BROADCOM®

Emulex® Drivers for Solaris for OneConnect® Adapters

User Guide

Version 11.2
December 30, 2016

DRVSol-OCA-UG112

Corporate Headquarters	Website
San Jose, CA	www.broadcom.com

Table of Contents

Chapter 1: Introduction

This document provides information for using the Emulex® Fibre Channel over Ethernet (FCoE) out-of-box drivers for Solaris StorEdge SAN Foundation Software (SFS), which is part of the SFS (Leadville) stack. The module name for this FCoE driver is `elxfc`.

This document also provides information for using the Emulex network interface card (NIC) out-of-box driver for the Solaris operating system, which operates the NIC function of the Emulex OneConnect® adapters. The module name for this NIC driver is `elxnic`.

1.1   Supported Operating Systems

One of the following operating systems must be installed:

- Solaris 10 (x64)
- Solaris 11.2 (x64)
- Solaris 11.3 (x64)

> **NOTE**   SPARC is not supported.

1.2   Supported Adapters and Firmware

These drivers support the following Emulex OneConnect converged network adapters (CNAs):

- OCe11000-series adapters
- OCe14000-series adapters

For compatible firmware versions, refer to the Downloads page at http://www.broadcom.com for the specific adapter.

The NIC driver supports the Solaris Ethernet stack, including NIC virtualization (Crossbow), virtual switch (vSwitch), and virtual router (vRouter).

1.3   Abbreviations

APIC	advanced programmable interrupt controller
CLI	command line interface
CNA	Converged Network Adapter
CPU	central processing unit
DHCHAP	Diffie-Hellman Challenge Handshake Authentication Protocol
ELS	Extended Link Service
FC	Fibre Channel
FCA	Fibre Channel adapter
FCIO	FC input/output
FC-IP	Fibre Channel over IP
FCoE	Fibre Channel over Ethernet
FCTL	FC transport library
FW	firmware
Gb/s	gigabits per second
HW	hardware
IOCTL	input/output control
IRM	interrupt resource management
IRQ	interrupt request
LACP	Link Aggregation Control Protocol
MSI-X	message signaled interrupts – extended
MTU	maximum transmission unit
NIC	network interface card
NPIV	N_Port ID Virtualization
PCI	Peripheral Component Interconnect
PCI-X	PCI Extended
POST	power-on self-test
SCSI	Small Computer Systems Interface
SFS	SAN Foundation Software
SLI	Service Level Interface
tar	tape archive
UMC	universal multichannel
VF	virtual function
VLAN	virtual local area network
VM	virtual machine
VPort	virtual port
vRouter	virtual router
vSwitch	virtual switch
WWNN	World Wide Node Name

WWPN   World Wide Port Name

Chapter 2: Installing and Uninstalling the Solaris Drivers

This chapter details the procedures for installing and uninstalling the Solaris drivers.

2.1 Installing the Drivers for Solaris 10

2.1.1 FCoE Out-of-Box Driver (elxfc)

To install the FCoE out-of-box driver, perform these steps:

1. Log in as `root`.
2. Download the platform-specific FCoE out-of-box driver from the Broadcom® website to a convenient directory. The file is a regular tape archive (tar) file.
3. Extract the installation image from the tar file. For example:
   ```
   cd <location of driver package>
   tar xvf elxfc_kit-11.x.xx.yy-s10-i386.tar
   ```
4. Install the driver kit. For example:
   ```
   cd <location of driver package>
   pkgadd -d ./
   ```
5. Reboot the system.

2.1.2 NIC Out-of-Box Driver (elxnic)

To install the NIC out-of-box driver, perform these steps:

1. Log in as `root`.
2. Download the platform-specific NIC out-of-box driver from the Broadcom website to a convenient directory. The file is a regular tar file.
3. Extract the installation image from the tar file. For example:
   ```
   cd <location of driver package>
   tar xvf elxnic_kit-11.x.xx.yy-s10-i386.tar
   ```
4. Install the driver kit. For example:
   ```
   cd <location of driver package>
   pkgadd -d./
   ```
5. Reboot the system.

2.2 Uninstalling the Drivers for Solaris 10

2.2.1 FCoE Driver

To uninstall the FCoE out-of-box driver, perform these steps:

1. Log in as `root`.
2. Remove the out-of-box driver by typing:
   ```
   pkgrm <OOB_pkg_name>
   ```

For example:
```
pkgrm elxfc
```

3. Reboot the system.

2.2.2 NIC Driver

To uninstall the NIC out-of-box driver, perform these steps:

1. Log in as *root*.
2. Remove the out-of-box driver by typing:
   ```
   pkgrm <nic_pkg_name>
   ```
   For example:
   ```
   pkgrm elxnic
   ```
3. Reboot the system.

2.3 Installing the Drivers for Solaris 11

This section details the procedures for installing the drivers for Solaris 11.

2.3.1 FCoE Out-of-Box Driver (elxfc)

You can use a scripted installation or a manual installation.

Scripted Installation

1. Log in as *root*.
2. Download the `p5p` tar file from the Broadcom website.
3. Untar the archive:
   ```
   $ tar -xvf elxfc_p5p-11.x.xx.yy-s11.tar
   ```
4. Run the installation script:
   ```
   $ ./elxfc_install
   ```
5. Reboot the system:
   ```
   $ reboot
   ```

Manual Installation

1. Log in as *root*.
2. Download the `p5p` tar file from the Broadcom website.
3. Extract the p5p archive from the tar file:
   ```
   $ tar -xvf elxfc_p5p-<version>.tar
   ```
4. Run the following command to install the package:
   ```
   $ pfexec pkg install -g elxfc_<version>.p5p
   ```
5. Reboot the system:
   ```
   $ reboot
   ```

2.3.2   NIC Out-of-Box Driver (elxnic)

1. Log in as *root*.
2. Download the `p5p` tar file from the Broadcom website.
3. Untar the archive:
   ```
   tar -xvf elxnic_p5p-11.x.xx.yy-s11.tar
   ```
4. Run the installation script:
   ```
   ./elxnic_install
   ```
5. Reboot the system:
   ```
   reboot
   ```

2.4   Uninstalling the Drivers for Solaris 11

2.4.1   FCoE Driver

You can uninstall the FCoE out-of-box driver by using scripted removal or manual removal.

2.4.1.1   Scripted Removal

1. Log in as *root*.
2. Run the following command to remove `elxfc` from the system:
   ```
   $ ./elxfc_remove
   ```
3. Reboot the system. Type
   ```
   $ reboot
   ```

The script is included in the downloadable `p5p` tar file that can be retrieved from the Broadcom website. The scripted uninstallation also cleans up the files in the installation directory.

2.4.1.2   Manual Removal

1. Log in as *root*.
2. Run the following command to remove `elxfc` from the system:
   ```
   $ pkg uninstall elxfc
   ```
3. Reboot the system:
   ```
   $ reboot
   ```

2.4.2   NIC Driver

To uninstall the NIC out-of-box driver, perform these steps:

1. Log in as *root*.
2. Remove the out-of-box driver by typing
   ```
   pkg uninstall <driver_name>
   ```
   For example:
   ```
   pkg uninstall elxnic
   ```
3. Reboot the system.

2.4.2.1 Scripted Removal

1. Log in as *root.*

2. Run the following command to remove `elxnic` driver by typing

   ```
   ./elxnic_remove
   ```

3. Reboot the system.

Chapter 3: Utilities for Solaris Drivers

Emulex provides the following utilities to facilitate configuring and using the Solaris drivers:

- `brcmadm` utility (included in the Fibre Channel adapter [FCA] utilities)
- OneCommand® Converged Network Adapter (CNA) Manager application

> **NOTE**   Solaris FCA utilities are installed automatically when you install the OneCommand CNA Manager application.

3.1   OneCommand CNA Manager Application

The OneCommand CNA Manager application provides the functionality of the `brcmadm` utility along with additional features, allowing you to remotely manage multiple systems. It offers a choice of a graphical user interface and a scriptable command line interface (CLI). It is a direct-user interface to the FC input/output (FCIO) interface provided by the Oracle StorEdge SFS. The FCIO interface provides an Oracle-common input/output control (IOCTL) interface to the FC transport library (FCTL), which manages the FCA drivers for each FCoE adapter attached to the host system. The OneCommand CNA Manager application also directly interfaces with the Emulex network driver, allowing you to manage the NIC function of Emulex adapters. For more information, refer to the *Emulex OneCommand CNA Manager Application for OneConnect Adapters User Guide*, which is available on the Broadcom website.

3.2   brcmadm Utility

The `brcmadm` utility changes driver parameters through a local interactive or CLI mode. It can also update firmware on non-Oracle branded devices. For more information, refer to the *Emulex Solaris FCA Utilities for OneConnect Adapters User Guide*.

Chapter 4: FCoE Driver Configuration

The module name for the Emulex SFS FCA out-of-box driver is `elxfc`. You can configure the Emulex SFS FCA driver parameters by doing the following:

- Editing the configuration file (`elxfc.conf`), which is described in this section.
- Using the Emulex `brcmadm` utility. For more information, refer to the *Emulex Solaris FCA Utilities for OneConnect Adapters User Guide*.
- Using the OneCommand CNA Manager application. For more information, refer to the *Emulex OneCommand CNA Manager Application for OneConnect Adapters User Guide*.

4.1 Editing the FCoE Configuration File (elxfc.conf)

The configuration file contains all the parameters necessary to initialize the Emulex SFS FCA out-of-box driver.

4.1.1 Changing Driver Parameters

The configurable driver parameters are defined in Table 1, FCoE Configuration File Parameters.

To change driver parameters:

1. Open the configuration file in a text editor.
2. Change the parameters to the desired settings.
3. Save the file.
4. If the driver is already loaded, you must unload and reload the driver to implement your changes. For more information, see Chapter 2, Installing and Uninstalling the Solaris Drivers.

To determine the requirement for the parameter change to take effect, see the Activation column in Table 1, FCoE Configuration File Parameters.

4.1.2 Configuring NPIV Support

This section details the procedure for configuring N_Port ID Virtualization (NPIV) support.

4.1.2.1 Enabling NPIV Support on Solaris 10

To enable NPIV support in the driver:

1. Log in as *root*, or *su* to root.
2. Set `enable-npiv=1` in the configuration file.
3. The FCoE port (fp) driver parameters are updated when the Broadcom `brcmxu` utilities package is installed. Entries from 2 to 254 can be added to the `/kernel/drv/fp.conf` file. For example:
   ```
   name="fp" class="fibre-channel" port=0;
   name="fp" class="fibre-channel" port=1;
   name="fp" class="fibre-channel" port=2;
   name="fp" class="fibre-channel" port=3;
   name="fp" class="fibre-channel" port=4;
   name="fp" class="fibre-channel" port=5;
   name="fp" class="fibre-channel" port=6;
   name="fp" class="fibre-channel" port=7;
   name="fp" class="fibre-channel" port=8;
   ```

```
name="fp" class="fibre-channel" port=9;
```

The first two lines are listed by default because ports 0 and 1 are required. The preceding example shows that ports 2 to 9 have been added to support up to 10 virtual ports. The port number of each entry must increment by 1 in sequential order, without gaps in the number sequence. That is, you cannot have `"name="fp" class="fibre-channel" port=7;"` and then have the next line be `"name="fp" class="fibre-channel" port=10;"`.

4. Reboot the system.

To create, delete, or list the virtual ports after a system reboot, refer to the *Emulex OneCommand CNA Manager Application for OneConnect Adapters User Guide*.

4.1.2.2 Enabling NPIV Support on Solaris 11

To enable NPIV support in the driver:

1. Log in as *root*, or *su* to root.
2. The `enable-npiv` parameter is set to 0 by default. To enable NPIV, set `enable-npiv` to 1 in the configuration file.

Refer to Chapter 6 of the *Solaris Express SAN Configuration and Multipathing Guide*, which is available on the Oracle website, for additional information.

4.1.2.3 NPIV Limitations

The following limitations apply to NPIV:

- Due to the limitation of the Solaris SFS stack, deleting a virtual port causes that virtual port to go offline.
- Fibre Channel over IP (FC-IP) is not supported on virtual ports.
- You cannot delete a virtual port with a mounted file system.

4.1.2.4 NPIV and Operating System Virtualization

Solaris has several operating system virtualization solutions, including Oracle virtual machine (VM) for x86 and Solaris containers. Devices configured to be seen on a Emulex FCoE port (either a physical port or a virtual port) can be used with any of these operating system virtualization solutions. Refer to the latest document on these technologies to learn the best use of resources related to NPIV technology.

4.1.2.5 Using VPorts with Oracle VM Server for Solaris Containers or Oracle VM Server for 86

To use NPIV with Solaris containers or Oracle VM Server for x86 (formerly xVM) user domains:

1. Create virtual ports for the domains or containers to which you want to present dedicated storage.
2. Discover and attach the targets to the virtual ports.
3. Assign the target to the domain or container. The attachment runs through the virtual port that provides the path to the target.

4.2 FCoE Configuration File Parameters

NOTE

- All adapter-specific parameters have an `elxfcX` prefix, where *X* is the driver instance number. For example, setting `elxfc0-log-errors=0xffffffff` enables all the error messages for the 0 instance of the `elxfc` driver.
- All parameters are adapter-specific.

- To override a driver parameter for a single driver-loading session, you can specify it as a parameter to the `modload` command. The following example is for 64-bit platforms:

  ```
  modload /kernel/drv/amd64/elxfc automap=0
  ```

- The OneCommand CNA Manager application reflects the configuration file driver parameters. For more information, refer to the *Emulex OneCommand CNA Manager Application for OneConnect Adapters User Guide*.

- The Activation column in Table 1 shows the requirement for the parameter change to take effect. Activation requirements include adapter reset, dynamic reset (no reset is necessary), link reset, and reboot.

The following table lists the FCoE configuration file parameters.

Table 1 FCoE Configuration File Parameters

Parameter	Description	Activation
ack0	Indicates whether the adapter uses or attempts to use ACK0 for Class 2. ■ 0 = The adapter only uses ACK1 when running Class 2 traffic (default). ■ 1 = The adapter attempts to use ACK0 when running Class 2 traffic to a device. If the device does not support ACK0, the adapter uses ACK1.	Adapter reset
adisc-support	Sets the driver level support for the Fibre Channel (FC) ADISC login I/O recovery method. ■ 0 = No support. Flushes active I/Os for all FCP target devices at link down. ■ 1 = Partial support. Flushes I/Os for non-FCP2 target devices at link down. ■ 2 = Full support. Holds active I/Os for all devices at link down.	Dynamic reset
assign-alpa	If multiple adapter instances on the same host are on the same loop, set this property differently for each adapter. Possible values are 0x00 to 0x0EF. A 0x00 setting (default) means no preference. **NOTE** This property is only applicable if the topology is set to loop.	Link reset
console-errors	Verbose mask for driver error messages to the console. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0x0000000.	Dynamic reset
console-notices	Verbose mask for driver notice messages to the console. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0x0000000.	Dynamic reset
console-warnings	Verbose mask for driver warning messages to the console. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0x0000000.	Dynamic reset
cr-count	Specifies a count of I/O completions after an interrupt response is generated. The possible values are 1 to 255. The default value is 1. **NOTE** This property is disabled if `cr-delay=0`.	Link reset
cr-delay	Specifies a count in milliseconds after which an interrupt response is generated if `cr-count` has not been satisfied. Possible values are 0 to 63. The default value is 0. **NOTE** If `cr-delay=0`, the Coalesce Response parameter (`cr_count`) is disabled.	Link reset
enable-auth	Diffie-Hellman Challenge Handshake Authentication Protocol (DHCHAP) support in the driver. ■ 0 = Disabled (default) ■ 1 = Enabled	Link reset
enable-npiv	NPIV support in the driver. ■ 0 = Disabled; remove all vPorts first ■ 1 = Enabled (requires Service Level Interface (SLI)-3 and later)	Adapter reset

Table 1 FCoE Configuration File Parameters (Continued)

Parameter	Description	Activation
linkup-delay	Sets the driver wait period (seconds) for a link up after adapter initialization. Possible values are 0 to 60. The default value is 10.	Adapter reset
log-errors	Verbose mask for driver error messages to the messages file. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0xFFFFFFFF.	Dynamic reset
log-notices	Verbose mask for driver notice messages to the messages file. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0xFFFFFFFF.	Dynamic reset
log-warnings	Verbose mask for driver warning messages to the messages file. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0xFFFFFFFF.	Dynamic reset
max-xfer-size	Sets the maximum Small Computer Systems Interface (SCSI) transfer size in bytes per I/O. This property determines the scatter gather list buffer size. A pool of buffers is reallocated by the driver during boot. A larger transfer size requires a larger memory allocation. The minimum value is 131072. The maximum value is 1388544. The default value is 339968. **NOTE** For OCe11100-series adapters, the maximum transfer size is 1040384 bytes.	Reboot
network-on	IP networking support in the driver. ■ 0 = Disabled ■ 1 = Enabled (default)	Reboot
num-iocbs	The number of IOCB buffers to allocate. Possible values are 128 to 10240. The default value is 1024.	Adapter reset
num-nodes	The number of remote FC nodes (N_Ports) the driver supports. Possible values are 0 to 4096. The default value is 0, which means no limit.	Adapter reset
pci-max-read	Sets the PCI Extended (PCI-X) maximum memory read byte count. Possible values are 512, 1024, 2048, or 4096. The default value is 2048.	Adapter reset
pm-support	Power management support in the driver. ■ 0 = Disabled (default) ■ 1 = Enabled	Reboot
target-depth	Sets the remote FCP target queue depth. Possible values are 0 to 2048. The default value is 512. A value of 0 means no limit.	Link reset
topology	Sets the topology. Set to point-to-point mode to run as an N_Port. Set to loop mode to run as an NL_Port. ■ 0 = Loop, then point-to-point (default) ■ 2 = Point-to-point only ■ 4 = Loop only ■ 6 = Point-to-point, then loop	Link reset

Table 1 FCoE Configuration File Parameters (Continued)

Parameter	Description	Activation
ub-bufs	Sets the number of unsolicited buffers the driver should allocate. Possible values are 40 to 16320. The default value is 1000.	Reboot
vport	**NOTE** This property is applicable to Solaris 10 only. Virtual port registration table. The enable-npiv parameter must be set to 1. The virtual port registration table can have any number of comma delimited entries. Each entry must be of the form: "PHYS_WWPN:VPORT_WWNN:VPORT_WWPN:VPORT_ID" Where: ■ PHYS_WWPN = World Wide Port Name of adapter's physical port ■ VPORT_WWNN = Desired World Wide Node Name of virtual port ■ VPORT_WWPN = Desired World Wide Port Name of virtual port ■ VPORT_ID = Desired virtual port ID (1 to maximum vPorts) For entries with the same PHYS_WWPN, VPORT_WWNN, and VPORT_WWPN, the VPORT_ID must start at 1, and increment by one sequentially, without gaps in the number sequence. The VPORT_ID=0 is reserved for the physical port. For example: vport= "10000000c9123456:28010000c9123456:20010000c9123456:1", "10000000c9123456:28020000c9123456:20020000c9123456:2", "10000000c9123457:28010000c9123457:20010000c9123457:1", "10000000c9123457:28020000c9123457:20020000c9123457:2", "10000000c9123457:28030000c9123457:20030000c9123457:3"; All entries are automatically created or removed by the OneCommand CNA Manager application.	Link reset
vport-restrict-login	Sets the virtual port behavior when discovering targets in the SAN. ■ 1 = Prevents the virtual port (vPort) from logging into other initiator ports on the SAN. Also rejects logins from other ports in the SAN because it assumes that all ports that send a PLOGI are initiators. This is the default value. ■ 0 = The driver attempts to login to every port that it can access in the SAN and accept logins from all ports. **NOTE** In a SAN that has other initiators, this parameter greatly reduces the use of hardware resources by the driver.	Link reset

Chapter 5: NIC Driver Configuration

In Solaris 10, the driver exports certain parameters that can be configured by editing the Emulex NIC out-of-box driver for Solaris configuration file (`elxnic.conf`). See Section 5.1.2, NIC Configuration File Parameters, and Table 2, NIC Configuration File Parameters.

In Solaris 11, while you can configure the driver using the `elxnic.conf` file, you should use the `dladm` utility (available by default on Solaris 11 systems). For more information on using this Solaris NIC configuration utility, refer to the `dladm(1M)` man page.

> **NOTE**   man pages are available on the Oracle website.

5.1 Using the NIC Configuration File (elxnic.conf)

This section provides information on using and configuring the NIC driver parameters.

5.1.1 Changing Driver Parameters

The configurable driver parameters are described in Table 2, NIC Configuration File Parameters.

To edit the `elxnic.conf` file:

1. Open the file in a text editor.
2. Change the parameters to the desired settings.
   The syntax of single lines in the file:
   ```
   <variable> = <value>;
   ```
   For example:
   ```
   default_mtu = 9000;
   ```
   Comment lines must start with a # character.
3. Save the file.
4. If the driver is already loaded, unload and reload it. Changes to the configuration file require you to unload and reload the driver. For more information, see Chapter 2, Installing and Uninstalling the Solaris Drivers.

5.1.2 NIC Configuration File Parameters

For the Solaris 10 driver, you can configure the `elxnic` driver parameters using either the `elxnic.conf` file (for driver parameters that are common to all the elxnic devices) or the `ndd` utility (for driver parameters that might need to be set with different values for different elxnic devices). For the Solaris 11 driver, use the `dladm` utility or the `elxnic.conf` file.

The following table lists the NIC configuration file parameters.

Table 2 NIC Configuration File Parameters

Parameter	Definition
default_mtu	Sets the default maximum transmission unit (MTU) for the driver. The possible values are 1500 and 9000. The default value is 1500.
flow_control	Sets the Ethernet flow control. The possible values are: ■ 0 - Flow control disabled ■ 1 - Transmit only ■ 2 - Receive only ■ 3 - Both transmit and receive (default) Flow control cannot be disabled on NIC+FCoE adapters.
fm_capability	Sets the driver device fault management capability to one of the values defined for Solaris fault management capability. The fm_capability value is a bitmap of one or more of these values: ■ 0x00000000 = DDI_FM_NOT_CAPABLE; a value of zero indicates that the fm_capability is disabled. ■ 0x00000001 = DDI_FM_EREPORT_CAPABLE ■ 0x00000002 = DDI_FM_ACCCHK_CAPABLE ■ 0x00000004 = DDI_FM_DMA_CHK_CAPABLE The default value is 0x00000007 (DDI_FM_ACCCHK_CAPABLE, DDI_FM_ACCCHK_CAPABLE, and DDI_FM_DMA_CHK_CAPABLE).
log_level	Sets the driver's verbosity for logs in /var/adm/messages. The log_level parameter is comprised of MOD_MASK (upper 16 bits) and SEVERITY (lower 16 bits). The MOD_MASK value is a bitmap of one or more of these values: ■ 0x10000 = MOD_CONFIG ■ 0x20000 = MOD_TX ■ 0x40000 = MOD_RX ■ 0x80000 = MOD_ISR The possible values for SEVERITY: ■ 0x0 = CE_CONT ■ 0x1 = CE_NOTE ■ 0x2 = CE_WARN ■ 0x3 = CE_PANIC ■ 0x4 = CE_IGNORE For details on MOD_MASK and SEVERITY, see Section 7.1, Setting MOD_MASK and SEVERITY.
max_rx_rings	Sets the maximum number of receive queues. Possible values are: ■ 1 to 5: OCe11102. The default value is 5. ■ 1 to 16: OCe11102 in advanced mode on Solaris 10. The default value is 8. ■ 1 to 16: OCe11102 in advanced mode on Solaris 11. The default value is 8. The actual number of transmit and receive queues that are created depends on the number of vectors allocated. The actual number can be checked using the ndd(1m) command on Solaris 10 or the dladm(1M) command on Solaris 11. For more information, run man ndd on Solaris 10 or man dladm on Solaris 11. On non-interrupt resource management (IRM) capable machines with FLEX10 or multi-adapter setup, some of the functions might fail to attach because of a lack of interrupts. See Section 5.1.6, Changing Interrupt Priorities for more information.
max_tx_rings	Sets the maximum number of transmit queues. The possible values are: ■ 1 for OCe11102 in legacy mode. ■ 1 to 8 for OCe11102 in advanced mode on Solaris 10. The default value is 8. ■ 1 to 16 for OCe11102 on Solaris 11. The default value is 8.
rss_key_static	Enables static RSS key generation. The possible values are 0 (disabled) or 1 (enabled). The default value is 0.

Table 2 NIC Configuration File Parameters (Continued)

Parameter	Definition
rx_bcopy_limit	Sets the receive buffer size threshold to use the copy mode. The possible values are any values that are less than or equal to the `default_mtu` value. The default value is 128. **NOTE** On Solaris 11, consider changing this value to fine-tune the receive performance.
rx_frag_size	Sets the size of the pre-allocated receive buffer. A higher value results in better resource utilization. The possible values are 2048, 4096, and 8192. The default value is 2048. **NOTE** On Solaris 10, consider changing the value to 4096 or 8192 if the `default_mtu` value is 9000.
rx_max_bufs	Sets the maximum number of pre-allocated receive buffers. The possible values are 1024 to 8192. The default value is 2048. **NOTE** On Solaris 10, consider increasing the `rx_max_bufs` value if the `rx_drops_no_frags_q[x]` increments continuously in the kstat output.
tx_bcopy_limit	Sets the transmit buffer size threshold to use the copy mode. The possible values are 128, 256, 512, 1024, and 2048. The default value is 512. **NOTE** On Solaris 11, consider reducing this value if a `wqb pool empty` message is frequently observed. Consider increasing the value if a `wqm pool empty` message is frequently observed. For descriptions of these messages, Table 11, Log Messages for the NIC Driver for Solaris 11.
tx_intr_enable	Enables the handling of transmit completions in interrupt mode. The possible values are 0 (disabled) or 1 (enabled). The default value is 0. **NOTE** Enable this parameter if you observe frequent frame loss. Also, consider using this in `rx_max_bufs`.

5.1.3   Configuring the NIC Interface

The NIC interface must be created before you can configure it. You can verify that the driver is loaded on the system and the NIC interface is created with one of the following commands:

For the Solaris 10 driver:

```
dladm show-dev
```

For the Solaris 11 driver:

```
dladm show-phys
```

After you have determined that the NIC interface has been created, you can proceed to configure it.

To configure the NIC interfaces:

1. Plumb the interface.

   For the Solaris 10 driver:

   ```
   #ifconfig elxnic<X> plumb
   ```

   For the Solaris 11 driver:

   ```
   #ifconfig net<X> plumb
   ```

   Where <X> is the interface number.

   To see the created interfaces, run the following command:

   ```
   dladm show-link
   ```

   The `dladm show-link` command lists all the interfaces in the system.

2. Assign a static IP address.

   For the Solaris 10 driver:

   ```
   #ifconfig elxnic<X> <IP_Address> netmask <NetMask> up
   ```

For the Solaris 11 driver:

```
#ifconfig net<X> <IP_Address> netmask <NetMask> up
```

3. Edit the `/etc/hosts` file (a symlink to `/etc/inet/hosts`), and add the IP address and host name that you want to assign to the given NIC interface. Refer to the `hosts(4)` man page for more information.

4. Edit the `/etc/inet/ipnodes` file, and add an entry for the IP address and host name for the given interface. The `/etc/inet/ipnodes` file is primarily for IPv6 only, but this step is necessary for the IP address change to take effect.

5. Edit the `/etc/netmasks` file, and add an entry with the IP address and desired subnet mask for the given interface. Refer to the `netmasks(4)` man page for more information.

6. Restart the network service. Run the following command, or reboot the system.

```
svcadm restart network/physical
```

5.1.4   Removing the NIC Interface using elxnic

To remove the NIC interface:

1. Remove all the `elxnic` entries from `/etc/hosts`.

2. Remove all the entries from `/etc/inet/ipnodes` that are related to `elxnic` interfaces.

3. Remove all the `elxnic` netmask entries from `/etc/netmasks`.

4. Unplumb the interface, using the following command:

   For the Solaris 10 driver:

   ```
   ifconfig elxnic<X> down unplumb
   ```

   For the Solaris 11 driver:

   ```
   ifconfig net<X> down unplumb
   ```

   where <X> is the interface number.

Alternatively, you can use the `sys-unconfig(1M)` utility for Solaris 10 or `sysconfig(1M)` utility for Solaris 11 to remove a device configuration. These utilities can delete the configurations of IP address, netmask, host name, NFS mounts, LDAP, and so on, on the host. The `sys-unconfig(1M)` or `sysconfig(1M)` utility reboots the system and clears the existing IP configuration, so you must enter all the information again, even for the existing NICs already configured in the system. The `sys-unconfig(1M)` or `sysconfig(1M)` utility must be executed from a console. On reboot, you are presented with a set of user-interface-based data entry forms that facilitate the required change in configuration.

5.1.5   Multi-ring Mode and the Effect of ddi_msix_alloc_limit

5.1.5.1   Solaris 10

The `elxnic` driver is enabled to work in the multi-ring mode by default. To check the number of active transmit and receive (tx/rx) rings, type

```
#ndd -get /dev/elxnic<X> rx_rings
#ndd -get /dev/elxnic<X> tx_rings
```

The variable <X> is the interface number of the `elxnic` driver.

The following table shows how the value of `ddi_msix_alloc_limit` affects receive traffic distribution across central processing unit (CPU) cores for Solaris 10.

Table 3 Effect of `ddi_msix_alloc_limit` **across CPU Cores for Solaris 10**

NIC	Mode	APIC	Value of ddi_msix_alloc_limit in /etc/system	CPU Cores Participating in RX Processing	Limiting Factor
OCe11102	Legacy	xAPIC	Default	2	Solaris and advanced programmable interrupt controller (APIC) in platform
OCe11102	Legacy	xAPIC	4	4	Firmware and driver
OCe11102	Legacy	x2APIC	Default	4	Firmware and driver
OCe11102	Advanced	xAPIC	Default	2	Solaris and APIC in platform
OCe11102	Advanced	xAPIC	8	8	Firmware and driver
OCe11102	Advanced	x2APIC	Default	8	Firmware and driver

5.1.5.2   Solaris 11

Solaris 11 is multi-ring enabled by default. It creates eight transmit rings and eight receive rings on OCe11102 devices in advanced mode. To check the number of rx and tx rings, use the `dladm show-phys -H` command. In Table 4, the number 8 in column 5 is applicable only if the number of rx rings is increased to the maximum (16).

> **NOTE**   On non-IRM systems with multiple adapters, the attach for few NIC functions might fail because of lack of interrupt vectors allowed at the default level. To solve this issue, use Table 4.

The following table shows how the value of `ddi_msix_alloc_limit` affects receive traffic distribution across CPU cores for Solaris 11.

Table 4 Effect of `ddi_msix_alloc_limit` **across CPU Cores for Solaris 11**

NIC	Mode	APIC	Value of ddi_msix_alloc_limit in /etc/system	CPU Cores Participating in RX Processing	Limiting Factor
OCe11102	Legacy	xAPIC	Default	2	Solaris and APIC in platform
OCe11102	Legacy	xAPIC	4	4	Firmware and driver
OCe11102	Legacy	x2APIC	Default	4	Firmware and driver
OCe11102	Advanced	xAPIC	Default	2	Solaris and APIC in platform
OCe11102	Advanced	xAPIC	8	8	Firmware and driver
OCe11102	Advanced	x2APIC	Default	8	Firmware and driver

5.1.6   Changing Interrupt Priorities

The Solaris operating system divides the available interrupts among multiple priority levels; each priority level has a maximum of 31 vectors. By default, each function has a maximum of two message signaled interrupts – extended (MSI-X) vectors. For example, with three OCe11102 adapters in FLEX10 mode, there are 24 functions, and the driver needs at least two vectors per function for a total of 48 vectors. If the MSI-X allocation fails, the driver reverts to the INTx allocation, which results in suboptimal performance. Normally, the network drivers allocate vectors at level 6, but the requirement of 48 vectors is beyond the limit of 31, which causes issues. A solution is to assign level 6 to 12 functions and level 5 to the remaining functions so that all functions can get two MSI-X vectors for optimum performance. It is not uncommon to assign level 5 or 6 on Solaris under these conditions.

The following example shows how the interrupt-priority for a Peripheral Component Interconnect (PCI) function can be configured to a level 5:

1. Type

```
#  grep elxnic /etc/path_to_inst
"/pci@0,0/pci8086,340d@6/pci10df,e743@0" 0 "elxnic"
"/pci@0,0/pci8086,340d@6/pci10df,e743@0,1" 1 "elxnic"
"/pci@0,0/pci8086,3410@9/pci10df,e742@0" 2 "elxnic"
"/pci@0,0/pci8086,3410@9/pci10df,e742@0,1" 3 "elxnic"
#  grep elxnic /etc/driver_aliases
elxnic "pciex19a2,710"
```

> **NOTE** In the previous example, `/pci@n,n/pcinnnn,nnnd@n` represents the parent of the PCI function, the unit number follows the last @ symbol, and pciexnnan,nnn indicates the device name.

2. Add the following entry to `/kernel/drv/elxnic.conf` file:

```
name= "pciex19a2,710"  parent = "/pci@0,0/pci8086,3410@9" unit-address =
"0" interrupt-priorities = 5;
```

> **NOTE** The path and unit number might vary on your system.

3. Reboot the system.

Interrupt priorities can be examined using the mdb command. For example:

```
# echo "::interrupts "| mdb -k | grep oce
```

The third column shows the interrupt priority level of the vector in the following output:

```
IRQ  Vect IPL Bus   Trg Type   CPU   Share APIC/INT#  ISR(s)
64   0x42 5         Edg MSI-X  3     1     -          oce_isr
65   0x43 5         Edg MSI-X  0     1     -          oce_isr
```

5.2 Using the dladm Utility for Solaris 11

5.2.1 Configuring the NIC Interface Using dladm

To configure the interface on Solaris 11, use the `dladm` utility to configure the NIC interface and to perform a runtime update of the following driver parameter.

For example, to change the MTU in Solaris 11, unplumb the interface and run:

```
$> dladm set-linkprop -p mtu=9000 <interface>
```

where `<interface>` is net<0,1,2...>.

To see the plumbed interfaces, run:

```
$> ifconfig -a.
```

5.2.1.1 Tunable Parameters

Tunable parameters can be changed during runtime. On Solaris 10, use the `ndd` command to change the value of tunable parameters. On Solaris 11, use the `dladm` command to change these values. The parameters on Solaris 11 are named differently. They include a prefix of _ to each of the parameter names. Through the `dladm` utility, the following tunable parameters are provided by the driver along with their usage:

log_level

```
$> dladm set-linkprop -p _log_level=<value> <interface>
```

> **NOTE**   See Section 7.1, Setting MOD_MASK and SEVERITY, for `log_level` values.

rx_bcopy_limit

```
$> dladm set-linkprop -p _rx_bcopy_limit=<value in bytes> <interface>
```

tx_bcopy_limit

```
$> dladm set-linkprop -p _tx_bcopy_limit=<value in bytes> <interface>
```

tx_ring_size

```
$> dladm set-linkprop -p _tx_ring_size=<values between 256 and 2048> <interface>
```

or

```
$> dladm set-linkprop -p _tx_ring_size=<values less than tx_ring_size> <interface>
```

5.2.1.2   Private (or Unlisted) Parameters

Set private (or unlisted) parameters during driver attach through the `dladm` utility and also the `elxnic.conf` file.

To see the value of private driver parameters using the `dladm` utility, type:

```
$> dladm show-linkprop -p <property name> <interface>
```

The driver provides the following private parameters:

- `fw_version`
- `log_level`
- `rx_bcopy_limit`
- `rx_ring_size`
- `rx_rings`
- `rx_rings_per_group`
- `tx_bcopy_limit`
- `tx_reclaim_threshold`
- `tx_ring_size`
- `tx_rings_fw_version` (read only)

5.2.2   Creating a Virtual NIC Using dladm

To create a vNIC, use the `dladm` utility with the `create-vnic` option. For example:

```
$ dladm create-vnic -l  net0  vnic1
```

After a vNIC is created, it can be assigned to a zone using the `zonecfg` utility. For example:

```
$ zonecfg -z zone1
zonecfg:zone1: No such zone configured
zonecfg:zone1> create
zonecfg:zone1> set zonepath=/export/zone1
zonecfg:zone1> create
zonecfg:zone1> add net
zonecfg:zone1> set physical=vnic1
zonecfg:zone1> set address=192.168.1.100
zonecfg:zone1> verify
zonecfg:zone1> commit
```

Up to 63 VLANs can be used with each universal multichannel (UMC) virtual channel. For information on configuring UMC, refer to the *Emulex Universal Multichannel Reference Manual*, which is available on the Broadcom website.

Considerations

■ UMC can be configured using the OneCommand CNA Manager application CLI. For more information, refer to the *Emulex OneCommand CNA Manager Command Line Interface User Guide*, which is available on the Broadcom website.

■ Using UMC in a hypervisor environment is not advised if the UMC interface is going to be part of the hypervisor virtual switch. This configuration might cause performance issues.

■ You cannot run Link Aggregation Control Protocol (LACP) if UMC is enabled.

Chapter 6: FCoE Driver Console and Log Messages

This chapter describes the situations, console messages, and log messages you might see from the FCoE driver.

6.1　Setting Notices, Warnings, and Error Logging Levels

The following table lists the types of notices, warnings, and error logging levels you might set using the appropriate FCoE driver parameters.

Table 5　Setting Types of Console and Log Messages for the FCoE Driver

`elxfc` **Parameter**	**Description**	`lpfc` **Parameter**
`console-notices`	Verbose mask for driver error messages to the console. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0x0000000.	`log-only`: If set to 0, log messages are logged to the system log file and also printed on the console. Default = Disabled
`console-warnings`	Verbose mask for driver notice messages to the console. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0x0000000.	
`console-errors`	Verbose mask for driver warning messages to the console. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0x0000000.	
`log-notices`	Verbose mask for driver error messages to the messages file. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0xFFFFFFFF.	`log-verbose`: If set to nonzero, verbose messages are generated. Default = Disabled
`log-warnings`	Verbose mask for driver notice messages to the messages file. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0xFFFFFFFF.	
`log-errors`	Verbose mask for driver warning messages to the messages file. Possible values are 0x0000000 to 0xFFFFFFFF. The default value is 0xFFFFFFFF.	

6.2　Log Messages

6.2.1　Types of Log Messages Logged in the System File

Log messages are logged to the `/var/adm/messages` system file. The following table lists the types of log messages that can be logged to the system file.

Table 6　Log Message Types for the FCoE Driver

Verbose Bit	Log Message Verbose Mask	Verbose Description	Pages
0x00000001	LOG_MISC	Miscellaneous events	28 to 29
0x00000002	LOG_DRIVER	Driver attach and detach events	29 to 30
0x00000004	LOG_INIT	Adapter initialization events	31 to 31
0x00000008	LOG_MEM	Memory management events	32 to 33
0x00000010	LOG_SLI	SLI events	33 to 36

Table 6 Log Message Types for the FCoE Driver (Continued)

Verbose Bit	Log Message Verbose Mask	Verbose Description	Pages
0x00000020	LOG_MBOX	Mailbox events	36 to 36
0x00000040	LOG_NODE	Node events	37 to 38
0x00000080	LOG_LINK	Link events	38 to 39
0x00000100	LOG_ELS	Extended Link Service (ELS) events	39 to 40
0x00000200	LOG_PKT	General I/O packet events	40 to 41
0x00000400	LOG_FCP	FCP traffic events	42
0x00001000	LOG_IP	IP traffic events	42 to 43
0x00002000	LOG_SFS	Solaris SFS events	43 to 44
0x00004000	LOG_IOCTL	IOCTL events	44 to 45
0x00008000	LOG_FIRMWARE	Firmware download events	45 to 46
0x00010000	LOG_CT	Common transport events	47 to 47
0x00020000	LOG_FCSP	FCSP events	48 to 49
0x00040000	LOG_FCF	FCF events	49 to 49
0x007C0000	LOG_RESERVED	Reserved	—
0x01000000	LOG_FCSP_DETAIL	Detailed FCSP events	48
0x02000000	LOG_NODE_DETAIL	Detailed node events	37, 37
0x04000000	LOG_IOCTL_DETAIL	Detailed IOCTL events	44, 45
0x08000000	LOG_IP_DETAIL	Detailed IP events	42
0x10000000	LOG_FIRMWARE_DETAIL	Detailed firmware events	45
0x20000000	LOG_SFS_DETAIL	Detailed Solaris SFS events	43
0x40000000	LOG_MBOX_DETAIL	Detailed mailbox events	36
0x80000000	LOG_SLI_DETAIL	Detailed adapter SLI events	36
0XFFFFFFFF	LOG_ALL_MSG	All logging on all events	-

6.2.2 Log Message Severity Levels

The following table lists the severity type of the FCoE driver log message in the order of severity.

Table 7 Severity Levels of FCoE Driver Log Messages

Severity Level	Description
DEBUG (Informational)	Message provides engineering debugging information.
NOTICE (Informational)	Message provides general purpose information.
WARNING	Message provides a general purpose warning.
ERROR	Message indicates that a driver error has occurred.
PANIC (Severe)	Message indicates that the driver has forced a system panic to occur.

6.2.3 Log Message Example

The following is an example of a log message on the system console:

```
[5.0336]elxfc0: NOTICE: 720: Link up. (8Gb, fabric)
```

The following is an example of the same message in the system message log (`/var/adm/messages`) file:

```
Jan 19 14:45:36 sunv240 elxfc: [ID 349649 kern.info] [5.0336]elxfc0:
NOTICE: 720: Link up. (8Gb, fabric)
```

In the preceding system log message:

- `Jan 19 14:45:36` – The date and time when the error or event occurred.
- `sunv240` – The name of the host machine.
- `elxfc:` – The module where the message originates. In this case, `elxfc` means that it is from Emulex SFS FCA out-of-box driver.
- `[ID 349649 kern.info]` – A Solaris-specific message ID and kernel message level. This changes from one driver message to another.
- `[5.0336]` – The driver message context tag. This might change from one driver version to another.
- `elxfc0:` – The module or instance where the message originates. In this case, `elxfc0` means that it is from Emulex SFS FCA out-of-box driver, instance of 0. This changes from one driver instance to another.
- `NOTICE:` – Identifies the driver message severity level. This might change from one driver version to another. For other severity information, see Table 7, Severity Levels of FCoE Driver Log Messages.
- `720:` – Identifies the driver message identification number. This number does change from one driver version to another.
- `Link up.` – Identifies the actual error or event message. This message does not change from one driver version to another.
- `(8 Gb/s, fabric)` – Identifies additional information specific to the error or event message. This information is normally intended for technical support. This message might change from one driver version to another.

6.3 Log Messages for the FCoE Driver

6.3.1 Miscellaneous Events (Message IDs: 0001 to 0099)

```
0001 DEBUG:
      VERBOSE_MASK: LOG_MISC (0x00000001)
      DESCRIPTION: This is a general purpose informational message.
      SEVERITY LEVEL: Debug
      MESSAGE: None
      ACTION: No action needed, informational.
0002 NOTICE:
      VERBOSE_MASK: LOG_MISC (0x00000001)
      DESCRIPTION: This is a general purpose informational message.
      SEVERITY LEVEL: Notice
      MESSAGE: None
      ACTION: No action needed, informational.
0003 WARNING:
      VERBOSE_MASK: LOG_MISC (0x00000001)
      DESCRIPTION: This is a general purpose warning message.
      SEVERITY LEVEL: Warning
      MESSAGE: None
      ACTION: No action needed, informational.
0004 ERROR:
      VERBOSE_MASK: LOG_MISC (0x00000001)
      DESCRIPTION: This is a general purpose error message.
      SEVERITY LEVEL: Error
```

```
           MESSAGE: None
           ACTION: No action needed, informational.
      0005 PANIC:
           VERBOSE_MASK: LOG_MISC (0x00000001)
           DESCRIPTION: This is a general purpose panic message.
           SEVERITY LEVEL: Panic (Severe)
           MESSAGE: None
           ACTION: Contact your customer service representative.
      0010 DEBUG: Event.
           VERBOSE_MASK: LOG_MISC (0x00000001)
           DESCRIPTION: This is debug information about a driver event.
           SEVERITY LEVEL: Debug
           MESSAGE: Event.
           ACTION: No action needed, informational.
      0011 DEBUG: Event queued.
           VERBOSE_MASK: LOG_MISC (0x00000001)
           DESCRIPTION: This indicates that a driver event is being queued.
           SEVERITY LEVEL: Debug
           MESSAGE: Event queued.
           ACTION: No action needed, informational.
      0012 DEBUG: Event dequeued.
           VERBOSE_MASK: LOG_MISC (0x00000001)
           DESCRIPTION: This indicates that a driver event is being dequeued.
           SEVERITY LEVEL: Debug
           MESSAGE: Event dequeued.
           ACTION: No action needed, informational.
```

6.3.2   Driver Events (Message IDs: 0100 to 0199)

```
      0100 NOTICE: Driver attach.
           VERBOSE_MASK: LOG_DRIVER (0x00000002)
           DESCRIPTION: This indicates that the driver is performing an attach
           operation.
           SEVERITY LEVEL: Notice
           MESSAGE: Driver attach.
           ACTION: No action needed, informational.
      0101 ERROR: Driver attach failed.
           VERBOSE_MASK: LOG_DRIVER (0x00000002)
           DESCRIPTION: This indicates that the driver was unable to attach due to some
           issue.
           SEVERITY LEVEL: Error
           MESSAGE: Driver attach failed.
           ACTION: Check your hardware and software configuration. If the problem
           persists, report this error to your customer service representative.
      0102 DEBUG: Driver attach.
           VERBOSE_MASK: LOG_DRIVER (0x00000002)
           DESCRIPTION: This indicates that the driver is performing a attach
           operation.
           SEVERITY LEVEL: Debug
           MESSAGE: Driver attach.
           ACTION: No action needed, informational.
      0110 NOTICE: Driver detach.
           VERBOSE_MASK: LOG_DRIVER (0x00000002)
```

```
          DESCRIPTION: This indicates that the driver is performing a detach
          operation.
          SEVERITY LEVEL: Notice
          MESSAGE: Driver detach.
          ACTION: No action needed, informational.
     0111 ERROR: Driver detach failed.
          VERBOSE_MASK: LOG_DRIVER (0x00000002)
          DESCRIPTION: This indicates that the driver was unable to detach due to some
          issue.
          SEVERITY LEVEL: Error
          MESSAGE: Driver detach failed.
          ACTION: Check your hardware and software configuration. If the problem
          persists, report this error to your customer service representative.
     0112 DEBUG: Driver detach.
          VERBOSE_MASK: LOG_DRIVER (0x00000002)
          DESCRIPTION: This indicates that the driver is performing a detach
          operation.
          SEVERITY LEVEL: Debug
          MESSAGE: Driver detach.
          ACTION: No action needed, informational.
     0120 DEBUG: Driver suspend.
          VERBOSE_MASK: LOG_DRIVER (0x00000002)
          DESCRIPTION: This indicates that the driver is performing a suspend
          operation.
          SEVERITY LEVEL: Debug
          MESSAGE: Driver suspend.
          ACTION: No action needed, informational.
     0121 ERROR: Driver suspend failed.
          VERBOSE_MASK: LOG_DRIVER (0x00000002)
          DESCRIPTION: This indicates that the driver was unable to suspend due to
          some issue.
          SEVERITY LEVEL: Error
          MESSAGE: Driver suspend failed.
          ACTION: Check your hardware and software configuration. If the problem
          persists, report this error to your customer service representative.
     0130 DEBUG: Driver resume.
          VERBOSE_MASK: LOG_DRIVER (0x00000002)
          DESCRIPTION: This indicates that the driver is performing a resume
          operation.
          SEVERITY LEVEL: Debug
          MESSAGE: Driver resume.
          ACTION: No action needed, informational.
     0131 ERROR: Driver resume failed.
          VERBOSE_MASK: LOG_DRIVER (0x00000002)
          DESCRIPTION: This indicates that the driver was unable to resume due to some
          issue.
          SEVERITY LEVEL: Error
          MESSAGE: Driver resume failed.
          ACTION: Check your hardware and software configuration. If the problem
          persists, report this error to your customer service representative.
```

6.3.3 Adapter Initialization Events (Message IDs: 0200 to 0299)

```
0200 NOTICE: Adapter initialization.
     VERBOSE_MASK: LOG_INIT (0x00000004)
     DESCRIPTION: This indicates that the adapter is initializing.
     SEVERITY LEVEL: Notice
     MESSAGE: Adapter initialization.
     ACTION: No action needed, informational.
0201 ERROR: Adapter initialization failed.
     VERBOSE_MASK: LOG_INIT (0x00000004)
     DESCRIPTION: This indicates that an attempt to initialize the adapter has
     failed.
     SEVERITY LEVEL: Error
     MESSAGE: Adapter initialization failed.
     ACTION: Check your hardware configuration. If the problem persists, report
     this error to your customer service representative.
0202 DEBUG: Adapter initialization.
     VERBOSE_MASK: LOG_INIT (0x00000004)
     DESCRIPTION: This indicates that the adapter is initializing.
     SEVERITY LEVEL: Debug
     MESSAGE: Adapter initialization.
     ACTION: No action needed, informational.
0210 DEBUG: Adapter transition.
     VERBOSE_MASK: LOG_INIT (0x00000004)
     DESCRIPTION: This indicates that the adapter is changing states.
     SEVERITY LEVEL: Debug
     MESSAGE: Adapter transition.
     ACTION: No action needed, informational.
0220 DEBUG: Adapter online.
     VERBOSE_MASK: LOG_INIT (0x00000004)
     DESCRIPTION: This indicates that the adapter is online and ready to
     communicate.
     SEVERITY LEVEL: Debug
     MESSAGE: Adapter online.
     ACTION: No action needed, informational.
0230 DEBUG: Adapter offline.
     VERBOSE_MASK: LOG_INIT (0x00000004)
     DESCRIPTION: This indicates that the adapter is offline and unable to
     communicate.
     SEVERITY LEVEL: Debug
     MESSAGE: Adapter offline.
     ACTION: No action needed, informational.
0231 WARNING: Adapter shutdown.
     VERBOSE_MASK: LOG_INIT (0x00000004)
     DESCRIPTION: This indicates that the adapter has been shutdown and will
     require a reboot to reinitialize.
     SEVERITY LEVEL: Warning
     MESSAGE: Adapter shutdown.
     ACTION: Contact your customer service representative.
0240 ERROR: Adapter reset failed.
     VERBOSE_MASK: LOG_INIT (0x00000004)
     DESCRIPTION: This indicates that an attempt to reset the adapter has failed.
     SEVERITY LEVEL: Error
```

```
MESSAGE: Adapter reset failed.
ACTION: Check your hardware configuration. If the problem persists, report
this error to your customer service representative.
```

6.3.4   Memory Management Events (Message IDs: 0300 to 0399)

```
0300 DEBUG: Memory alloc.
     VERBOSE_MASK: LOG_MEM (0x00000008)
     DESCRIPTION: This indicates that the driver allocated system memory.
     SEVERITY LEVEL: Debug
     MESSAGE: Memory alloc.
     ACTION: No action needed, informational.
0301 ERROR: Memory alloc failed.
     VERBOSE_MASK: LOG_MEM (0x00000008)
     DESCRIPTION: This indicates that the driver was unable to allocate system
     memory. The system is low on memory resources.
     SEVERITY LEVEL: Error
     MESSAGE: Memory alloc failed.
     ACTION: No action needed, informational. However, if the problem persists,
     report this error to your system administrator.
0310 ERROR: Memory pool error.
     VERBOSE_MASK: LOG_MEM (0x00000008)
     DESCRIPTION: This indicates that a problem has occurred with the memory
     buffer pool management.
     SEVERITY LEVEL: Error
     MESSAGE: Memory pool error.
     ACTION: No action needed, informational. However, if the problem persists,
     report this error to your customer service representative.
0311 DEBUG: Memory pool alloc failed.
     VERBOSE_MASK: LOG_MEM (0x00000008)
     DESCRIPTION: This indicates that the driver was unable to allocate memory
     from one of its own memory pools.
     SEVERITY LEVEL: Debug
     MESSAGE: Memory pool alloc failed.
     ACTION: If the problem occurs frequently you may be able to configure more
     resources for that pool. If this does not solve the problem, report these
     errors to customer service.
0312 DEBUG: Memory pool detail.
     VERBOSE_MASK: LOG_MEM (0x00000008)
     DESCRIPTION: This provides detailed information about memory bufferpool
     management.
     SEVERITY LEVEL: Debug
     MESSAGE: Memory pool detail.
     ACTION: No action needed, informational.
0320 NOTICE: No unsolicited buffer available.
     VERBOSE_MASK: LOG_MEM (0x00000008)
     DESCRIPTION: This indicates that the driver's unsolicited buffer pool is
     exhausted. The I/O will be dropped and most likely retried by the remote
     device.
     SEVERITY LEVEL: Notice
     MESSAGE: No unsolicited buffer available.
```

```
            ACTION: If the problem occurs frequently you may be able to configure more
            resources for that pool. If this does not solve the problem, report these
            errors to customer service.
0330 ERROR: Invalid access handle.
            VERBOSE_MASK: LOG_MEM (0x00000008)
            DESCRIPTION: This indicates that the driver had an invalid access handle
            assigned by the system.
            SEVERITY LEVEL: Error
            MESSAGE: Invalid access handle.
            ACTION: If the problem occurs frequently, report these errors to customer
            service.
0331 ERROR: Invalid DMA handle.
            VERBOSE_MASK: LOG_MEM (0x00000008)
            DESCRIPTION: This indicates that the driver had an invalid dma handle
            assigned by the system.
            SEVERITY LEVEL: Error
            MESSAGE: Invalid DMA handle.
            ACTION: If the problem occurs frequently, report these errors to customer
            service.
```

6.3.5   Service Level Interface Events (Message IDs: 0400 to 0499)

```
0400 DEBUG: Vital Product Data.
            VERBOSE_MASK: LOG_SLI (0x00000010)
            DESCRIPTION: This provides vendor specific information about the adapter.
            SEVERITY LEVEL: Debug
            MESSAGE: Vital Product Data.
            ACTION: No action needed, informational.
0410 DEBUG: Link atten.
            VERBOSE_MASK: LOG_SLI (0x00000010)
            DESCRIPTION: This indicates that the adapter has triggered a link attention
            interrupt.
            SEVERITY LEVEL: Debug
            MESSAGE: Link atten.
            ACTION: No action needed, informational.
0411 DEBUG: State change.
            VERBOSE_MASK: LOG_SLI (0x00000010)
            DESCRIPTION: This indicates that the adapter has changed state.
            SEVERITY LEVEL: Debug
            MESSAGE: State change.
            ACTION: No action needed, informational.
0412 DEBUG: Link Up atten.
            VERBOSE_MASK: LOG_SLI (0x00000010)
            DESCRIPTION: This indicates that the adapter has triggered a link up
            attention interrupt.
            SEVERITY LEVEL: Debug
            MESSAGE: Link Up atten.
            ACTION: No action needed, informational.
0413 DEBUG: Link Down atten.
            VERBOSE_MASK: LOG_SLI (0x00000010)
            DESCRIPTION: This indicates that the adapter has triggered a link down
            attention interrupt.
            SEVERITY LEVEL: Debug
```

```
        MESSAGE: Link Down atten.
        ACTION: No action needed, informational.
0420 ERROR: Adapter hardware error.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that an interrupt has occurred and the status
        register indicates a nonrecoverable hardware error.
        SEVERITY LEVEL: Error
        MESSAGE: Adapter hardware error.
        ACTION: This error usually indicates a hardware problem with the adapter.
        Try running adapter diagnostics. Report these errors to customer service.
0421 NOTICE: Adapter temperature.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that the adapter has provided general
        information about the adapter's temperature.
        SEVERITY LEVEL: Notice
        MESSAGE: Adapter temperature.
        ACTION: No action needed, informational.
0422 WARNING: Adapter temperature.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that the adapter's temperature is too hot.
        SEVERITY LEVEL: Warning
        MESSAGE: Adapter temperature.
        ACTION: Check hardware ventilation. Reduce adapter usage. Shutdown host
        system.
0423 NOTICE: Adapter notice.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that the adapter has provided general
        information about the adapter's condition.
        SEVERITY LEVEL: Notice
        MESSAGE: Adapter notice.
        ACTION: No action needed, informational.
0424 WARNING: Adapter warning.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that an interrupt has occurred indicating a
        recoverable adapter error.
        SEVERITY LEVEL: Warning
        MESSAGE: Adapter warning.
        ACTION: This error usually indicates a hardware or firmware problem with the
        adapter. Check and/or update firmware levels. Report these errors to
        customer service.
0425 ERROR: Adapter error.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that a recoverable adapter error has occurred.
        SEVERITY LEVEL: Error
        MESSAGE: Adapter error.
        ACTION: This error usually indicates a hardware or firmware problem with the
        adapter. Check and/or update firmware levels. Report these errors to
        customer service.
0426 NOTICE: Adapter Async Status.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that the adapter has provided general
        information about the adapter's async status.
        SEVERITY LEVEL: Notice
        MESSAGE: Adapter Async Status.
```

```
        ACTION: No action needed, informational.
0430 DEBUG: Ring event.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that an SLI ring event has occurred.
        SEVERITY LEVEL: Debug
        MESSAGE: Ring event.
        ACTION: No action needed, informational.
0431 DEBUG: Ring error.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that an SLI ring error is being reported by the
        adapter.
        SEVERITY LEVEL: Debug
        MESSAGE: Ring error.
        ACTION: No action needed, informational.
0432 DEBUG: Ring reset.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that an SLI ring is being reset.
        SEVERITY LEVEL: Debug
        MESSAGE: Ring reset.
        ACTION: No action needed, informational.
0440 DEBUG: Adapter msg.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that a message was sent to the driver from the
        adapter.
        SEVERITY LEVEL: Debug
        MESSAGE: Adapter msg.
        ACTION: No action needed, informational.
0450 ERROR: IOCB invalid.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that an IOCB was received from the adapter with
        an illegal value. This error could indicate a driver or firmware problem.
        SEVERITY LEVEL: Error
        MESSAGE: IOCB invalid.
        ACTION: No action needed, informational. However, if the problem persists,
        report this error to your customer service representative.
0451 DEBUG: IOCB queue full.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that the IOCB queue is full. This will occur
        during normal operation.
        SEVERITY LEVEL: Debug
        MESSAGE: IOCB queue full.
        ACTION: No action needed, informational.
0452 DEBUG: IOCB event.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that an IOCB local error event is being reported
        by the adapter.
        SEVERITY LEVEL: Debug
        MESSAGE: IOCB event.
        ACTION: No action needed, informational.
0453 DEBUG: IOCB stale.
        VERBOSE_MASK: LOG_SLI (0x00000010)
        DESCRIPTION: This indicates that an IOCB completed after its associated
        packet completed.
        SEVERITY LEVEL: Debug
```

```
                MESSAGE: IOCB stale.
                ACTION: No action needed, informational.
        0460 DEBUG: SLI.
                VERBOSE_MASK: LOG_SLI_DETAIL (0x80000000)
                DESCRIPTION: This provides detailed information about an SLI event.
                SEVERITY LEVEL: Debug
                MESSAGE: SLI detail.
                ACTION: No action needed, informational.
        0461 ERROR: SLI ERROR.
                VERBOSE_MASK: LOG_SLI (0x00000010)
                DESCRIPTION: This error provides information about an SLI event.
                SEVERITY LEVEL: Error
                MESSAGE: SLI ERROR.
                ACTION: No action needed, informational.
        0462 DEBUG: SLI DEBUG.
                VERBOSE_MASK: LOG_SLI (0x00000010)
                DESCRIPTION: This provides debug information about an SLI event.
                SEVERITY LEVEL: Debug
                MESSAGE: SLI DEBUG.
                ACTION: No action needed, informational.
```

6.3.6   Mailbox Events (Message IDs: 0500 to 0599)

```
        0500 DEBUG: Mailbox event.
                VERBOSE_MASK: LOG_MBOX (0x00000020)
                DESCRIPTION: This indicates that a mailbox event has occurred.
                SEVERITY LEVEL: Debug
                MESSAGE: Mailbox event.
                ACTION: No action needed, informational.
        0501 DEBUG: Mailbox detail.
                VERBOSE_MASK: LOG_MBOX_DETAIL (0x40000000)
                DESCRIPTION: This provides detailed information about a mailbox event.
                SEVERITY LEVEL: Debug
                MESSAGE: Mailbox detail.
                ACTION: No action needed, informational.
        0510 DEBUG: Stray mailbox interrupt.
                VERBOSE_MASK: LOG_MBOX (0x00000020)
                DESCRIPTION: This indicates that a mailbox command completion interrupt was
                received and the mailbox is not valid. This error could indicate a driver or
                firmware problem.
                SEVERITY LEVEL: Debug
                MESSAGE: Stray mailbox interrupt.
                ACTION: No action needed, informational. However, if the problem persists,
                report this error to your customer service representative.
        0520 DEBUG: Mailbox error.
                VERBOSE_MASK: LOG_MBOX (0x00000020)
                DESCRIPTION: This indicates that an unsupported or illegal mailbox command
                was completed. This error could indicate a driver or firmware problem.
                SEVERITY LEVEL: Debug
                MESSAGE: Mailbox error.
                ACTION: No action needed, informational. However, if the problem persists,
                report this error to your customer service representative.
        0530 ERROR: Mailbox timeout.
```

VERBOSE_MASK: LOG_MBOX (0x00000020)
DESCRIPTION: The firmware did not response a mailbox command. This error
could indicate a hardware or firmware problem.
SEVERITY LEVEL: Error
MESSAGE: Mailbox timeout.
ACTION: No action needed, informational. However, if the problem persists,
report this error to your customer service representative.

6.3.7   Node Events (Message IDs: 0600 to 0699)

0600 DEBUG: Node create.
      VERBOSE_MASK: LOG_NODE (0x00000040)
      DESCRIPTION: This indicates that a node has been created for a remote
      device.
      SEVERITY LEVEL: Debug
      MESSAGE: Node create.
      ACTION: No action needed, informational.
0601 DEBUG: Node opened.
      VERBOSE_MASK: LOG_NODE_DETAIL (0x02000000)
      DESCRIPTION: This indicates that a node has been opened for I/O transport.
      SEVERITY LEVEL: Debug
      MESSAGE: Node opened.
      ACTION: No action needed, informational.
0602 NOTICE: Node create failed.
      VERBOSE_MASK: LOG_NODE (0x00000040)
      DESCRIPTION: This indicates that a node create request for a remote device
      has failed.
      SEVERITY LEVEL: Notice
      MESSAGE: Node create failed.
      ACTION: No action needed, informational.
0603 DEBUG: Node updated.
      VERBOSE_MASK: LOG_NODE (0x00000040)
      DESCRIPTION: This indicates that a node has been updated for a remote
      device.
      SEVERITY LEVEL: Debug
      MESSAGE: Node updated.
      ACTION: No action needed, informational.
0610 DEBUG: Node destroy.
      VERBOSE_MASK: LOG_NODE (0x00000040)
      DESCRIPTION: This indicates that a node has been destroyed for a remote
      device.
      SEVERITY LEVEL: Debug
      MESSAGE: Node destroy.
      ACTION: No action needed, informational.
0611 DEBUG: Node closed.
      VERBOSE_MASK: LOG_NODE_DETAIL (0x02000000)
      DESCRIPTION: This indicates that a node has been temporarily closed for I/O
      transport.
      SEVERITY LEVEL: Debug
      MESSAGE: Node closed.
      ACTION: No action needed, informational.
0612 NOTICE: Node missing.
      VERBOSE_MASK: LOG_NODE (0x00000040)

DESCRIPTION: This indicates that a FCP2 device node has been found missing.
SEVERITY LEVEL: Notice
MESSAGE: Node missing.
ACTION: No action needed, informational.

0620 DEBUG: Node not found.
VERBOSE_MASK: LOG_NODE (0x00000040)
DESCRIPTION: This indicates that there was an attempt to send an I/O pkt to an unknown device node. The driver maintains a node table entry for every device it needs to communicate with on the FC network.
SEVERITY LEVEL: Debug
MESSAGE: Node not found.
ACTION: No action needed, informational. However, if the problem persists, report this error to your customer service representative.

0621 DEBUG: Node timeout.
VERBOSE_MASK: LOG_NODE (0x00000040)
DESCRIPTION: This indicates that the node timer expired; the node is ready to be opened, or it has been offline too long and needs to be flushed.
SEVERITY LEVEL: Debug
MESSAGE: Node timeout.
ACTION: No action needed, informational.

6.3.8 Link Events (Message IDs: 0700 to 0799)

0700 DEBUG: Link event.
VERBOSE_MASK: LOG_SLI (0x00000010) or LOG_LINK (0x00000080)
DESCRIPTION: This indicates that a link event has occurred.
SEVERITY LEVEL: Debug
MESSAGE: Link event.
ACTION: No action needed, informational.

0710 NOTICE: Link down.
VERBOSE_MASK: LOG_LINK (0x00000080)
DESCRIPTION: This indicates that the Fibre Channel link is down to the adapter.
SEVERITY LEVEL: Notice
MESSAGE: Link down.
ACTION: Check your network connections. If the problem persists, report this error to your system administrator.

0720 NOTICE: Link up.
VERBOSE_MASK: LOG_LINK (0x00000080)
DESCRIPTION: This indicates that the Fibre Channel link is up.
SEVERITY LEVEL: Notice
MESSAGE: Link up.
ACTION: No action needed, informational.

0721 NOTICE: NPIV Link up.
VERBOSE_MASK: LOG_LINK (0x00000080)
DESCRIPTION: This indicates that the Fibre Channel link is up for all virtual ports.
SEVERITY LEVEL: Notice
MESSAGE: NPIV Link up.
ACTION: No action needed, informational.

0730 NOTICE: Link reset.
VERBOSE_MASK: LOG_LINK (0x00000080) or LOG_SFS (0x00002000)

DESCRIPTION: This indicates that an issue has forced the Fibre Channel link
to be reset.
SEVERITY LEVEL: Notice
MESSAGE: Link reset.
ACTION: No action needed, informational.

0731 ERROR: Link reset failed.
VERBOSE_MASK: LOG_LINK (0x00000080) or LOG_SFS (0x00002000)
DESCRIPTION: This indicates that an attempt to reset the Fibre Channel link
has failed.
SEVERITY LEVEL: Error
MESSAGE: Link reset failed.
ACTION: No action needed, informational. However, if the problem persists,
report this error to your customer service representative.

6.3.9   ELS Events (Message IDs: 0800 to 0899)

0800 DEBUG: ELS sent.
VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an ELS command is being sent.
SEVERITY LEVEL: Debug
MESSAGE: ELS sent.
ACTION: No action needed, informational.

0801 DEBUG: ELS comp.
VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an ELS command completed normally.
SEVERITY LEVEL: Debug
MESSAGE: ELS comp.
ACTION: No action needed, informational.

0810 ERROR: Stray ELS completion.
VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an ELS command completion was received
without issuing a corresponding ELS command. This error could indicate a
driver or firmware problem.
SEVERITY LEVEL: Error
MESSAGE: Stray ELS completion.
ACTION: No action needed, informational. However, if the problem persists,
report this error to your customer service representative.

0811 DEBUG: Abnormal ELS completion.
VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an ELS command completed with a status
error in the IOCB. It could mean the Fibre Channel device on the network is
not responding or the Fibre Channel device is not an FCP target. The driver
will automatically
SEVERITY LEVEL: Debug
MESSAGE: Abnormal ELS completion.
ACTION: retry this ELS command if needed. If the command is a PLOGI or PRLI,
and the destination PortID is not an FCP Target, no action is needed.
Otherwise, check physical connections to Fibre Channel network and the state
the remote PortID is in.

0820 DEBUG: ELS rcvd.
VERBOSE_MASK: LOG_ELS (0x00000100)
DESCRIPTION: This indicates that an unsolicited ELS command was received.
SEVERITY LEVEL: Debug

```
            MESSAGE: ELS rcvd.
            ACTION: No action needed, informational.
0821 DEBUG: Unsolicited ELS dropped.
            VERBOSE_MASK: LOG_ELS (0x00000100)
            DESCRIPTION: This indicates that an unsolicited ELS command was received and
            then dropped for some reason.
            SEVERITY LEVEL: Debug
            MESSAGE: Unsolicited ELS dropped.
            ACTION: No action needed, informational.
0822 DEBUG: ELS reply.
            VERBOSE_MASK: LOG_ELS (0x00000100)
            DESCRIPTION: This indicates that a reply is being sent for an unsolicited
            ELS command.
            SEVERITY LEVEL: Debug
            MESSAGE: ELS reply.
            ACTION: No action needed, informational.
0830 ERROR: Invalid ELS command found.
            VERBOSE_MASK: LOG_ELS (0x00000100)
            DESCRIPTION: This indicates that an ELS command was found with an invalid
            command code.
            SEVERITY LEVEL: Error
            MESSAGE: Invalid ELS command found.
            ACTION: No action needed, informational. However, if the problem persists,
            report this error to your customer service representative.
```

6.3.10   General I/O Packet Events (Message IDs: 0900 to 0999)

```
0900 NOTICE: Packet abort.
            VERBOSE_MASK: LOG_PKT (0x00000200)
            DESCRIPTION: This indicates that an I/O packet is being aborted.
            SEVERITY LEVEL: Notice
            MESSAGE: Packet abort.
            ACTION: No action needed, informational.
0901 WARNING: Packet abort failed.
            VERBOSE_MASK: LOG_PKT (0x00000200)
            DESCRIPTION: This indicates that an attempt to abort an I/O packet has
            failed.
            SEVERITY LEVEL: Warning
            MESSAGE: Packet abort failed.
            ACTION: No action needed, informational. However, if the problem persists,
            report this error to your customer service representative.
0910 DEBUG: Packet timeout.
            VERBOSE_MASK: LOG_PKT (0x00000200)
            DESCRIPTION: This indicates that an I/O packet has timed out and is being
            aborted.
            SEVERITY LEVEL: Debug
            MESSAGE: Packet timeout.
            ACTION: No action needed, informational.
0911 DEBUG: CHANNEL watchdog.
            VERBOSE_MASK: LOG_PKT (0x00000200)
            DESCRIPTION: This indicates that I/O(s) are getting stale waiting on a I/O
            channel tx queue.
            SEVERITY LEVEL: Debug
```

```
         MESSAGE: CHANNEL watchdog.
         ACTION: No action needed, informational.
0912 DEBUG: TXQ watchdog.
         VERBOSE_MASK: LOG_PKT (0x00000200)
         DESCRIPTION: This indicates that an I/O was found missing from the transmit
         queue.
         SEVERITY LEVEL: Debug
         MESSAGE: TXQ watchdog.
         ACTION: No action needed, informational.
0920 DEBUG: Packet flush.
         VERBOSE_MASK: LOG_PKT (0x00000200)
         DESCRIPTION: This indicates that an I/O packet is being flushed.
         SEVERITY LEVEL: Debug
         MESSAGE: Packet flush.
         ACTION: No action needed, informational.
0921 DEBUG: Packet flushed.
         VERBOSE_MASK: LOG_PKT (0x00000200)
         DESCRIPTION: This indicates that an I/O packet has been flushed.
         SEVERITY LEVEL: Debug
         MESSAGE: Packet flushed.
         ACTION: No action needed, informational.
0922 NOTICE: Packet flush timeout.
         VERBOSE_MASK: LOG_PKT (0x00000200)
         DESCRIPTION: This indicates that an I/O packet flush request has timed out
         with some I/O packets's still not completed. The driver will attempt to
         recover by itself.
         SEVERITY LEVEL: Notice
         MESSAGE: Packet flush timeout.
         ACTION: No action needed, informational. However, if the problem persists,
         report this error to your customer service representative.
0930 NOTICE: Packet transport failed.
         VERBOSE_MASK: LOG_PKT (0x00000200)
         DESCRIPTION: This indicates that an attempt to send an I/O packet failed.
         The I/O packet will be retried by the upper layer.
         SEVERITY LEVEL: Notice
         MESSAGE: Packet transport failed.
         ACTION: No action needed, informational.
0931 ERROR: Packet transport error.
         VERBOSE_MASK: LOG_PKT (0x00000200)
         DESCRIPTION: This indicates that an error occurred while attempting to send
         an I/O packet. The I/O packet will likely be failed back to the user
         application.
         SEVERITY LEVEL: Error
         MESSAGE: Packet transport error.
         ACTION: No action needed, informational. However, if the problem persists,
         report this error to your customer service representative.
0932 DEBUG: Packet transport.
         VERBOSE_MASK: LOG_PKT (0x00000200)
         DESCRIPTION: This provides additional information about a packet being
         sent.
         SEVERITY LEVEL: Debug
         MESSAGE: Packet transport.
         ACTION: No action needed, informational.
0940 DEBUG: Packet completion error.
```

```
VERBOSE_MASK: LOG_PKT (0x00000200)
DESCRIPTION: This indicates that an I/O packet was completed with an error
status. This can occur during normal operation.
SEVERITY LEVEL: Debug
MESSAGE: Packet completion error.
ACTION: No action needed, informational. However, if the problem persists,
report this error to your customer service representative.
```

6.3.11   FCP Traffic Events (Message IDs: 1000 to 1099)

```
1000 DEBUG: Stray FCP completion.
     VERBOSE_MASK: LOG_FCP (0x00000400)
     DESCRIPTION: This indicates that an FCP command completion was received
     without issuing a corresponding FCP command. This error could indicate a
     driver or firmware problem.
     SEVERITY LEVEL: Debug
     MESSAGE: Stray FCP completion.
     ACTION: No action needed, informational. However, if the problem persists,
     report this error to your customer service representative.
1001 DEBUG: FCP completion error.
     VERBOSE_MASK: LOG_FCP (0x00000400)
     DESCRIPTION: This indicates that an FCP command completed with an error
     status. These errors can occur during normal operation.
     SEVERITY LEVEL: Debug
     MESSAGE: FCP completion error.
     ACTION: No action needed, informational.
```

6.3.12   IP Traffic Events (Message IDs: 1200 to 1299)

```
1200 DEBUG: IP detail.
     VERBOSE_MASK: LOG_IP_DETAIL (0x08000000)
     DESCRIPTION: This provides detailed information about the driver's IP
     interface.
     SEVERITY LEVEL: Debug
     MESSAGE: IP detail.
     ACTION: No action needed, informational.
1210 ERROR: Stray IP completion.
     VERBOSE_MASK: LOG_IP (0x00001000)
     DESCRIPTION: This indicates that an IP sequence completion was received
     without issuing a corresponding IP sequence. This error could indicate a
     driver or firmware problem.
     SEVERITY LEVEL: Error
     MESSAGE: Stray IP completion.
     ACTION: No action needed, informational. However, if the problem persists,
     report this error to your customer service representative.
1211 DEBUG: Abnormal IP completion.
     VERBOSE_MASK: LOG_IP (0x00001000)
     DESCRIPTION: This indicates that an IP sequence completed with a status
     error in the IOCB. It could mean the Fibre Channel device on the network is
     not responding.
     SEVERITY LEVEL: Debug
     MESSAGE: Abnormal IP completion.
```

```
        ACTION: No action needed, informational. However, if the problem persists,
        report this error to your system administrator.
1220 DEBUG: Unsolicited IP dropped.
        VERBOSE_MASK: LOG_IP (0x00001000)
        DESCRIPTION: This indicates that an unsolicited IP sequence was received,
        but was dropped for some reason.
        SEVERITY LEVEL: Debug
        MESSAGE: Unsolicited IP dropped.
        ACTION: No action needed, informational.
1221 DEBUG: IP recvd.
        VERBOSE_MASK: LOG_IP (0x00001000)
        DESCRIPTION: This indicates that an unsolicited IP sequence was received.
        SEVERITY LEVEL: Debug
        MESSAGE: IP recvd.
        ACTION: No action needed, informational.
1230 ERROR: Invalid IP sequence found.
        VERBOSE_MASK: LOG_IP (0x00001000)
        DESCRIPTION: This indicates that an IP sequence was found with an invalid
        code.
        SEVERITY LEVEL: Error
        MESSAGE: Invalid IP sequence found.
        ACTION: No action needed, informational. However, if the problem persists,
        report this error to your customer service representative.
```

6.3.13   Solaris SFS Events (Message IDs: 1300 to 1399)

```
1300 DEBUG: SFS.
        VERBOSE_MASK: LOG_SFS (0x00002000)
        DESCRIPTION: This provides general information about the driver's SFS
        interface.
        SEVERITY LEVEL: Debug
        MESSAGE: SFS.
        ACTION: No action needed, informational.
1301 DEBUG: SFS detail.
        VERBOSE_MASK: LOG_SFS_DETAIL (0x20000000)
        DESCRIPTION: This provides detailed information about the driver's SFS
        interface.
        SEVERITY LEVEL: Debug
        MESSAGE: SFS detail.
        ACTION: No action needed, informational.
1310 WARNING: Diagnostic error.
        VERBOSE_MASK: LOG_SFS (0x00002000)
        DESCRIPTION: This indicates that a diagnostic request did not complete
        because of some issue.
        SEVERITY LEVEL: Warning
        MESSAGE: Diagnostic error.
        ACTION: No action needed, informational. However, if the problem persists,
        report this error to your customer service representative.
1311 DEBUG: ECHO diagnostic completed.
        VERBOSE_MASK: LOG_SFS (0x00002000)
        DESCRIPTION: This indicates that an ECHO diagnostic has completed.
        SEVERITY LEVEL: Debug
        MESSAGE: ECHO diagnostic completed.
```

```
                ACTION: No action needed, informational.
    1312 WARNING: ECHO diagnostic failed.
                VERBOSE_MASK: LOG_SFS (0x00002000)
                DESCRIPTION: This indicates that an ECHO diagnostic has failed to return a
                positive result. This could indicate a connectivity problem with your FC
                network.
                SEVERITY LEVEL: Warning
                MESSAGE: ECHO diagnostic failed.
                ACTION: Check your network connections. If the problem persists, report this
                error to your system administrator.
    1313 DEBUG: BIU diagnostic completed.
                VERBOSE_MASK: LOG_SFS (0x00002000)
                DESCRIPTION: This indicates that a BIU diagnostic has completed.
                SEVERITY LEVEL: Debug
                MESSAGE: BIU diagnostic completed.
                ACTION: No action needed, informational.
    1314 ERROR: BIU diagnostic failed.
                VERBOSE_MASK: LOG_SFS (0x00002000)
                DESCRIPTION: This indicates that a BIU diagnostic has failed to return a
                positive result. This usually caused by an adapter hardware problem.
                SEVERITY LEVEL: Error
                MESSAGE: BIU diagnostic failed.
                ACTION: Contact your customer service representative.
    1315 DEBUG: POST diagnostic completed.
                VERBOSE_MASK: LOG_SFS (0x00002000)
                DESCRIPTION: This indicates that a POST diagnostic has completed.
                SEVERITY LEVEL: Debug
                MESSAGE: POST diagnostic completed.
                ACTION: No action needed, informational.
    1316 ERROR: POST diagnostic failed.
                VERBOSE_MASK: LOG_SFS (0x00002000)
                DESCRIPTION: This indicates that a POST diagnostic has failed to return a
                positive result. This is usually caused by an adapter hardware problem.
                SEVERITY LEVEL: Error
                MESSAGE: POST diagnostic failed.
                ACTION: Contact your customer service representative.
```

6.3.14   IOCTL Events (Message IDs: 1400 to 1499)

```
    1400 DEBUG: IOCTL.
                VERBOSE_MASK: LOG_IOCTL (0x00004000)
                DESCRIPTION: This provides general information about the driver's IOCTL
                interface.
                SEVERITY LEVEL: Debug
                MESSAGE: IOCTL.
                ACTION: No action needed, informational.
    1401 DEBUG: IOCTL detail.
                VERBOSE_MASK: LOG_IOCTL_DETAIL (0x04000000)
                DESCRIPTION: This provides detailed information about the driver's IOCTL
                interface.
                SEVERITY LEVEL: Debug
                MESSAGE: IOCTL detail.
                ACTION: No action needed, informational.
```

```
1410 DEBUG: DFC
        VERBOSE_MASK: LOG_IOCTL (0x00004000)
        DESCRIPTION: This provides general information about the driver's DFC
        interface.
        SEVERITY LEVEL: Debug
        MESSAGE: DFC
        ACTION: No action needed, informational.
1411 DEBUG: DFC detail.
        VERBOSE_MASK: LOG_IOCTL_DETAIL (0x04000000)
        DESCRIPTION: This provides detailed information about the driver's DFC
        interface.
        SEVERITY LEVEL: Debug
        MESSAGE: DFC detail.
        ACTION: No action needed, informational.
1420 DEBUG: DFC Error.
        VERBOSE_MASK: LOG_IOCTL (0x00004000)
        DESCRIPTION: This indicates that an error was found while processing a DFC
        request.
        SEVERITY LEVEL: Debug
        MESSAGE: DFC Error.
        ACTION: No action needed, informational.
```

6.3.15   Firmware Download Events (Message IDs: 1500 to 1599)

```
1500 DEBUG: Firmware image.
        VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
        DESCRIPTION: This provides general information about the firmware image.
        SEVERITY LEVEL: Debug
        MESSAGE: Firmware image.
        ACTION: No action needed, informational.
1501 DEBUG: Firmware detail.
        VERBOSE_MASK: LOG_FIRMWARE_DETAIL (0x10000000)
        DESCRIPTION: This provides detailed information about the firmware image.
        SEVERITY LEVEL: Debug
        MESSAGE: Firmware detail.
        ACTION: No action needed, informational.
1502 NOTICE: Firmware Library
        VERBOSE_MASK: LOG_DRIVER (0x00000002)
        DESCRIPTION: This shows the versions of firmware contained in the driver's
        library.
        SEVERITY LEVEL: Notice
        MESSAGE: Firmware Library
        ACTION: No action needed, informational.
1510 ERROR: Bad firmware image.
        VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
        DESCRIPTION: This indicates that a bad firmware image was provided to the
        download function.
        SEVERITY LEVEL: Error
        MESSAGE: Bad firmware image.
        ACTION: Obtain the proper image file. If the problem persists, report this
        error to your customer service representative.
1511 ERROR: Firmware image not compatible.
        VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
```

      DESCRIPTION: This indicates that the firmware image provided was not
      compatible with the existing hardware.
      SEVERITY LEVEL: Error
      MESSAGE: Firmware image not compatible.
      ACTION: Obtain the proper image file. If the problem persists, report this
      error to your customer service representative.

1520 NOTICE: Firmware download.
      VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
      DESCRIPTION: This indicates that an attempt to download a firmware image has
      occurred.
      SEVERITY LEVEL: Notice
      MESSAGE: Firmware download.
      ACTION: No action needed, informational.

1521 NOTICE: Firmware download complete.
      VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
      DESCRIPTION: This indicates that an attempt to download a firmware image was
      successful.
      SEVERITY LEVEL: Notice
      MESSAGE: Firmware download complete.
      ACTION: No action needed, informational.

1522 ERROR: Firmware download failed.
      VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
      DESCRIPTION: This indicates that an attempt to download a firmware image was
      failed.
      SEVERITY LEVEL: Error
      MESSAGE: Firmware download failed.
      ACTION: Check your hardware configuration. If the problem persists, report
      this error to your customer service representative.

1523 WARNING: Firmware updated.
      VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
      DESCRIPTION: This indicates that new firmware has been updated on the
      adapter.
      SEVERITY LEVEL: Warning
      MESSAGE: Firmware updated.
      ACTION: A reboot or adapter power cycle will be required to activate the new
      firmware.

1530 DEBUG: Firmware dump.
      VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
      DESCRIPTION: This indicates that a firmware core dump has occurred.
      SEVERITY LEVEL: Debug
      MESSAGE: Firmware dump.
      ACTION: Check your hardware configuration. If the problem persists, report
      this error to your customer service representative.

1540 WARNING: Firmware update required.
      VERBOSE_MASK: LOG_FIRMWARE (0x00008000)
      DESCRIPTION: This indicates that a firmware update is required on the
      adapter.
      SEVERITY LEVEL: Warning
      MESSAGE: Firmware update required.
      ACTION: The user must perform a manual adapter reset or link reset once the
      host environment is stable to trigger an automatic firmware download. Do not
      power cycle or reboot the system during the download operation.

6.3.16   Common Transport Events (Message IDs: 1600 to 1699)

```
1600 DEBUG: CT sent.
     VERBOSE_MASK: LOG_CT (0x00010000)
     DESCRIPTION: This indicates that a CT command is being sent.
     SEVERITY LEVEL: Debug
     MESSAGE: CT sent.
     ACTION: No action needed, informational.
1601 DEBUG: CT comp.
     VERBOSE_MASK: LOG_CT (0x00010000)
     DESCRIPTION: This indicates that a CT command completed normally.
     SEVERITY LEVEL: Debug
     MESSAGE: CT comp.
     ACTION: No action needed, informational.
1610 ERROR: Stray CT completion.
     VERBOSE_MASK: LOG_CT (0x00010000)
     DESCRIPTION: This indicates that a CT command completion was received
     without issuing a corresponding CT command. This error could indicate a
     driver or firmware problem.
     SEVERITY LEVEL: Error
     MESSAGE: Stray CT completion.
     ACTION: No action needed, informational. However, if the problem persists,
     report this error to your customer service representative.
1611 DEBUG: Abnormal CT completion.
     VERBOSE_MASK: LOG_CT (0x00010000)
     DESCRIPTION: This indicates that a CT command completed with a status error
     in the IOCB. It could mean the Fibre Channel device on the network is not
     responding. The driver will automatically retry this CT command if needed.
     SEVERITY LEVEL: Debug
     MESSAGE: Abnormal CT completion.
     ACTION: Check physical connections to Fibre Channel network and the state
     the remote PortID is in.
1620 DEBUG: CT rcvd.
     VERBOSE_MASK: LOG_CT (0x00010000)
     DESCRIPTION: This indicates that an unsolicited CT command was received.
     SEVERITY LEVEL: Debug
     MESSAGE: CT rcvd.
     ACTION: No action needed, informational.
1621 DEBUG: Unsolicited CT dropped.
     VERBOSE_MASK: LOG_CT (0x00010000)
     DESCRIPTION: This indicates that an unsolicited CT command was received and
     then dropped for some reason.
     SEVERITY LEVEL: Debug
     MESSAGE: Unsolicited CT dropped.
     ACTION: No action needed, informational.
1622 DEBUG: CT reply.
     VERBOSE_MASK: LOG_CT (0x00010000)
     DESCRIPTION: This indicates that a reply is being sent for an unsolicited CT
     command.
     SEVERITY LEVEL: Debug
     MESSAGE: CT reply.
     ACTION: No action needed, informational.
1630 ERROR: Invalid CT command found.
```

```
VERBOSE_MASK: LOG_CT (0x00010000)
DESCRIPTION: This indicates that a CT command was found with an invalid
command code.
SEVERITY LEVEL: Error
MESSAGE: Invalid CT command found.
ACTION: No action needed, informational. However, if the problem persists,
report this error to your customer service representative.
```

6.3.17 FCSP Events (Message IDs: 1700 to 1799)

```
1700 DEBUG: FCSP
     VERBOSE_MASK: LOG_FCSP (0x00020000)
     DESCRIPTION: This provides general information about the driver's FCSP
     interface.
     SEVERITY LEVEL: Debug
     MESSAGE: FCSP
     ACTION: No action needed, informational.
1701 DEBUG: FCSP detail.
     VERBOSE_MASK: LOG_FCSP_DETAIL (0x01000000)
     DESCRIPTION: This provides detailed information about the driver's FCSP
     interface.
     SEVERITY LEVEL: Debug
     MESSAGE: FCSP detail.
     ACTION: No action needed, informational.
1702 DEBUG: FCSP error.
     VERBOSE_MASK: LOG_FCSP (0x00020000)
     DESCRIPTION: This indicates that an error was found while processing a DFC
     request.
     SEVERITY LEVEL: Debug
     MESSAGE: FCSP error.
     ACTION: No action needed, informational.
1705 DEBUG: FCSP state.
     VERBOSE_MASK: LOG_FCSP (0x00020000)
     DESCRIPTION: This indicates that an authentication state is changing.
     SEVERITY LEVEL: Debug
     MESSAGE: FCSP state.
     ACTION: No action needed, informational.
1706 DEBUG: FCSP event.
     VERBOSE_MASK: LOG_FCSP (0x00020000)
     DESCRIPTION: This indicates that an authentication event has occurred.
     SEVERITY LEVEL: Debug
     MESSAGE: FCSP event
     ACTION: No action needed, informational.
1707 DEBUG: FCSP status.
     VERBOSE_MASK: LOG_FCSP (0x00020000)
     DESCRIPTION: This indicates that an authentication status is being updated.
     SEVERITY LEVEL: Debug
     MESSAGE: FCSP status.
     ACTION: No action needed, informational.
1710 DEBUG: FCSP start.
     VERBOSE_MASK: LOG_FCSP (0x00020000)
     DESCRIPTION: This indicates that authentication is being started to a
     specific node.
```

```
          SEVERITY LEVEL: Debug
          MESSAGE: FCSP start.
          ACTION: No action needed, informational.
     1720 DEBUG: FCSP comp.
          VERBOSE_MASK: LOG_FCSP (0x00020000)
          DESCRIPTION: This indicates that authentication is being stopped or
          completed to a specific node.
          SEVERITY LEVEL: Debug
          MESSAGE: FCSP comp.
          ACTION: No action needed, informational.
```

6.3.18   FCF Events (Message IDs: 1800 to 1899)

```
     1800 DEBUG: FCF
          VERBOSE_MASK:
          DESCRIPTION: This provides general information about the driver's FCF
          interface.
          SEVERITY LEVEL: Debug
          MESSAGE: FCF
          ACTION: No action needed, informational.
     1801 DEBUG: FCF detail.
          VERBOSE_MASK:
          DESCRIPTION: This provides detailed information about the driver's FCF
          interface.
          SEVERITY LEVEL: Debug
          MESSAGE: FCF detail.
          ACTION: No action needed, informational.
     1810 DEBUG: FCF error.
          VERBOSE_MASK:
          DESCRIPTION: This indicates that an error was found while processing an FCF
          request.
          SEVERITY LEVEL: Debug
          MESSAGE: FCF error.
          ACTION: No action needed, informational.
     1820 DEBUG: FCF state.
          VERBOSE_MASK:
          DESCRIPTION: This indicates that an FCF object state is changing.
          SEVERITY LEVEL: Debug
          MESSAGE: FCF state.
          ACTION: No action needed, informational.
     1830 DEBUG: FCF event.
          VERBOSE_MASK:
          DESCRIPTION: This indicates that an FCF event has occurred.
          SEVERITY LEVEL: Debug
          MESSAGE: FCF event.
          ACTION: No action needed, informational.
```

Chapter 7: NIC Driver Log Messages

This chapter describes the log messages that you might see from the NIC driver.

NOTE   To reset the adapter, reboot the system.

7.1 Setting MOD_MASK and SEVERITY

Log messages are generated based on the settings of the Module Mask (MOD_MASK) and the Severity Level (SEVERITY) as indicated in the log_level parameter. See the log_level parameter in Table 2, NIC Configuration File Parameters.

The following table details the values and meanings of MOD_MASK in the log_level parameter.

Table 8 MOD_MASK **Information for NIC Driver Log Messages**

Name	MOD_MASK[a]	Meaning
MOD_CONFIG	0x10000	Messages in the device configuration path are logged.
MOD_TX	0x20000	Messages in the transmit data path are logged.
MOD_RX	0x40000	Messages in the receive data path are logged.
MOD_ISR	0x80000	Messages in the interrupt path are logged.

a.   MOD_MASK is a bitmap of one or more of these values.

The following table details the values and meanings of SEVERITY in the log_level parameter.

Table 9 Severity Level Information for NIC Driver Log Messages

Name	SEVERITY **Value**	Meaning
CE_CONT	0x0	Continuation
CE_NOTE	0x1	Information
CE_WARN	0x2	Warning
CE_PANIC	0x3	Causes the operating system to panic
CE_IGNORE	0x4	No action

7.2 Log Messages for the NIC Driver

7.2.1 For Solaris 10

The following table details the log messages provided by the NIC driver for Solaris 10 operating systems.

Table 10 Log Messages for the NIC Driver for Solaris 10

Log Message	Action	MOD_MASK	SEVERITY
"Insufficient Vectors"	Reload the driver after changing the interrupt priorities.	MOD_CONFIG	CE_WARN
<code> POST ERROR!!	Hardware error – Download the same firmware image to the adapter and reboot; replace the adapter if necessary.	MOD_CONFIG	CE_WARN
<code> soft_reset bit asserted[1]. Reset failed	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit	MOD_CONFIG	CE_WARN
Alloc intr failed: ret <ret code>, navail <code>, nallocd <code>	Reload the driver.	MOD_CONFIG	CE_WARN
Block Enable Interrupts Failed <code>	Reload the driver.	MOD_CONFIG	CE_WARN
Could not allocate fn_config_dbuf <code>	Reload the driver.	MOD_CONFIG	CE_WARN
Could not reset firmware, ERR bit set pd_ctrl: <code>	Reset the adapter.	MOD_CONFIG	CE_WARN
Could not reset firmware, ERR bit set pd_ctrl:<reg>	Firmware reset failed. Reset the system.	MOD_CONFIG	CE_WARN
CQ create failed – status: <code>	Hardware error – Download the same firmware image to the adapter and reboot; replace the adapter if necessary.	MOD_CONFIG	CE_WARN
ddi_dma_addr_bind_handle() failed rc: <code>	Increase the memory.	MOD_CONFIG	CE_WARN
ddi_intr_get_pri() failed	Reboot the system.	MOD_CONFIG	CE_WARN
Device in error state	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
device is in Suspended state	Resume the device.	MOD_CONFIG	CE_WARN
Device Unknown	The driver does not support a particular adapter. Contact Broadcom Technical Support for further information.	MOD_CONFIG	CE_WARN
EQ create failed – status: <code>	Hardware error – Download the same firmware image to the adapter and reboot; replace the adapter if necessary.	MOD_CONFIG	CE_WARN
EQ ring alloc failed	Increase the memory.	MOD_CONFIG	CE_WARN

Table 10 Log Messages for the NIC Driver for Solaris 10 (Continued)

Log Message	Action	MOD_MASK	SEVERITY
`Error recovery failed, device is in error state`	The hardware or driver cannot recover from a serious hardware error. Reset the system.	MOD_CONFIG	CE_WARN
`Error recovery failed/timed out`	Reset the adapter.	MOD_CONFIG	CE_WARN
`Error recovery timed out`	The hardware or driver cannot recover from a serious hardware error. Reset the system.	MOD_CONFIG	CE_WARN
`Extended MQ create failed - status: <code>`	Hardware error – Download the same firmware image to the adapter and reboot; replace the adapter if necessary.	MOD_CONFIG	CE_WARN
`Failed  to Enable Interrupt <code>`	Reload the driver.	MOD_CONFIG	CE_WARN
`Failed to add interrupt handlers`	Reload the driver.	MOD_CONFIG	CE_WARN
`Failed to allocate bmbx: size <code>`	Reload the driver.	MOD_CONFIG	CE_WARN
`Failed to allocate Queue memory`	Increase the memory.	MOD_CONFIG	CE_WARN
`Failed to Configure the Adapter`	Reboot the system.	MOD_CONFIG	CE_WARN
`Failed to create WQ ring`	Increase the memory.	MOD_CONFIG	CE_WARN
`Failed to del q - status: <status code>`	Reboot the system.	MOD_CONFIG	CE_WARN
`Failed to init stats`	Reboot the system.	MOD_CONFIG	CE_WARN
`Failed to init tx/rx rings`	Reboot the system.	MOD_CONFIG	CE_WARN
`Failed to read MAC address - status: <status code>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Failed to register mac`	Reboot the system.	MOD_CONFIG	CE_WARN
`Failed to setup interrupts`	Reload the driver after changing the interrupt priorities.	MOD_CONFIG	CE_WARN
`FUNCTION RESET FAILED rc: <code>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Hardware UE Detected: <code>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`HW Initialization Failed`	Hardware error – Download the same firmware image to the adapter and reboot; replace the adapter if necessary.	MOD_CONFIG	CE_WARN
`HW POST FAILED`	Reset the adapter.	MOD_CONFIG	CE_WARN
`Insufficient Vectors`	Reload the driver after changing the interrupt priorities.	MOD_CONFIG	CE_WARN
`Invalid q length. Must be [256, 512, 1024, 2048]: <code>`	Correct the tx q length in the `elxnic.conf` file.	MOD_CONFIG	CE_WARN

Table 10 Log Messages for the NIC Driver for Solaris 10 (Continued)

Log Message	Action	MOD_MASK	SEVERITY
Invalid rx_frag_size: <code> valid values are 2K, 4K, 8K, 16K only.	Correct the rx_frag_size in the elxnic.conf file.	MOD_CONFIG	CE_WARN
IRM action <code> count <code> ddi_intr_free() failed rc: <ret_code>	Reboot the system.	MOD_CONFIG	CE_WARN
IRM: Alloc intr failed: ret <ret_code>, count <code>, nallocd <code>	Reboot the system.	MOD_CONFIG	CE_WARN
lancer physdev reset failed	Reset the adapter.	MOD_CONFIG	CE_WARN
lancer_physdev_reset INP bit polling timedout: <timeout> pd_ctrl: <code>	Reset the adapter.	MOD_CONFIG	CE_WARN
Legacy MQ create failed – status: <code>	Hardware error – Download the same firmware image to the adapter and reboot; replace the adapter if necessary.	MOD_CONFIG	CE_WARN
Legacy MQ ring alloc failed	Increase the memory.	MOD_CONFIG	CE_WARN
Mailbox initialization Failed with rc: <code>	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
mod_install failed	Uninstall the driver; try installing the driver after a reboot.	MOD_CONFIG	CE_WARN
MQ EXT ring alloc failed	Increase the memory.	MOD_CONFIG	CE_WARN
No response from FW for Mailbox command. Mailbox stalled till recovery	Reset the adapter.	MOD_CONFIG	CE_WARN
oce_detach: Rx buffers pending with stack	Reboot the system.	MOD_CONFIG	CE_WARN
OCE_INITIATE_DUMP Dump Image not present	The diagnostic dump image is not present. Retry the dump initiation.	MOD_CONFIG	CE_WARN
OCE_INITIATE_DUMP failed	The diagnostic dump initiation by an application failed. Retry the dump initiation.	MOD_CONFIG	CE_WARN
OCE_INITIATE_DUMP failed due to recovery timeout	Device recovery failed after diagnostic dump initiation. Reboot the system.	MOD_CONFIG	CE_WARN
oce_m_unicast: Factory mac addition failed, status: <status_code>	Reboot the system.	MOD_CONFIG	CE_WARN
oce_m_unicast: mac delete failed, status: <code>	Reboot the system.	MOD_CONFIG	CE_WARN
oce_m_unicast: ucast mac addition failed, staus:<status_code> Resetting to Factory MAC.	Reboot the system.	MOD_CONFIG	CE_WARN
oce_rx:no frags?	Malfunctioning hardware – Check the SFP and the adapter; replace them if necessary.	MOD_RX	CE_WARN
PCI Initialization Failed	Reset or replace the adapter, or move the adapter to different slot.	MOD_CONFIG	CE_WARN

Table 10 Log Messages for the NIC Driver for Solaris 10 (Continued)

Log Message	Action	MOD_MASK	SEVERITY
`pci_config_setup() failed with rc: <code>`	Reset or replace the adapter.	MOD_CONFIG	CE_WARN
`Pending buffers <code> cannot destroy RQ <code> cache`	Reboot the system.	MOD_CONFIG	CE_WARN
`Ring buffer allocation failed`	Increase the memory.	MOD_CONFIG	CE_WARN
`RQ bdesc alloc failed`	Increase the memory.	MOD_CONFIG	CE_WARN
`RQ Buffer Pool  create failed`	Increase the memory.	MOD_CONFIG	CE_WARN
`RQ create failed: <code>`	Hardware error – Download the same firmware image to the adapter and reboot; replace the adapter if necessary.	MOD_CONFIG	CE_WARN
`RQ ring create failed`	Hardware error – Download the same firmware image to the adapter and reboot; replace the adapter if necessary.	MOD_CONFIG	CE_WARN
`RQ shadow ring alloc failed`	Increase the memory.	MOD_CONFIG	CE_WARN
`SLIPORT Error Detected: SLIPORT_ERROR1: <code> SLIPORT_ERROR2: <code>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`WQ Buffer Pool create failed`	Increase the memory.	MOD_CONFIG	CE_WARN
`WQ create failed <code>`	Hardware error – Download the same firmware image to the adapter and reboot; replace the adapter if necessary.	MOD_CONFIG	CE_WARN
`WQ MAP Handles Pool create failed`	Increase the memory.	MOD_CONFIG	CE_WARN
`WQ Packet Desc Pool create failed`	Increase the memory.	MOD_CONFIG	CE_WARN
`WQCQ create failed`	Hardware error – Download the same firmware image to the adapter and reboot; replace the adapter if necessary.	MOD_CONFIG	CE_WARN
`Unrecognized Optics state`	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE
`Optics faulted/incorrectly installed/not installed – Reseat optics. If issue not resolved, replace`	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE
`Optics of two types installed. Remove one optic or install matching pair of optics.`	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE
`Incompatible optics. Replace with compatible optics for card to function.`	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE
`Unqualified optics. Replace with Avago optics for Warranty and Technical Support.`	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE
`Uncertified optics. Replace with Avago-certified optics to enable link operation.`	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE

7.2.2 For Solaris 11

The following table details the log messages provided by the NIC driver for Solaris 11 operating systems.

Table 11 Log Messages for the NIC Driver for Solaris 11

Log Message	Action	MOD_MASK	Severity
Alloc intr failed	Reload the driver.	MOD_CONFIG	CE_WARN
Could not allocate dbuf	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system. Increase the memory.	MOD_CONFIG	CE_WARN
Could not allocate dbuf for getting mac list	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system. Increase the memory.	MOD_CONFIG	CE_WARN
Could not allocate stats_dbuf	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
Could not allocate vstats_dbuf	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
Could not find the MAC <mac addr>	Provide the correct MAC address.	MOD_CONFIG	CE_WARN
Could not get size of bar	Reset the adapter.	MOD_CONFIG	CE_WARN
Could not get supported intrs	Reload the driver.	MOD_CONFIG	CE_WARN
Could not map bar	Reset the adapter.	MOD_CONFIG	CE_WARN
Could not retrieve num_bars	Reset the adapter.	MOD_CONFIG	CE_WARN
CQ allocation failed	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
CQ create failed – status: <code>	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
CQ ring alloc failed	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
ddi_intr_free failed: <ret code>	Reboot the system.	MOD_CONFIG	CE_WARN
Device in error state	Reboot the system.	MOD_CONFIG	CE_WARN
Device Unknown	The device is not supported by the driver.	MOD_CONFIG	CE_WARN
EQ create failed – status:<code>	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN

Table 11 Log Messages for the NIC Driver for Solaris 11 (Continued)

Log Message	Action	MOD_MASK	Severity
`EQ ring alloc failed`	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
`error recovery failed, device is in suspended state`	Reboot the system.	MOD_CONFIG	CE_WARN
`Error recovery failed/timed out`	Reset the system.	MOD_CONFIG	CE_WARN
`Error recovery timed out`	The hardware or driver cannot recover from a serious hardware error. Reset the system.	MOD_CONFIG	CE_WARN
`Failed  to enable interrupt <int num>, type <int type>, cnt <num_vectors>, ret <ret code>`	Reload the driver.	MOD_CONFIG	CE_WARN
`Failed to add interrupt handler`	Reload the driver.	MOD_CONFIG	CE_WARN
`Failed to allocate bmbx`	Reload the driver.	MOD_CONFIG	CE_WARN
`Failed to allocate rings`	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system. Increase the memory.	MOD_CONFIG	CE_WARN
`Failed to Configure RSS - status:<code>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Failed to create WQ ring`	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
`Failed to del  - status:<code>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Failed to delete MAC <mac_add>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Failed to disable the interrupts`	Reset the adapter.	MOD_CONFIG	CE_WARN
`Failed to enable interrupts during IRM`	Reboot the system.	MOD_CONFIG	CE_WARN
`Failed to get controller attributes - status: <code>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Failed to get hardware stats, version <ver num>, status <code>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN

Table 11 Log Messages for the NIC Driver for Solaris 11 (Continued)

Log Message	Action	MOD_MASK	Severity
`Failed to get pport stats`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN

`Failed to get vlan list –`
`status:<code>`
`Failed to configure vlan list –`
`status:<code>`
`Failed to activate vlan list on vf`
	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN

`Failed to get vport stats`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Failed to init rings`	Low system resources. Reboot the system to see if the issue resolves itself. If possible, add more memory.	MOD_CONFIG	CE_WARN
`Failed to remove interrupt`	Reload the driver.	MOD_CONFIG	CE_WARN
`Failed to retrieve intr types`	Unload and then reload the driver.	MOD_CONFIG	CE_WARN
`Failed to set EQ delay –` `status:<code>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Failed to Setup handlers`	Reload the driver.	MOD_CONFIG	CE_WARN
`Failed to Setup handlers during IRM`	Reboot the system.	MOD_CONFIG	CE_WARN
`Failed to setup_interface during IRM`	Reboot the system.	MOD_CONFIG	CE_WARN
`Failed to get firmware version –` `status`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Failed to get firmware configuration – status`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`FUNCTION RESET FAILED`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Hardware UE Detected`	Upgrade the firmware to the correct version.	MOD_CONFIG	CE_WARN

Table 11 Log Messages for the NIC Driver for Solaris 11 (Continued)

Log Message	Action	MOD_MASK	Severity
`HW initialization failed with ret_code`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`HW POST1 FAILED`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Interrupt block disable failed`	Reset the adapter.	MOD_CONFIG	CE_WARN
`Interrupt setup failed with <ret_code>`	Reload the driver.	MOD_CONFIG	CE_WARN
`Interrupts block enable failed`	Reload the driver.	MOD_CONFIG	CE_WARN
`Invalid q length. Must be [256, 2000]`	Set the correct queue length using the dladm utility.	MOD_CONFIG	CE_WARN
`IRM: Alloc intr failed: ret <ret code> count <code> nallocd <alloc_vectors>`	Reload the driver.	MOD_CONFIG	CE_WARN
`kstat creation failed`	Reload the driver.	MOD_CONFIG	CE_WARN
`kstat setup Failed with <ret_code>`	Reload the driver.	MOD_CONFIG	CE_WARN
`lancer_physdev_reset INP bit polling timedout: <timeout> pd_ctrl: <reg>`	Reset the adapter.	MOD_CONFIG	CE_WARN
`Legacy MQ create failed`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`Legacy MQ ring alloc failed`	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
`Failed to read MAC address – status`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
`MAC allocation Failed`	Unload and then reload the driver.	MOD_CONFIG	CE_WARN
`MAC registration failed`	Unload and then reload the driver.	MOD_CONFIG	CE_WARN
`MAC unregister failed. Detach failed.`	Reboot the system.	MOD_CONFIG	CE_WARN
`Mailbox initialization failed with <ret code>`	Check for the faulty hardware using `fmadm faulty`. If the hardware is faulty, acquit the hardware using `fmadm acquit <dev_path>` and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN

Table 11 Log Messages for the NIC Driver for Solaris 11 (Continued)

Log Message	Action	MOD_MASK	Severity
MAP FAILED	Low System resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_TX	CE_WARN
Map PCI config failed with <ret_code>	Reset the adapter.	MOD_CONFIG	CE_WARN
mcast ADD/DEL failed	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
mod_install failed rval	Reset the adapter.	MOD_CONFIG	CE_WARN
MSIX not available	The system continues to work with INTx.	MOD_CONFIG	CE_WARN
oce_detach : Rx buffers pending with stack	Reboot the system.	MOD_CONFIG	CE_WARN
OCE_INITIATE_DUMP Dump Image not present	Diagnostic Dump image is not present. Retry the Dump initiation.	MOD_CONFIG	CE_WARN
OCE_INITIATE_DUMP failed	Diagnostic Dump initiation by an application failed. Retry the Dump initiation.	MOD_CONFIG	CE_WARN
OCE_INITIATE_DUMP failed due to recovery timeout	Device Recovery failed after Diagnostic Dump initiation. Reboot the system.	MOD_CONFIG	CE_WARN
oce_isr EQ Not started	Unplumb and plumb the device. If the error persists reload the driver.	MOD_CONFIG	CE_WARN
oce_rx:no frags?	The system cannot receive data because the driver is low on resources. Reload the driver. If the same error recurs, reset the adapter.	MOD_RX	CE_WARN
PCI initialization failed	Reset the adapter.	MOD_CONFIG	CE_WARN
POST ERROR	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
Ring buffer allocation failed <code>	Configure the server with more memory.	MOD_CONFIG	CE_WARN
RQ bdesc alloc failed - status: <code>	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
RQ create failed	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
RQ ring create failed	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
RQ shadow ring alloc failed	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN

Table 11 Log Messages for the NIC Driver for Solaris 11 (Continued)

Log Message	Action	MOD_MASK	Severity
soft_reset bit asserted[1]. Reset failed	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
UE Detected or FW Dump is requested SLIPORT_ERR1 = <code> and SLIPORT_ERR2 = <code>	Reboot the system.	MOD_CONFIG	CE_WARN
Unable to get intr priority	Reload the driver.	MOD_CONFIG	CE_WARN
Failed to create an interface - VPORT_SRIOV - staus: <code> Failed to create an interface - VPORT_NORMAL - staus: <code>	Upgrade the firmware to the correct version.	MOD_CONFIG	CE_WARN
WCCQ create failed	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
WQ Buffer Pool create failed	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
WQ create failed - status: <code>	Check for the faulty hardware using fmadm faulty. If the hardware is faulty, acquit the hardware using fmadm acquit <dev_path> and reset the adapter. If the hardware is not faulty, reset the adapter without issuing acquit.	MOD_CONFIG	CE_WARN
WQ MAP Handles Pool create failed	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
WQ Packet Desc Pool create failed	Low system resources. Repeat the operation that generated the log. If the error log is generated again, reboot the system.	MOD_CONFIG	CE_WARN
wqb pool empty	The system cannot transmit data because the driver is low on resources. Check for a process generating heavy traffic.	MOD_TX	CE_WARN
wqm pool empty	The system cannot transmit data because the driver is low on resources. Check for a process generating heavy traffic.	MOD_TX	CE_WARN
Unrecognized Optics state	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE
Optics faulted/incorrectly installed/not installed – Reseat optics. If issue not resolved, replace	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE
Optics of two types installed. Remove one optic or install matching pair of optics.	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE

Table 11 Log Messages for the NIC Driver for Solaris 11 (Continued)

Log Message	Action	MOD_MASK	Severity
Incompatible optics. Replace with compatible optics for card to function.	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE
Unqualified optics. Replace with Avago optics for Warranty and Technical Support.	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE
Uncertified optics. Replace with Avago-certified optics to enable link operation.	Replace with a compatible SFP.	MOD_CONFIG	CE_WARN/CE_NOTE