.
#

```

Related Information

- `ibdiagnet` man page
- [“ibdiagpath Command” on page 51](#)

ibdiagpath Command

Traces the InfiniBand fabric diagnostic path.

Syntax

```
ibdiagpath -n[src_name,]dst_name|-l[src_lid,]dst_lid|-d p1,p2,p3,...[-c
count][-v][-o outputdir][-t topology][-s system][-i device][-p port][-wt
topology][-pm][-pc][-P PM = value][-lw 1x|4x|12x][-ls
2.5|5|10][-h][-V]
```

where:

- *src_name* is the source port.
- *dst_name* is the destination port.
- *src_lid* is the source LID.
- *dst_lid* is the destination LID.
- *p1,p2,p3*,... is the directed route.
- *count* is the number of packets.
- *outputdir* is the output directory.
- *topology* is the topology file.
- *system* is the local system name.
- *device* is the index of the device connecting to the InfiniBand fabric.
- *port* is the port of the device.
- *PM* is the Performance Manager counter number.
- *value* is the threshold of the Performance Manager counter.

Description

This InfiniBand command traces a path between two end-points and provides information regarding the nodes and ports traversed along the path. The command uses device-specific health queries for the different devices encountered. The way the `ibdiagpath` command operates is determined from the addressing mode specified on the command line:

- If directed route addressing is used, the local node is the source node and the route to the destination port is known.
- If LID route addressing is used, the source and destination ports of a route are specified by their LIDs.

In LID route addressing, the actual path from the local port to the source port, and from the source port to the destination port, is defined by means of Subnet Management Linear Forwarding Table queries of the switch nodes along those paths.

Note – When the `ibdiagpath` command queries the performance counters along the path between the source and destination ports, it always traverses the LID route, regardless of whether a directed route is specified. If one or more links along the LID route are not in the `Active` state, the `ibdiagpath` command reports an error.

This command produces a set of files in the output directory. By default, the output directory is `/tmp`. The following table describes the files.

File Name	Description
<code>ibdiagpath.log</code>	Dump of all the application reports generated according to the provided flags.
<code>ibdiagpath.pm</code>	Dump of the Performance Manager counters values, for the fabric links.

Options

The following table describes the options to the `ibdiagpath` command and their purposes:

Option	Purpose
<code>-n</code>	Identifies the source and destination ports.
<code>-l</code>	Identifies the source and destination LIDs.
<code>-d</code>	Sets directed route from the local node to the destination node.
<code>-c</code>	Sets the minimum number of packets sent across each link.
<code>-v</code>	Provides verbose output.
<code>-t</code>	Specifies the topology file name.
<code>-s</code>	Specifies the local system name.
<code>-i</code>	In the case of multiple devices on the local system, this option specifies the index of the device of the port used to connect to the InfiniBand fabric.
<code>-p</code>	Specifies the local device port number used to connect to the InfiniBand fabric.

Option	Purpose
-o	Specifies the output directory.
-lw	Specifies the expected link width.
-ls	Specifies the expected link speed.
-pm	Dumps all the fabric link Performance Manager counters into <code>ibdiagpath.pm</code> .
-pc	Resets all the fabric link Performance Manager counters.
-P	Use the Performance Manager counter of PM set to the threshold of value.
-h	Provides help.
-V	Displays the version information.

Example

The following example shows how to display the route from LID 1 to LID 44 with the `ibdiagpath` command.

```
# ibdiagpath -l 1,44
Loading IBDIAGPATH from: /usr/lib64/ibdiagpath1.2
-W- Topology file is not specified.
  Reports regarding cluster links will use direct routes.
Loading IBDM from: /usr/lib/ibdm1.2
-I- Using port 0 as the local port.
-I-----
-I- Traversing the path from local to source
-I-----
-I- From: lid=0x0010 guid=0x0021283a8620b0f0 dev=48438 Port=2
-I- To:   lid=0x0009 guid=0x0021283a8620b0a0 dev=48438 Port=8
-I- From: lid=0x0009 guid=0x0021283a8620b0a0 dev=48438 Port=26
-I- To:   lid=0x0005 guid=0x0021283a8189a000 dev=48438 Port=14
-I- From: lid=0x0005 guid=0x0021283a8189a000 dev=48438 Port=20
-I- To:   lid=0x0001 guid=0x00141213180211a2 dev=48438 Port=31
-I-----
-I- Traversing the path from source to destination
-I-----
-I- From: lid=0x0001 guid=0x00141213180211a2 dev=48438 Port=17
-I- To:   lid=0x000a guid=0x001412134402f2b2 dev=48438 Port=36
-I- From: lid=0x000a guid=0x001412134402f2b2 dev=48438 Port=26
-I- To:   lid=0x0003 guid=0x00141213180211c2 dev=48438 Port=18
-I- From: lid=0x0003 guid=0x00141213180211c2 dev=48438 Port=33
-I- To:   lid=0x0004 guid=0x00141213180211d2 dev=48438 Port=28
-I- From: lid=0x0004 guid=0x00141213180211d2 dev=48438 Port=19
-I- To:   lid=0x0021 guid=0x0003ba7aa1a3b0a0 dev=48438 Port=35
-I- From: lid=0x0021 guid=0x0003ba7aa1a3b0a0 dev=48438 Port=2
```

```

-I- To:      lid=0x002c guid=0x0003ba7aa1a3b0f0 dev=48438 Port=8
-I-----
-I- PM Counters Info
-I-----
-I- No illegal PM counters values were found
-I-----
-I- Path Partitions Report
-I-----
-I- Source Port=31 lid=0x0001 guid=0x00141213180211a2 dev=48438 Port 31
  PKeys:0xffff
-I- Destination lid=0x002c guid=0x0003ba7aa1a3b0f0 dev=48438
  PKeys:0xffff
-I- Path shared PKeys: 0xffff
-I-----
-I- IPoIB Path Check
-I-----
-I- Subnet: IPv4 PKey:0x7fff QKey:0x00000b1b MTU:2048Byte rate:10Gbps SL:0x00
-I-----
-I- QoS on Path Check
-I-----
-I- The following SLs can be used:0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
-I-----
-I- Stages Status Report:
  STAGE                               Errors Warnings
  LFT Traversal: local to source      0         0
  LFT Traversal: source to destination 0         0
  Performance Counters Report         0         0
  Path Partitions Check                0         0
  Path IPoIB Check                     0         0
  QoS on Path Check                    0         0
Please see /tmp/ibdiagpath.log for complete log
-I-----
-I- Done. Run time was 0 seconds.
#

```

Related Information

- [ibdiagpath man page](#)
- [“ibdiagnet Command” on page 46](#)

ibhosts Command

Displays host nodes.

Syntax

```
ibhosts [-h] [topology] [-C ca_name] [-P ca_port] [-t timeout]
```

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that discovers the InfiniBand fabric topology or uses the existing topology file to extract the channel adapter nodes.

Options

The following table describes the options to the `ibhosts` command and their purposes:

Option	Purpose
-h	Provides help.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display the host node GUIDs with the `ibhosts` command.

```
# ibhosts
Ca      : 0x5080020000911314 ports 1 "nsn32-50 HCA-1"
Ca      : 0x5080020000911310 ports 1 "nsn32-20 HCA-1"
Ca      : 0x50800200008e532c ports 1 "ib-71 HCA-1"
Ca      : 0x50800200008e5328 ports 1 "ib-70 HCA-1"
Ca      : 0x50800200008296a4 ports 2 "ib-90 HCA-1"
Ca      : 0x50800200008296a0 ports 2 "ib-91 HCA-1"
Ca      : 0x508002000082dc34 ports 1 "ib-231 HCA-2"
```

```
Ca      : 0x0144f6c666b50100 ports 2 "nsn32-10 HCA-1"  
.  
.  
.  
#
```

Note – The output in the example is just a portion of the full output and varies for each InfiniBand topology.

Related Information

- [ibhosts man page](#)
- [“ibnetdiscover Command” on page 56](#)
- [“ibnodes Command” on page 59](#)

ibnetdiscover Command

Discovers the InfiniBand topology.

Syntax

```
ibnetdiscover [-d] [-e] [-v] [-s] [-l] [-g] [-H] [-S] [-R] [-C ca_name] [-P  
ca_port] [-t timeout] [-V] [--node-name-map map] [-p] [-h] [topology]
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *map* is the file name of the node name map.
- *topology* is the topology file.

Description

This InfiniBand command performs InfiniBand fabric discovery and outputs a human readable topology file. Nodes, node types, node descriptions, links, port numbers, port LIDs, and GUIDs are displayed. The output is directed to a topology file, if the file name is specified.

The output of the topology file follows this basic format for each node:

```
vendid=vendor_ID_in_hex
devid=device_ID_in_hex
and
sysimguid=GUID_in_hex
and/or
switchguid=GUID_in_hex (portGUID_in_hex)
Switch ports_total "type-nodeGUID_in_hex" # "NodeDescription" base port 0 lid LID lmc 0
or
caguid=GUID_in_hex
Ca ports_total "type-nodeGUID_in_hex" # "NodeDescription"
and
[port] "type-nodeGUID_in_hex" [port] (portGUID_in_hex) # "NodeDescription" lid LID widthspeed
[port] "type-nodeGUID_in_hex" [port] (portGUID_in_hex) # "NodeDescription" lid LID widthspeed
.
.
.
```

For example:

```
vendid=0x2c9
devid=0xbd36
sysimguid=0x21283a8620b0d3
switchguid=0x21283a8620b0d0 (21283a8620b0d0)
Switch 36 "S-0021283a8620b0d0" # "Sun DCS 72 QDR switch 1.2(LC)" base port 0
lid 23 lmc 0
[18] "S-0021283a8620b0e0"[16] # "Sun DCS 72 QDR switch 1.1(FC)" lid 24 4xQDR
.
.
.
```

Options

The following table describes the options to the `ibnetdiscover` command and their purposes:

Option	Purpose
-d	Sets the debug level. Can be used several times to increase the debug level.
-e	Displays send and receive errors.
-v	Provides verbose output.
-s	Shows more information.

Option	Purpose
-l	Lists the connected nodes.
-g	Shows the grouping and switch external ports correspondence.
-H	Lists the connected channel adapters.
-S	Lists the connected switches.
-R	Lists the connected routers.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.
-V	Displays the version information.
--node-name-map	Reads the node name map file.
-p	Returns a list of connected ports, including status information: <ul style="list-style-type: none"> • LID • portnum • GUID • link width • link speed • NodeDescription
-h	Provides help.

Example

The following example shows how to discover the InfiniBand fabric topology with the `ibnetdiscover` command.

```
# ibnetdiscover
#
# Topology file: generated on Thu Aug 13 22:40:51 2009
#
# Max of 2 hops discovered
# Initiated from node 0021283a8620b0f0 port 0021283a8620b0f0
vendid=0x2c9
devid=0xbd36
sysimgguid=0x21283a8620b0f3
switchguid=0x21283a8620b0f0(21283a8620b0f0)
Switch 36 "S-0021283a8620b0f0" # "Sun DCS 72 QDR FC switch o4nm2-72p-2"
enhanced port 0 lid 25 lmc 0
[36] "S-0021283a8620b0d0"[4] # "Sun DCS 72 QDR switch 1.2(LC)" lid 23 4xQDR
[35] "S-0021283a8620b0b0"[5] # "Sun DCS 72 QDR switch 1.2(LC)" lid 21 4xQDR
```

```
[34]      "S-0021283a8620b0d0"[6] # "Sun DCS 72 QDR switch 1.2(LC)" lid 23 4xQDR
.
.
.
vendid=0x2c9
devid=0x673c
sysimgguid=0x21283a8620b0e3
switchguid=0x21283a8620b0e0(21283a8620b0e0)
Switch 36 "S-0021283a8620b0e0" # "Sun DCS 72 QDR switch 1.1(FC)" base port 0
lid 24 lmc 0
[22]      "S-0021283a8620b0c0"[18] # "Sun DCS 72 QDR switch 1.2(LC)" lid 22 4xQDR
[24]      "S-0021283a8620b0c0"[16] # "Sun DCS 72 QDR switch 1.2(LC)" lid 22 4xQDR
.
.
.
#
```

Note – The output in the example is just a portion of the full output.

Related Information

- `ibnetdiscover` man page

ibnodes Command

Displays InfiniBand nodes in topology.

Syntax

```
ibnodes [-h] [topology | -C ca_name] [-P ca_port] [-t timeout]
```

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that discovers the InfiniBand fabric topology or uses the existing topology file to extract the InfiniBand nodes of the channel adapters, switches, and routers.

Options

The following table describes the options to the `ibnodes` command and their purposes:

Option	Purpose
-h	Provides help.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display the node GUIDs with the `ibnodes` command.

```
# ibnodes
Switch : 0x0021283a8620b0f0 ports 36 "Sun DCS 72 QDR FC switch o4nm2-72p-2"
enhanced port 0 lid 25 lmc 0
Switch : 0x0021283a8620b0d0 ports 36 "Sun DCS 72 QDR switch 1.2(LC)" base port
0 lid 23 lmc 0
Switch : 0x0021283a8620b0b0 ports 36 "Sun DCS 72 QDR switch 1.2(LC)" base port
0 lid 21 lmc 0
Switch : 0x0021283a8620b0a0 ports 36 "Sun DCS 72 QDR switch 1.2(LC)" base port
0 lid 20 lmc 0
Switch : 0x0021283a8620b0c0 ports 36 "Sun DCS 72 QDR switch 1.2(LC)" base port
0 lid 22 lmc 0
.
.
.
#
```

Note – The output in the example is just a portion of the full output.

Related Information

- `ibnodes` man page
- [“ibnetdiscover Command” on page 56](#)

ibportstate Command

Manages the state and link speed of an InfiniBand port.

Syntax

```
ibportstate [-d] [-D] [-e] [-G] [-h] [-s smlid] [-v] [-C ca_name] [-P  
ca_port] [-t timeout] lid|dr_path|guid port [op]
```

where:

- *smlid* is the Subnet Manager LID.
- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *guid* is the global unit identifier.
- *port* is the port being validated.
- *op* is the operation to perform on the port:
 - `enable`
 - `disable`
 - `reset`
 - `speed number` (where *number* is 1 for SDR, 2 for DDR, and 4 for QDR)
 - `query` (default)

Description

This InfiniBand command queries the logical and physical state of an InfiniBand port. The command can return the link width and speed of a switch port, as well as enabling, disabling, or resetting the port. The command can also set the link speed of any InfiniBand port.

Note – Speed changes are not affected until the port undergoes link renegotiation. Additionally, speed values are additive for enabling. For example, speed 7 is 2.5, 5.0, and 10.0 Gbyte/sec.

Options

The following table describes the options to the `ibportstate` command and their purposes:

Option	Purpose
-d	Sets the debug level. Can be used several times to increase the debug level.
-D	Uses the directed path address. The path is a comma delimited sequence of out ports.
-e	Displays send and receive errors.
-G	Uses the port GUID address.
-h	Provides help.
-s	Uses <i>smlid</i> as the target LID for Subnet Manager or Subnet Administrator queries.
-v	Provides verbose output.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to query the state and link speed of LID 24, port 8 with the `ibportstate` command.

```
# ibportstate 24 8
PortInfo:
# Port info: Lid 24 port 8
LinkState:.....Active
PhysLinkState:.....LinkUp
LinkWidthSupported:.....1X or 4X
LinkWidthEnabled:.....1X or 4X
LinkWidthActive:.....4X
LinkSpeedSupported:.....2.5 Gbps or 5.0 Gbps or 10.0 Gbps
```

```

LinkSpeedEnabled:.....2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedActive:.....10.0 Gbps
Peer PortInfo:
# Port info: Lid 24 DR path slid 25; dlid 65535; 0,8 port 2
LinkState:.....Active
PhysLinkState:.....LinkUp
LinkWidthSupported:.....1X or 4X
LinkWidthEnabled:.....1X or 4X
LinkWidthActive:.....4X
LinkSpeedSupported:.....2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedEnabled:.....2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedActive:.....10.0 Gbps
#

```

Related Information

- [ibportstate man page](#)
- [“disableswitchport Command” on page 8](#)
- [“enablesm Command” on page 9](#)
- [“i4reset Command” on page 14](#)
- [“listlinkup Command” on page 15](#)
- [“setlinkspeed Command” on page 19](#)

ibroute Command

Queries InfiniBand switch forwarding tables.

Syntax

```
ibroute [-d] [-a] [-n] [-D] [-e] [-G] [-h] [-M] [-s smlid] [-v] [-V] [-C
ca_name] [-P ca_port] [-t timeout] [lid|dr_path|guid [startlid [endlid]]]
```

where:

- *smlid* is the Subnet Manager LID.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *guid* is the global unit identifier.

- *startlid* is the starting local identifier.
- *endlid* is the ending local identifier.

Description

This InfiniBand command uses SMPs to display the forwarding tables for the specified switch LID and optionally, the LID range. By default, the range is all valid entries from 1 to `FDBTop`.

Options

The following table describes the options to the `ibroute` command and their purposes:

Option	Purpose
-a	Shows all LIDs in the range, including invalid entries.
-n	Does not try to resolve destinations.
-d	Sets the debug level. Can be used several times to increase the debug level.
-D	Uses the directed path address. The path is a comma delimited sequence of out ports.
-e	Displays send and receive errors.
-G	Uses the port GUID address.
-h	Provides help.
-M	Shows multicast forwarding tables.
-s	Uses <i>smlid</i> as the target LID for Subnet Manager or Subnet Administrator queries.
-v	Provides verbose output.
-V	Displays the version information.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display the forwarding table for LID 24 with the `ibroute` command.

```
# ibroute 24
Unicast lids [0x0-0x19] of switch Lid 24 guid 0x0021283a8620b0e0 (Sun DCS 72 QDR
switch 1.1(FC)):
  Lid  Out   Destination
      Port   Info
0x0014 001 : (Switch portguid 0x0021283a8620b0a0: 'Sun DCS 72 QDR switch
1.2(LC)')
0x0015 011 : (Switch portguid 0x0021283a8620b0b0: 'Sun DCS 72 QDR switch
1.2(LC)')
0x0016 002 : (Switch portguid 0x0021283a8620b0c0: 'Sun DCS 72 QDR switch
1.2(LC)')
.
.
.
0x0018 000 : (Switch portguid 0x0021283a8620b0e0: 'Sun DCS 72 QDR switch
1.1(FC)')
0x0019 001 : (Switch portguid 0x0021283a8620b0f0: 'Sun DCS 72 QDR FC switch
o4nm2-72p-2')
6 valid lids dumped
#
```

Note – The output in the example is just a portion of the full output.

Related Information

- `ibroute` man page
- [“ibtracert Command” on page 70](#)

ibstatus Command

Queries basic status of InfiniBand devices.

Syntax

```
ibstatus [-h] [devname[:ib_port]]...
```

where:

- *devname* is the InfiniBand device name.
- *ib_port* is the port number of the InfiniBand device.

Description

This InfiniBand command displays basic information retrieved from the local InfiniBand driver. Output of the command includes:

- LID
- SMLID
- port logical state
- link width
- port physical state

Options

The `-h` option provides help.

Example

The following example shows how to display the basic status from the local InfiniBand driver with the `ibstatus` command.

```
# ibstatus
Infiniband device 'is4_0' port 0 status:
  default gid:      fe80:0000:0000:0000:0021:283a:8620:b0f0
  base lid:         0x19
  sm lid:           0x19
  state:            4: ACTIVE
  phys state:       5: LinkUp
  rate:             40 Gb/sec (4X QDR)
#
```

Related Information

- `ibstatus` man page

ibswitches Command

Displays InfiniBand switch node in the topology.

Syntax

```
ibswitches [-h] [topology] [-C ca_name] [-P ca_port] [-t timeout]
```

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that discovers the InfiniBand fabric topology or uses an existing topology file to extract the switch nodes.

Options

The following table describes the options to the `ibswitches` command and their purposes:

Option	Purpose
-h	Provides help.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display the switch GUIDs with the `ibswitches` command.

```
# ibswitches
Switch : 0x0021283a8620b0f0 ports 36 "Sun DCS 72 QDR FC switch o4nm2-72p-2"
enhanced port 0 lid 25 lmc 0
Switch : 0x0021283a8620b0d0 ports 36 "Sun DCS 72 QDR switch 1.2(LC)" base port
0 lid 23 lmc 0
Switch : 0x0021283a8620b0b0 ports 36 "Sun DCS 72 QDR switch 1.2(LC)" base port
0 lid 21 lmc 0
Switch : 0x0021283a8620b0a0 ports 36 "Sun DCS 72 QDR switch 1.2(LC)" base port
0 lid 20 lmc 0
Switch : 0x0021283a8620b0c0 ports 36 "Sun DCS 72 QDR switch 1.2(LC)" base port
0 lid 22 lmc 0
Switch : 0x0021283a8620b0e0 ports 36 "Sun DCS 72 QDR switch 1.1(FC)" base port
0 lid 24 lmc 0
.
.
.
#
```

Note – The output in the example is just a portion of the full output and varies for each InfiniBand topology.

Related Information

- [ibswitches man page](#)
- [“ibnetdiscover Command” on page 56](#)
- [“ibnodes Command” on page 59](#)

ibsysstat Command

Displays system status of an InfiniBand address.

Syntax

```
ibsysstat [-d] [-e] [-G] [-h] [-s smlid] [-v] [-V] [-C ca_name] [-P  
ca_port] [-t timeout] [-o oui] [-S] lid|guid [op]
```

where:

- *smlid* is the Subnet Manager LID.
- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *oui* is the OUI number.
- *lid* is the local identifier.
- *guid* is the global unit identifier.
- *op* is the operation to perform on the node:
 - `ping` – Verify connectivity to the server.
 - `host` – Obtain host information from the server.
 - `cpu` – Obtain CPU information from the server.

Description

This InfiniBand command uses vendor supplied management datagrams to validate connectivity between InfiniBand nodes and return other information about the node. The command is a client-server, in that a remote node is configured as a server, while a local node performs as a client.

The command is first run locally on the server. The command is then run again locally on the client, pointing to the LID of the server. The ping repeats every second on the server, until you type the Ctrl-C key combination to exit the command.

Options

The following table describes the options to the `ibsysstat` command and their purposes:

Option	Purpose
<code>-d</code>	Sets the debug level. Can be used several times to increase the debug level.
<code>-e</code>	Displays send and receive errors.
<code>-G</code>	Uses the port GUID address.
<code>-h</code>	Provides help.
<code>-s</code>	Uses <i>smlid</i> as the target LID for Subnet Manager/Subnet Administrator queries.
<code>-v</code>	Provides verbose output.

Option	Purpose
-V	Displays the version information.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.
-o	Uses specified OUI number to multiplex vendor management datagrams.
-S	Starts in server mode.

Example

The following example shows how to retrieve system information about a remote node server from a local node client with the `ibsysstat` command. First configure the remote node server:

```
# ibsysstat -S
<CTRL-C to exit server>
#
```

Then run the command on the local node client:

```
# ibsysstat 1 ping
sysstat ping succeeded
# ibsysstat 1 host
nsn105-100.nsn.sfbay.sun.com
# ibsysstat 1 cpu
cpu 0: model AMD Opteron(tm) Processor 848 MHZ 2189.680
cpu 1: model AMD Opteron(tm) Processor 848 MHZ 2189.680
cpu 2: model AMD Opteron(tm) Processor 848 MHZ 2189.680
cpu 3: model AMD Opteron(tm) Processor 848 MHZ 2189.680
#
```

Related Information

- [ibsysstat man page](#)

ibtracert Command

Traces the InfiniBand path.

Syntax

```
ibtracert [-d] [-D] [-G] [-h] [-m mlid] [-s smlid] [-v] [-V] [-C  
ca_name] [-P ca_port] [-t timeout] [lid|dr_path|guid [startlid [endlid]]]
```

where:

- *mlid* is the multicast LID.
- *smlid* is the Subnet Manager LID.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *guid* is the global unit identifier.
- *startlid* is the starting LID for a range.
- *endlid* is the ending LID for a range.

Description

This InfiniBand command uses SMPs to trace the path from a source GUID or LID to a destination GUID or LID. Each responding hop in the path is displayed. The `-m` option enables multicast path tracing between source and destination nodes.

Options

The following table describes the options to the `ibtracert` command and their purposes:

Option	Purpose
<code>-d</code>	Sets the debug level. Can be used several times to increase the debug level.
<code>-D</code>	Uses the directed path address. The path is a comma delimited sequence of out ports.
<code>-G</code>	Uses the port GUID address.
<code>-h</code>	Provides help.
<code>-m</code>	Shows the multicast trace of the specified MLID.
<code>-s</code>	Uses <i>smlid</i> as the target LID for Subnet Manager or Subnet Administrator queries.

Option	Purpose
-v	Provides verbose output.
-V	Displays the version information.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display the path from LID 25 to LID 24 with the `ibtracert` command.

```
# ibtracert 25 24
From switch {0x0021283a8620b0f0} portnum 0 lid 25-25 "Sun DCS 72 QDR FC switch
o4nm2-72p-2"
[1] -> switch port {0x0021283a8620b0c0}[9] lid 22-22 "Sun DCS 72 QDR switch
1.2(LC) "
[2] -> switch port {0x0021283a8620b0e0}[8] lid 24-24 "Sun DCS 72 QDR switch
1.1(FC) "
To switch {0x0021283a8620b0e0} portnum 0 lid 24-24 "Sun DCS 72 QDR switch
1.1(FC) "
#
```

Related Information

- [ibtracert man page](#)
- [“ibroute Command” on page 63](#)

opensm Command

Runs the InfiniBand Subnet Manager and Subnet Administrator.

Syntax

```
opensm [-F filename] [-c filename] [-g guid] [-l lmc] [-p priority] [-smkey
SMKey] [-r] [-R engine] [-A] [-z] [-M filename] [-U filename] [-S filename] [-a
path] [-u path] [-X path] [-m path] [-o] [-s interval] [-t timeout] [-maxsmpps
```



```
number [-console[off|local|socket|loopback]] [-console-port port]  
[-i filename] [-f path] [-L size] [-e] [-P filename] [-N] [-Q[-Y filename]]  
[-y] [-B] [-I] [-v] [-V] [-D flags] [-dopt] [-h]
```

where:

- *filename* is the path and file for the respective option.
- *guid* is the global unit identifier.
- *lmc* is the LMC.
- *priority* is the priority of the instance, 0 (low)–13 (high).
- *SMKey* is the Subnet Manager encryption key (64 bits).
- *engine* is the routing algorithm.
- *path* is the path to a file used by the option.
- *interval* is the time in seconds between fabric sweeps.
- *timeout* is the timeout in milliseconds.
- *number* is the number of SMP management datagrams.
- *port* is the alternate Telnet port.
- *size* is the maximum size of the log file, in MB.
- *flag* is the log verbosity flag.
- *opt* is the debug option.

Description

This InfiniBand command initiates an instance of the OpenSM Subnet Manager and Subnet Administrator, which is required to initialize the InfiniBand hardware.

Note – Similarly, both the `enablesm` and `opensm` commands invoke an instance of the same Subnet Manager. Conversely, the `enablesm` command merely starts the Subnet Manager while the `opensm` command can configure the Subnet Manager.

OpenSM can be started in a default configuration with the `opensm` command. Options to the `opensm` command can configure the operation of the Subnet Manager to better fit the needs of your InfiniBand fabric.

OpenSM uses several files for operation and configuration. The default directory is `/etc/opensm`. The following table lists the files and their description.

File Name	Description
<code>opensm.conf</code>	Default OpenSM configuration file.

File Name	Description
<code>ib-node-name-map</code>	Default node name map file.
<code>partitions.conf</code>	Default partition configuration file.
<code>qos-policy.conf</code>	Default Quality of Service policy configuration file.
<code>prefix-routes.conf</code>	Default prefix routes file.

Options

The following table describes the options to the `opensm` command and their purposes:

Option	Purpose
<code>-F</code>	Uses the OpenSM configuration <i>filename</i> . The file <code>/etc/opensm/opensm.conf</code> is used by default.
<code>-c</code>	Creates the OpenSM configuration file and exit.
<code>-g</code>	Binds OpenSM to the local port of <i>guid</i> value. If no value for <i>guid</i> is provided, OpenSM will display available GUIDs, and wait for user input.
<code>-l</code>	Specifies the InfiniBand fabric's LMC value. The value of <i>lmc</i> must be within the range of 0 to 7. <i>lmc</i> values greater than 0 permit multiple paths between ports and are used only if the topology has multiple paths between ports. The number of LIDs assigned to each port is 2^{lmc} . By default, <i>lmc</i> is 0, so there is only one path between any two ports.
<code>-p</code>	Sets the priority of the Subnet Manager instance. The master Subnet Manager and handover conditions are based upon the priority and GUID of the Subnet Managers.
<code>-smkey</code>	Specifies the Subnet Manager's 64-bit encryption key.
<code>-r</code>	Reassigns LIDs to all end nodes. By default, OpenSM attempts to preserve existing LID assignments while resolving multiple use of the same LID. Note - Use of the <code>-r</code> option might disrupt fabric traffic.

Option	Purpose
-R	<p>Selects the routing algorithm to use. Multiple algorithms can be specified in the order of use upon failure. The list is comma delimited. Algorithms supported are:</p> <ul style="list-style-type: none"> • <code>minhop</code> – Min Hop. This default algorithm routes through the minimum number of hops with an optimized path. • <code>updn</code> – Up Down. Like <code>minhop</code>, but constrained with ranking rules. This algorithm is used if the fabric is not a pure fat-tree, where a deadlock might occur, due to a loop in the fabric. • <code>ftree</code> – Fat-Tree. This algorithm optimizes routing for a congestion-free shift communication pattern. The algorithm is for symmetrical or almost symmetrical fat-tree fabrics. Fat-tree routing is also constrained to ranking rules. • <code>lash</code> – LASH. This algorithm uses InfiniBand virtual layers to provide deadlock-free shortest-path routing while distributing the path between the layers. An advantage of the LASH algorithm is that it avoids use of a potentially congested root node. • <code>dor</code> – DOR. Based upon the Min Hop algorithm, but avoids port equalization except for redundant links between the same two switches. The algorithm provides deadlock-free routes for hypercubes and meshes, when the fabric is cabled as a hypercube or mesh.
-A	Enables the unicast routing cache, which prevents routing recalculations in the event that there was no topology change detected during a sweep or if recalculation is not really necessary. For example, when a host is rebooted.
-z	Forces the routing engine to make connectivity between root switches for InfiniBand architecture compliance. Use of the <code>-z</code> option might permit the occurrence of a deadlock.
-M	Specifies the name of the LID matrix dump file from where the Min Hop tables are loaded.
-U	Specifies the name of the LFT file from where switch forwarding tables are loaded.
-S	Specifies the name of the Subnet Administrator DB dump file from where the Subnet Administrator database is loaded.
-a	Sets the root nodes for the Up Down or Fat-Tree algorithm to the GUIDs in the provided file. One per line.
-u	Sets the compute nodes for the Fat-Tree algorithm to the GUIDs in the provided file. One GUID per line.
-X	Sets the order port GUIDs are routed for the Min Hop and Up Down algorithms to the GUIDs in the provided file. One per line.
-m	Sets a mapping of the IDs to the node GUIDs used by the Up Down algorithm in the provided file. File format is one GUID and ID per line.
-o	OpenSM configures the fabric once and exits. Port remains <code>Active</code> .
-s	Specifies the amount of time, in seconds, between fabric sweeps. Default is 10 seconds. An <i>interval</i> of 0 disables sweeps.
-t	Overrides the default timeout.
-maxsmps	Specifies the maximum number of outstanding VL15 SMP management datagrams at any one time. Default is 4. A <i>number</i> of 0 permits unlimited SMP management datagrams.

Option	Purpose
-console	<p>Sets the state of the OpenSM console. States are the following:</p> <ul style="list-style-type: none"> • off – This is the default. • local • socket • loopback <p>Note - The <code>socket</code> and <code>loopback</code> states are only available if OpenSM was built with the <code>--enable-console-socket</code> option.</p>
-console-port	Specifies an alternative Telnet port for the <code>socket</code> state.
-i	Defines a set of node GUID and port pairs to be ignored by the link load equalization algorithm. Pairs are provided in the <i>filename</i> .
-f	Instructs OpenSM where to send the log file. Default is <code>/var/log/opensm.log</code> . To direct the log to standard output, set <i>path</i> to <code>stdout</code> .
-L	Sets the maximum <i>size</i> of the log file in MB. The log file is truncated when reaching this limit, so the newest entries are lost.
-e	Deletes the log file.
-P	Defines the optional partition configuration <i>filename</i> . Default is <code>/etc/opensm/partitions.conf</code> .
-N	Disables partition enforcement on switch external ports.
-Q	Enables Quality of Service setup.
-Y	<p>Specifies the optional Quality of Service policy <i>filename</i>. Default is <code>/etc/opensm/qos-policy.conf</code>. The policy file contains a list of configuration parameters. The parameters have the following format: <i>qostypeparameter</i>.</p> <p>Where <i>type</i> is one of the following:</p> <ul style="list-style-type: none"> • nothing – The parameter affects the entire fabric. • <code>_ca</code> – The parameters are set for the channel adapters. • <code>_rtr</code> – The parameters are set for the routers. • <code>_swn</code> – The parameters are set for switch port <i>n</i>. • <code>_swe</code> – The parameters are set for switch external ports. <p>Where <i>parameter</i> is one of the following:</p> <ul style="list-style-type: none"> • <code>_max_vls value</code> – The maximum number of virtual lanes in the fabric. • <code>_high_limit value</code> – The limit of the High Priority component of the virtual lane arbitration table (IBA 7.6.9). • <code>_vlarb_low data</code> – The low priority virtual lane arbitration table template. <i>data</i> is 15 pairs of virtual lane and weight (<i>x:y</i>), separated by commas. • <code>_vlarb_high data</code> – The high priority virtual lane arbitration table template. <i>data</i> is 15 pairs of virtual lane and weight (<i>x:y</i>), separated by commas. • <code>_sl2vl data</code> – The SL2VL mapping table (IBA 7.6.6) template. <i>data</i> is a comma delimited sequence of virtual lanes corresponding to service levels 0 through 15.
-y	Prevents OpenSM from exiting upon fatal initialization issues. For example, duplicate GUIDS or 12x links with poorly configured lane reversal.

Option	Purpose
-B	Runs OpenSM in the background as a daemon.
-I	Starts OpenSM in an inactive state.
-v	Increases the log verbosity.
-V	Sets log verbosity to maximum. The same as -D 0xFF.
-D	<p>Sets the log verbosity according to the following bit <i>flags</i>:</p> <ul style="list-style-type: none"> • 0x01 – ERROR (error messages) • 0x02 – INFO (basic messages, low volume) • 0x04 – VERBOSE (moderate volume) • 0x08 – DEBUG (diagnostic, high volume) • 0x10 – FUNCS (function entry and exit, very high volume) • 0x20 – FRAMES (dumps all SMP and GMP frames) • 0x40 – ROUTING (dumps FDB routing information) <p>The default is ERROR + INFO, or 0x03. The option -D 0 turns off all log messaging.</p> <p>Note - High verbosity levels might require increasing the transaction timeout with the -t <i>timeout</i> option.</p>
-d	<p>Specifies debug and an option to the debug. The value of -dopt is as follows:</p> <ul style="list-style-type: none"> • -d0 – Ignores other Subnet Manager nodes. • -d1 – Forces single threaded dispatching. • -d2 – Forces log flushing after each log message. • -d3 – Disables multicast support
-h	Displays usage information.

Example

The following example shows how to initiate the Subnet Manager using the Fat-Tree routing algorithm with the `opensm` command.

```
# opensm -R ftree -A -a /etc/opensm/guid.txt -s 120
```

Related Information

- `opensm` man page
- “`osmtest` Command” on page 82
- “`sminfo` Command” on page 90

opensmd Daemon

Starts and stops the Subnet Manager within the management controller.

Syntax

```
/etc/init.d/opensmd start|stop|status
```

Description

Like the [“opensm Command” on page 72](#), the `opensmd` daemon initiates an instance of the OpenSM Subnet Manager. The daemon reads the `/etc/opensm/opensm.conf` configuration file upon startup, which it uses to configure the Subnet Manager. The `opensmd` daemon is more user-friendly because it is not necessary to provide elaborate command-line options and arguments.

The following table describes the more common parameters provided to the `opensmd` daemon in the `/etc/opensm/opensm.conf` file and their default values.

Parameter	Description	Default Value
<code>guid</code>	Port GUID on which the Subnet Manager is running.	<code>0x0000000000000000</code>
<code>m_key</code>	M_Key value sent to all ports qualifying all Set (PortInfo).	<code>0x0000000000000000</code>
<code>m_key_lease_period</code>	Lease period used for the M_Key in seconds.	<code>0</code>
<code>sm_key</code>	SM_Key value of the Subnet Manager used for authentication.	<code>0x0000000000000001</code>
<code>sa_key</code>	SM_Key value used to qualify received Subnet Administrator queries as trusted.	<code>0x0000000000000001</code>
<code>subnet_prefix</code>	Subnet prefix.	<code>0xfe80000000000000</code>
<code>lmc</code>	LMC value used on this subnet.	<code>0</code>
<code>lmc_esp0</code>	<code>lmc_esp0</code> determines whether LMC value is used for enhanced switch port 0. If <code>TRUE</code> , LMC value for subnet is used for <code>ESP0</code> . Otherwise, LMC value for <code>ESP0s</code> is 0.	<code>FALSE</code>

Parameter	Description	Default Value
packet_life_time	Maximum time a packet can live in a switch. Actual time is $4.096 \text{ usec} * 2^{\text{packet_life_time}}$. A value of 0x14 disables the mechanism.	0x12
vl_stall_count	Number of sequential packets dropped that cause the port to enter the VLStalled state. Do not set to 0.	0x07
leaf_vl_stall_count	Similar to vl_stall_count, this value is for switch ports driving a CA or router port. Do not set to 0.	0x07
head_of_queue_lifetime	Maximum time a packet can wait at the head of the transmission queue. Actual time is $4.096 \text{ usec} * 2^{\text{head_of_queue_lifetime}}$. A value of 0x14 disables the mechanism.	0x12
leaf_head_of_queue_lifetime	Maximum time a packet can wait at the head of the queue on a switch port connected to a CA or router port. Actual time is $4.096 \text{ usec} * 2^{\text{leaf_head_of_queue_lifetime}}$.	0x10
max_op_vls	Maximum operational virtual lanes.	5
force_link_speed	Force PortInfo:LinkSpeedEnabled on switch ports. Values are: <ul style="list-style-type: none"> • 0 – Make no change. • 1 – 2.5 Gbps • 3 – 2.5 Gbps or 5.0 Gbps • 5 – 2.5 Gbps or 10.0 Gbps • 7 – 2.5 Gbps or 5.0 Gbps or 10.0 Gbps • 2, 4, 6, 8–14 Reserved Default is 15: set to PortInfo:LinkSpeedSupported	15
subnet_timeout	Subnet timeout for all the ports. Actual timeout is $4.096 \text{ usec} * 2^{\text{subnet_timeout}}$.	18
local_phy_errors_threshold	Threshold of local PHY errors for sending trap 129.	0x08
overrun_errors_threshold	Threshold of credit overrun errors for sending trap 130.	0x08
partition_config_file	Partition configuration file.	/etc/opensm/partitions.conf

Parameter	Description	Default Value
no_partition_enforcement	Disable partition enforcement by switches.	FALSE
sweep_interval	Number of seconds between subnet sweeps (0 disables).	10
reassign_lids	If TRUE, reassign all LIDs.	FALSE
force_heavy_sweep	If TRUE, force heavy sweeps.	FALSE
sweep_on_trap	If TRUE, heavy sweep on trap. Successive identical traps (>10) are suppressed.	TRUE
port_profile_switch_nodes	If TRUE, count switches as link subscriptions.	FALSE
routing_engine	Routing engine: minhop, updn, file, ftree, lash, or dor. Multiple routing engines are comma delimited.	ftree
sm_priority	Subnet Manager priority determines the master. Range is 0 (lowest priority) to 13 (highest).	0
ignore_other_sm	If TRUE, ignore other Subnet Managers.	FALSE
sminfo_polling_timeout	Timeout in msec between consecutive polls of active master Subnet Manager.	1000
polling_retry_number	Number of failing polls of remote Subnet Manager to declare it dead.	3
honor_guid2lid_file	If TRUE, honor the guid2lid file when coming out of standby state.	FALSE
max_wire_smps	Maximum number of SMPs sent in parallel.	4
transaction_timeout	Maximum time in msec allowed for a transaction to complete.	200
max_msg_fifo_timeout	Maximum time in msec a message can stay in the incoming message queue. If two or more messages exceed this time in the queue, any Subnet Administrator request is returned with a BUSY status.	10000
single_thread	Use a single thread for handling Subnet Administrator queries.	FALSE
daemon	Daemon mode.	FALSE
sm_inactive	Deactivate the Subnet Manager.	FALSE
babbling_port_policy	Babbling Port Policy.	FALSE
log_flags	Log flags used.	0x03

Parameter	Description	Default Value
<code>force_log_flush</code>	Force flush of the log file after each log message.	FALSE
<code>log_file</code>	Log file to be used.	<code>/var/log/opensm.log</code>
<code>log_max_size</code>	Maximum size of the log file in MB. If overrun, log is restarted.	0
<code>accum_log_file</code>	If TRUE, accumulates the log over multiple OpenSM sessions.	TRUE
<code>dump_files_dir</code>	Directory for the OpenSM dump files.	<code>/var/log/</code>
<code>no_clients_rereg</code>	If TRUE, disables client reregistration.	FALSE
<code>disable_multicast</code>	If TRUE, multicast support and multicast routing is disabled.	FALSE
<code>exit_on_fatal</code>	If TRUE, opensm exits on fatal initialization issues.	TRUE
<code>console</code>	Console is either <code>off</code> or <code>local</code> .	<code>off</code>
<code>console_port</code>	Telnet port for console (default 10000).	10000
<code>prefix_routes_file</code>	Prefix routes file name.	<code>/etc/opensm/prefix-routes.conf</code>

Options

The following table describes the options to the `opensmd` daemon and their purposes:

Option	Purpose
<code>start</code>	Initiates the OpenSM Subnet Manager using the <code>/etc/opensm/opensm.conf</code> file to configure the Subnet Manager.
<code>stop</code>	Terminates the OpenSM Subnet Manager.
<code>status</code>	Provides the status of the Subnet Manager.

Example

The following example shows how to start the Subnet Manager with the `opensmd` daemon.

```
# /etc/init.d/opensmd start
Starting IB Subnet Manager.           [ OK ]
#
```

Related Information

- [“opensm Command” on page 72](#)

osmtest Command

Runs the InfiniBand Subnet Manager and administration test program.

Syntax

```
osmtest [-f c|a|v|s|e|f|m|q|t] [-w time] [-dopt] [-m lid] [-g guid] [-p] [-i filename] [-sopt] [-Mopt] [-t timeout] [-l path] [-v] [-vf flags] [-h]
```

where:

- *time* is the wait time in seconds.
- *opt* is a numeric option.
- *lid* is the local identifier.
- *guid* is the global unit identifier.
- *filename* is the path and name of the inventory file.
- *timeout* is the timeout in milliseconds.
- *path* is the path and name of the log file.
- *flags* is the log verbosity.

Description

The `osmtest` command creates an inventory file of all available nodes, ports, and path records, and compares the file with a previously created file. After the first run of OpenSM, you can create the inventory file with the `-f c` option. At a later time, you can run `osmtest -f v` to identify where there have been changes.

The `osmtest` command conducts the following tests:

- Multicast compliancy test
- Event forwarding test
- Service record registration test
- RMPP stress test
- Small Subnet Administrator queries test

Options

The following table describes the options to the `osmtest` command and their purposes:

Option	Purpose
-f	Directs <code>osmtest</code> to run a specific test flow: <ul style="list-style-type: none">• <code>c</code> – Creates an inventory file with all nodes, ports, and paths.• <code>a</code> – Runs all validation tests.• <code>v</code> – Validates only the given inventory file.• <code>s</code> – Runs service registration, deregistration, and lease tests.• <code>e</code> – Runs the event forwarding test.• <code>f</code> – Floods the Subnet Administrator with queries according to the stress mode.• <code>m</code> – Multicast flow.• <code>q</code> – Quality of Service information, dumps the VLab and SL2VL tables.• <code>t</code> – Runs trap 64/65 flow.
-w	Specifies the wait time for the trap 64/65 flow.
-d	Specifies debug and an option to the debug. The value of <code>-dopt</code> is as follows: <ul style="list-style-type: none">• <code>-d0</code> – Ignore other Subnet Manager nodes.• <code>-d1</code> – Force single threaded dispatching.• <code>-d2</code> – Force log flushing after each log message.• <code>-d3</code> – Disable multicast support
-m	Specifies the maximum LID to be searched during the inventory file build.
-g	Binds OpenSM to the local port of <code>guid</code> value. If no value for <code>guid</code> is provided, OpenSM will display available GUIDs and wait for user input.
-p	Displays a list of available port GUIDs to which <code>osmtest</code> could bind.
-i	Specifies the <i>filename</i> of the inventory file. The default is <code>osmtest.dat</code> .
-s	Runs the specified stress test instead of the normal test suite. The value of <code>-sopt</code> is as follows: <ul style="list-style-type: none">• <code>-s1</code> – Single management datagram response Subnet Administrator queries.• <code>-s2</code> – Multimangement datagram RMPP response Subnet Administrator queries.• <code>-s3</code> – Multimangement datagram RMPP path record Subnet Administrator queries.

Option	Purpose
-M	<p>Specifies length of multicast test. The value of <i>-Mopt</i> is as follows:</p> <ul style="list-style-type: none"> • -M1 – Short multicast flow, single mode. • -M2 – Short multicast flow, multiple mode. • -M3 – Long multicast flow, single mode. • -M4 – Long multicast flow, multiple mode. <p>In single mode, <i>osmtest</i> is tested alone. In multiple mode, <i>osmtest</i> is run with other applications using multicast with OpenSM.</p>
-t	Overrides the default timeout.
-l	Sets the log file to be <i>path</i> . The default action is to direct the log to standard output.
-v	Increases the log verbosity.
-V	Sets the log verbosity to maximum. The same as <i>-vf 0xFF</i>
-vf	<p>Sets the log verbosity according to the following bit <i>flags</i>:</p> <ul style="list-style-type: none"> • 0x01 – ERROR (error messages) • 0x02 – INFO (basic messages, low volume) • 0x04 – VERBOSE (moderate volume) • 0x08 – DEBUG (diagnostic, high volume) • 0x10 – FUNCS (function entry and exit, very high volume) • 0x20 – FRAMES (dumps all SMP and GMP frames) • 0x40 – ROUTING (dumps FDB routing information) <p>The default is ERROR + INFO, or 0x03. The option <i>-vf 0</i> turns off all log messaging.</p> <p>Note - High verbosity levels might require increasing the transaction timeout with the <i>-t timeout</i> option.</p>
-h	Displays usage information.

Example

The following example shows how to run all validation tests with the *osmtest* command.

```
# osmtest -f a
Command Line Arguments
Done with args
  Flow = All Validations
Aug 13 23:00:48 919475 [B7E7B6C0] 0x7f -> Setting log level to: 0x03
Aug 13 23:00:48 920209 [B7E7B6C0] 0x02 -> osm_vendor_init: 1000 pending umads
specified
using default guid 0x21283a8620b0f0
Aug 13 23:00:48 930634 [B7E7B6C0] 0x02 -> osm_vendor_bind: Binding to port
0x21283a8620b0f0
Aug 13 23:00:48 957644 [B7E7B6C0] 0x02 -> osmtest_validate_sa_class_port_info:
```

```

-----
SA Class Port Info:
  base_ver:1
  class_ver:2
  cap_mask:0x2602
  cap_mask2:0x0
  resp_time_val:0x10
-----
Aug 13 23:00:48 962010 [B7E7B6C0] 0x01 -> osmtest_validate_node_data: Checking
node 0x0021283a8620b0a0, LID 0x14
Aug 13 23:00:48 962226 [B7E7B6C0] 0x01 -> osmtest_validate_node_data: Checking
node 0x0021283a8620b0b0, LID 0x15
.
.
.
Aug 13 23:01:03 083751 [B7E7B6C0] 0x02 -> osmt_delete_service_by_name: Trying to
Delete service name: osmt.srvc.719885380.7487
Aug 13 23:01:03 084267 [B7E7B6C0] 0x02 -> osmtest_run: The event forwarding flow
is not implemented yet!
OSMTEST: TEST "All Validations" PASS
#

```

Note – The output in the example is just a portion of the full output.

Related Information

- `osmtest` man page

perfquery Command

Queries InfiniBand port counters.

Syntax

```
perfquery [-d] [-e] [-G] [-h] [-a] [-l] [-r] [-R] [-v] [-V] [-C ca_name] [-P
ca_port] [-t timeout] [lid|guid [[port][reset_mask]]]
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

- *lid* is the local identifier.
- *guid* is the global unit identifier.
- *port* is the port being queried.
- *reset_mask* is the two-byte mask.

Description

This InfiniBand command uses the performance management GMPs to acquire the `PortCounters` or `PortExtendedCounters` from the Performance Manager agent at the node or port specified.

Note – The data values retrieved from `PortCounters` and `PortExtendedCounters` are represented as octets divided by 4.

Note – Providing a *port* value of 255 ensures that the operation is performed on all ports.

Options

The following table describes the options to the `perfquery` command and their purposes:

Option	Purpose
-d	Sets the debug level. Can be used several times to increase the debug level.
-e	Shows extended port counters.
-a	Shows aggregate counters for all ports.
-G	Uses the port GUID address.
-h	Provides help.
-l	Loops through all ports.
-r	Resets the counters after reading.
-R	Resets counters only.
-v	Provides verbose output.
-V	Displays the version information.
-C	Uses the specified channel adapter name.

Option	Purpose
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display all port counters for LID 24, port 8 with the `perfquery` command.

```
# perfquery 24 8
# Port counters: Lid 24 port 8
PortSelect:.....8
CounterSelect:.....0x1b01
SymbolErrors:.....0
LinkRecovers:.....0
LinkDowned:.....0
RcvErrors:.....0
RcvRemotePhysErrors:.....0
RcvSwRelayErrors:.....0
XmtDiscards:.....0
XmtConstraintErrors:.....0
RcvConstraintErrors:.....0
LinkIntegrityErrors:.....0
ExcBufOverrunErrors:.....0
VL15Dropped:.....0
XmtData:.....75384
RcvData:.....87696
XmtPkts:.....1047
RcvPkts:.....1218
#
```

Related Information

- [perfquery man page](#)
- [“ibcheckerrors Command” on page 27](#)
- [“ibdatacounters Command” on page 43](#)
- [“ibdatacounts Command” on page 45](#)

saquery Command

Queries InfiniBand fabric administration attributes.

Syntax

```
saquery [-h] [-d] [-p] [-N] [-D] [-S] [-I] [-L] [-l] [-G] [-O] [-U] [-c] [-s]
[-g] [-m] [-x] [-C ca_name] [-P ca_port] [-t timeout] [--src-to-dst
source:destination] [--sgid-to-dgid source-destination] [name | lid | guid]
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unit identifier.
- *name* is the query name described in the following table:

Query Names	Alias for <i>name</i>	Format
ClassPortInfo	CPI	
NodeRecord	NR	
PortInfoRecord	PIR	
SL2VLTableRecord	SL2VL	[[<i>lid</i>]/[<i>in_port</i>]/[<i>out_port</i>]]
PKeyTableRecord	PKTR	[[<i>lid</i>]/[<i>port</i>]/[<i>block</i>]]
VLArbitrationTableRecord	VLAR	[[<i>lid</i>]/[<i>port</i>]/[<i>block</i>]]
InformInfoRecord	IIR	
LinkRecord	LR	[[<i>from_lid</i>]/[<i>from_port</i>]] [[<i>to_lid</i>]/[<i>to_port</i>]]
ServiceRecord	SR	
PathRecord	PR	
MCMemberRecord	MCMR	
LFTRRecord	LFTR	[[<i>lid</i>]/[<i>block</i>]]
MFTRRecord	MFTR	[[<i>mlid</i>]/[<i>position</i>]/[<i>block</i>]]

Description

This InfiniBand command performs the selected Subnet Administrator query. Node records are queried by default.

Options

The following table describes the options to the `saquery` command and their purposes:

Option	Purpose
-h	Provides help.
-d	Sets the debug level. Can be used several times to increase the debug level.
-p	Displays the PathRecord information.
-N	Displays the NodeRecord information.
-D	Displays the NodeDescriptions of channel adapters only.
-S	Displays ServiceRecord information.
-I	Displays InformInfoRecord information.
-L	Returns the LIDs of the specified name.
-l	Returns the unique LID of the specified name.
-G	Returns the GUIDs of the specified name.
-O	Returns the name of the specified LID.
-U	Returns the name of the specified GUID.
-G	Uses the port GUID address.
-c	Displays the Subnet Administrator class port information.
-s	Returns the PortInforRecords with the <code>isSM</code> or <code>isSMdisabled</code> capability mask bit enabled.
-g	Displays multicast group information.
-m	Displays multicast member information. If a group is specified, provides only the GUID and node description for each entry.
-x	Displays LinkRecord information.
--src-to-dst	Displays a PathRecord for <i>source:destination</i> , where <i>source</i> and <i>destination</i> are either node names or LIDs.
--sgid-to-dgid	Displays a PathRecord for <i>source-destination</i> , where <i>source</i> and <i>destination</i> are GUIDs in an IPv6 format acceptable to <code>inet_pton</code> .
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to produce a node record dump of every LID in the InfiniBand fabric with the saquery command.

```
# saquery
NodeRecord dump:
  lid.....0x14
  reserved.....0x0
  base_version.....0x1
  class_version.....0x1
  node_type.....Switch
  num_ports.....0x24
  sys_guid.....0x0021283a8620b0a3
  node_guid.....0x0021283a8620b0a0
  port_guid.....0x0021283a8620b0a0
  partition_cap.....0x8
  device_id.....0xBD36
  revision.....0xA0
  port_num.....0x8
  vendor_id.....0x2C9
  NodeDescription.....Sun DCS 72 QDR switch 1.2(LC)
NodeRecord dump:
  lid.....0x15
  reserved.....0x0
.
.
.
#
```

Note – The output in the example is just a portion of the full output.

Related Information

- saquery man page

sminfo Command

Queries the InfiniBand SMInfo attribute.

Syntax

```
sminfo [-d] [-e] -s state -p priority -a activity [-D] [-G] [-h] [-V] [-C  
ca_name] [-P ca_port] [-t timeout] smlid | smdr_path
```

where:

- *state* is the state for the Subnet Manager.
- *priority* is the priority.
- *activity* is the activity count.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *smlid* is the Subnet Manager local identifier.
- *smdr_path* is the directed path for the Subnet Manager.

Description

This InfiniBand command conducts a query of the Subnet Manager and outputs the information in a human readable format. The target Subnet Manager is identified in the local port information, or it is specified by the *smlid* or *smdr_path*.

Note – Using the `sminfo` command for other than simple queries might fault the target Subnet Manager.

Options

The following table describes the options to the `sminfo` command and their purposes:

Option	Purpose
-d	Sets the debug level. Can be used several times to increase the debug level.
-D	Uses the directed path address. The path is a comma delimited sequence of out ports.
-e	Displays send and receive errors.

Option	Purpose
-s	Sets the Subnet Manager state: <ul style="list-style-type: none"> • 0 – Not active. • 1 – Discovering. • 2 – Standby. • 3 – Master.
-p	Sets the priority, (0–13).
-a	Sets the activity count.
-G	Uses the port GUID address.
-h	Provides help.
-V	Displays the version information.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display the SMInfo with the `sminfo` command.

```
# sminfo
sminfo: sm lid 25 sm guid 0x21283a8620b0f0, activity count 25950 priority 13
state 3 SMINFO_MASTER
#
```

Related Information

- [sminfo man page](#)
- [“smpdump Command” on page 92](#)

smpdump Command

Dumps the InfiniBand fabric management attributes.

Syntax

```
smpdump [-s] [-D] [-h] [-V] [-C ca_name] [-P ca_port] [-t timeout]  
lid | dr_path attr [mod]
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *attr* is the InfiniBand architecture attribute ID for Subnet Manager attribute.
- *mod* is the InfiniBand architecture modifier for Subnet Manager attribute.

Description

This InfiniBand command is a general purpose SMP utility that returns Subnet Manager attributes from a specified SMA. The output is in hexadecimal.

Options

The following table describes the options to the `smpdump` command and their purposes:

Option	Purpose
-D	Uses the directed path address. The path is a comma delimited sequence of out ports.
-e	Displays send and receive errors.
-g	Shows the GID address only.
-h	Provides help.
-V	Displays the version information.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display a raw dump of the InfiniBand management attributes for the directed path through ports 0 to 2, with attribute ID 0x19 with the `smpdump` command.

```
# smpdump D 0,2 0x19 1
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 1f03 0302
7452 0047 5040 0408 0805 f240 0000 0000
0000 0000 0088 0055 00ff ffff 0000 0000
SMP status: 0x8000
#
```

Related Information

- `smpdump` man page
- [“smpquery Command” on page 94](#)

smpquery Command

Queries InfiniBand fabric management attributes.

Syntax

```
smpquery [-d] [-D] [-e] [-G] [-h] [-v] [-V] [-C ca_name] [-P ca_port] [-t timeout] op lid [dr_path] [guid] [params]
```

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *op* is the supported operation.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *guid* is the global unit identifier.
- *params* is the parameter of the operation.

Description

This InfiniBand command permits a subset of the standard SMP queries, including the following:

- Node information
- Node description
- Switch information
- Port information

Output is in human-readable format.

Supported operations and parameters are as follows:

- `nodeinfo addr`
- `nodedesc addr`
- `portinfo addr [portnum]`
- `switchinfo addr`
- `pkeys addr [portnum]`
- `sl2v1 addr [portnum]`
- `vlarb addr [portnum]`
- `guids addr`

where:

- *addr* is the address.
- *portnum* is the port number.

Options

The following table describes the options to the `smpquery` command and their purposes:

Option	Purpose
-d	Sets the debug level. Can be used several times to increase the debug level.
-D	Uses the directed path address. The path is a comma delimited sequence of out ports.
-e	Displays send and receive errors.
-G	Uses the port GUID address.
-h	Provides help.
-v	Provides verbose output.

Option	Purpose
-V	Displays the version information.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display node information for LID 24 with the `smpquery` command.

```
# smpquery nodeinfo 24
# Node info: Lid 24
BaseVers:.....1
ClassVers:.....1
NodeType:.....Switch
NumPorts:.....36
SystemGuid:.....0x0021283a8620b0e3
Guid:.....0x0021283a8620b0e0
PortGuid:.....0x0021283a8620b0e0
PartCap:.....8
DevId:.....0xbd36
Revision:.....0x00000a0
LocalPort:.....1
VendorId:.....0x0002c9
#
```

Related Information

- [smpquery man page](#)
- [“saquery Command” on page 87](#)
- [“smpdump Command” on page 92](#)

Index

C

checkboot command, 2
checkpower command, 3
checkvoltages command, 4
command
 hardware, 1
 checkboot, 2
 checkpower, 3
 checkvoltages, 4
 connector, 5
 disablesm, 7
 disableswitchport, 8
 enablesm, 9
 enableswitchport, 10
 env_test, 11
 getfanspeed, 12
 getportstatus, 13
 i4reset, 14
 listlinkup, 15
 managementreset, 16
 nm2version, 17
 readchfru, 18
 setlinkspeed, 19
 setsmpriority, 20
 showtemps, 22
 showunhealthy, 23
InfiniBand, 23
 ibaddr, 26
 ibcheckerrors, 27
 ibchecknet, 29
 ibchecknode, 31
 ibcheckport, 32
 ibcheckportstate, 34
 ibcheckportwidth, 35
 ibcheckstate, 37
 ibcheckwidth, 39
 ibclearcounters, 40
 ibclearerrors, 42

 ibdatacounters, 43
 ibdatacounts, 45
 ibdiagnet, 46
 ibdiagpath, 51
 ibhosts, 54
 ibnetdiscover, 56
 ibnodes, 59
 ibportstate, 61
 ibroute, 63
 ibstatus, 65
 ibswitches, 67
 ibsysstat, 68
 ibtracert, 70
 opensm, 72
 opensmd, 78
 osmtest, 82
 perfquery, 85
 saquery, 87
 sminfo, 90
 smpdump, 92
 smpquery, 94
connector command, 5
 registers, 5

D
disablesm command, 7
disableswitchport command, 8

E

enablesm command, 9
enablesm parameters, 9, 78
enableswitchport command, 10
env_test command, 11

G

getfanspeed command, 12
getportstatus command, 13

H

hardware
 commands, 1

I

i4reset command, 14
ibaddr command, 26
ibcheckerrors command, 27
ibchecknet command, 29
ibchecknode command, 31
ibcheckport command, 32
ibcheckportstate command, 34
ibcheckportwidth command, 35
ibcheckstate command, 37
ibcheckwidth command, 39
ibclearcounters command, 40
ibclearerrors command, 42
ibdatacounters command, 43
ibdatacounts command, 45
ibdiagnet command, 46
 support files, 47
ibdiagpath command, 51
 support files, 51
ibhosts command, 54
ibnetdiscover command, 56
 topology file, 56
ibnodes command, 59
ibportstate command, 61
ibroute command, 63
ibstatus command, 65
ibswitches command, 67
ibsysstat command, 68
ibtracert command, 70
InfiniBand commands, 23

L

listlinkup command, 15

M

managementreset command, 16

N

nm2version command, 17

O

opensm command, 72
 options, 74
 support files, 73
opensm parameters, 9, 78
opensmd daemon, 78
osmtest command, 82

P

perfquery command, 85

R

readchfru command, 18

S

saquery command, 87
setlinkspeed command, 19
setsmpriority command, 20
showtemps command, 22
showunhealthy command, 23
sminfo command, 90
smpdump command, 92
smpquery command, 94

U

understanding
 commands, 1
 hardware commands, 1
 InfiniBand
 commands, 23