Sun Quad GbE UTP x8 PCI Express Card

User's Guide

Part No. 820-0115-16
May 2010, Revision A

Please
Recycle

Contents

Declaration of Conformity

To receive a copy of the latest Declaration of Conformity (DoC) for the product, either contact your local Sun sales representative, or create an online request at:

https://www2.sun.de/dct/forms/reg_us_1607_755_0.jsp

Using This Documentation

The *Sun Quad GbE UTP x8 PCI Express Card User's Guide* provides instructions for installing both the hardware and software for Oracle's express card. This manual also describes how to configure the `nxge` driver, which controls the card.

These instructions are designed for enterprise system administrators with experience in installing network hardware and software.

How This Document Is Organized

Chapter 1 provides an overview of the Sun Quad GbE UTP x8 PCI Express Card.

Chapter 2 explains how to install the `nxge` device driver software in a Solaris SPARC environment.

Chapter 3 describes how to install the Sun Quad GbE UTP x8 PCI Express Card in your system and verify that it has been installed correctly.

Chapter 4 describes how to edit the network host files after the adapter has been installed on your system. This chapter also describes how to set up a gigabit Ethernet port on a diskless client and install the Solaris Operating System over a gigabit Ethernet network.

Chapter 5 explains how to set the `nxge` device driver parameters to customize each device in your system.

Chapter 6 describes how to enable the Jumbo Frame feature.

Chapter 7 describes how to configure link aggregation.

Chapter 8 explains Virtual Local Area Networks (VLANs) in detail and provides configuration instructions and examples.

Appendix A lists the specifications for the Sun Quad GbE UTP x8 PCI Express Card.

Appendix B provides an overview of the SunVTS™ diagnostic application and instructions for updating the SunVTS software to recognize the adapter.

Using UNIX Commands

This document might not contain information about basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices. Refer to the following for this information:

- Software documentation that you received with your system
- Oracle Solaris Operating System documentation, which is at:

  `http://docs.sun.com`

Shell Prompts

Shell	Prompt
C shell	*machine-name*%
C shell superuser	*machine-name*#
Bourne shell and Korn shell	$
Bourne shell and Korn shell superuser	#

Related Documentation

The documents listed as online are available at:

http://docs.sun.com/app/docs/prod/net.inter.crds

Application	Title	Part Number	Format	Location
Release Notes	*Sun Quad GbE UTP x8 PCI Express Card Release Notes*	820-0116	PDF HTML	Online
Safety and compliance	*Safety and Compliance Manual*	821-1590	PDF HTML	Online

Documentation, Support, and Training

Sun Function	URL
Documentation	http://docs.sun.com/
Support	http://www.sun.com/support/
Training	http://www.sun.com/training/

Documentation Feedback

Submit comments about this document by clicking the Feedback [+] link at:

http://docs.sun.com

Include the title and part number of your document with your feedback:

Sun Quad GbE UTP x8 PCI Express Card User's Guide, part number 820-0115-16

CHAPTER 1

Product Overview

This chapter provides an overview of the Sun Quad GbE UTP x8 PCI Express Card, including:

Shipping Kit Contents

The carton in which your Sun Quad GbE UTP x8 PCI Express Card was shipped should contain the following items:

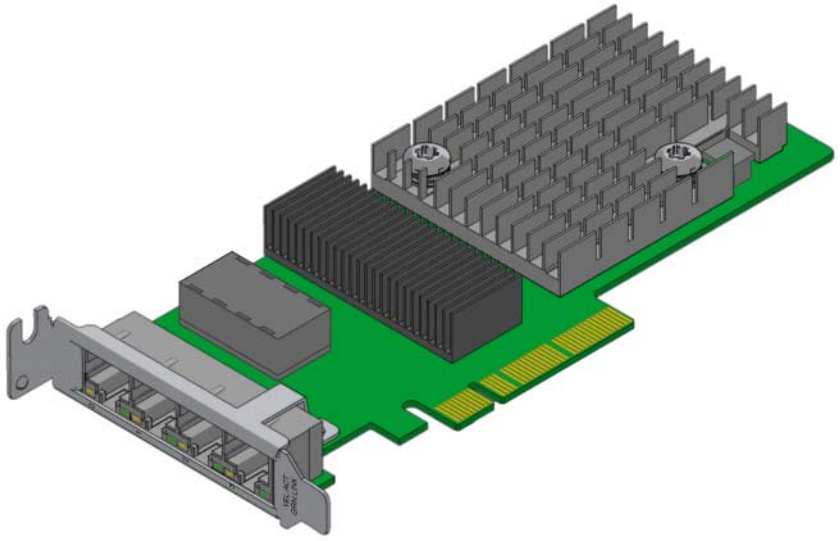
- Sun Quad GbE UTP x8 PCI Express Card with a low profile bracket attached.
- Standard height bracket, screws and washer, and installation instructions
- *Sun Quad GbE UTP x8 Express Card Getting Started Guide*.

Product Description

The Sun Quad GbE UTP x8 PCI Express Card provides high performance packet processing capability optimized for throughput computing and throughput networking architecture. It provides four 10/100/1000Base-T compliant Ethernet copper ports.

FIGURE 1-1 shows the PCI-express card.

FIGURE 1-1 Sun Quad GbE UTP x8 PCI Express Card



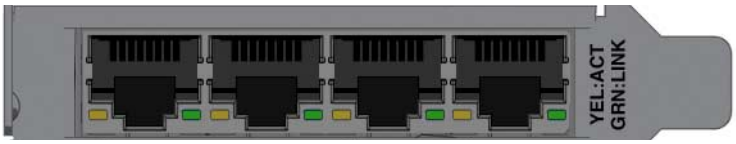
Hardware and Software Requirements

Before installing the adapter, make sure your system meets the required hardware and software. TABLE 1-1 lists the supported hardware and software.

TABLE 1-1 Hardware and Software Requirements

Requirements	Hardware or Software
Hardware	Sun Fire T1000, Sun Fire T2000, Sun Fire V445, Sun Fire V245, Sun Fire V215, Sun Fire U45
Operating System	Solaris 10 11/06 Operating System Windows XP Profession, SP2, 32-bit/64-bit Windows Server 2003 Standard Edition, SP1, 32-bit/64-bit Windows Server 2003 Enterprise Edition, SP1, 32-bit/64-bit

FIGURE 1-2 Sun Quad GbE UTP x8 PCI Express Card Ports



Features

The Sun Quad GbE UTP x8 PCI Express Card addresses the following requirements and provides additional features and benefits:

- Four 10/100/1000Base-T Ethernet interfaces
- IEEE 802.3ab compliant
- Uses Sun's ASIC and software for innovative throughput networking design
- Networking I/O virtualization supporting Solaris LDOM 1.0. (VMware support planned for mid 07)
- Hardware-based flow classification for extending parallelism and virtualization to networking
- Up to 16 Receive DMA channels and up to 24 Transmit DMA channels, multiple receive and transmit Descriptor Rings and dedicated networking hardware resources (DMA, interrupts, buffer, and more) for each thread or strand
- CPU/thread affinity and CPU load balancing at L1,L2,L3 and L4
- Dynamic Reconfiguration (DR)
- Jumbo frame support (up to 9KBytes)
- IPv4/IPv6 and IPMP support
- TCP/UDP/IP checksum and CRC32C support
- IEEE 802.1Q VLAN support
- IEEE 802.1AD Link Aggregation support

CHAPTER 2

Installing and Setting Up the Driver

This chapter explains how to download and install the `nxge` driver. The nxge gigabit Ethernet driver (`nxge`(7D)) is a multi-threaded, loadable, clonable, GLD-based STREAMS driver. The `nxge` driver is managed by the `dladm`(1M) command line utility, which allows VLANs to be defined on top of `nxge` instances and for `nxge` instances to be aggregated. See the `dladm`(1M) man page for more details on configuring the data-link interfaces and link aggregations.

This chapter contains the following sections:

Downloading and Installing the Driver on a Solaris SPARC Platform

This section explains how to download and install the `nxge` driver on a Solaris SPARC platform.

▼ To Download the Driver on a Solaris SPARC Platform

1. **Locate and download the `nxge` device driver software at the following web site:**

   `http://www.sun.com/products/networking/ethernet/index.html`

2. **Uncompress the gzipped tar file:**

TABLE 1

```
# gunzip nxge.tar.gz
```

3. **Unpack the tar file:**

TABLE 2

```
# tar xvf nxge.tar
```

   a. **For SPARC systems, change to the following directory:**

TABLE 2-1

```
# cd 10_GigabitEthernet/Solaris_10/sparc/Packages
```

4. **For SPARC systems, determine which architecture your system is running:**

TABLE 2-2

```
# uname -m
```

   a. **For sun4v systems, install the software packages by typing the following at the command line:**

TABLE 2-3

```
# /usr/sbin/pkgadd -d . SUNWnxge.v SUNWnxgem
```

b. For sun4u systems, install the software packages by typing the following at the command line:

TABLE 2-4

```
# /usr/sbin/pkgadd -d . SUNWnxge.u SUNWnxgem
```

A menu similar to the following displays:

TABLE 3

```
The following packages are available:

  1  SUNWnxge.u     Sun x8 10G/1G Ethernet Adapter Driver
                    (sparc.sun4u) 1.0,REV=2007.01.12.10.0
  2  SUNWnxge.v     Sun x8 10G/1G Ethernet Adapter Driver
                    (sparc.sun4v) 1.0,REV=2007.01.12.10.0
  3  SUNWnxgem      Sun x8 10G/1G Ethernet Adapter Driver Man Pages
                    (all) 1.0,REV=2007.01.12.10.0
Select package(s) you wish to process (or 'all' to process
all packages). (default: all) [?,??,q]:
```

5. Select the packages you want to install:

- Press Return or type **all** to accept the default and install all packages.
- Type the specific numbers, separated by a space, if you prefer not to install any optional packages.

6. **Verify that the** `nxge` **driver is installed on the system:**

TABLE 2-5

```
Sun x8 10G Ethernet Adapter Driver(sparc.sun4v) 1.0,REV=2007.01.06.10.0
Copyright 2007 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.
## Executing checkinstall script.
Using </> as the package base directory.
## Processing package information.
## Processing system information.
   8 package pathnames are already properly installed.
## Verifying package dependencies.
## Verifying disk space requirements.
## Checking for conflicts with packages already installed.
## Checking for setuid/setgid programs.

This package contains scripts which will be executed with super-user
permission during the process of installing this package.

Do you want to continue with the installation of <SUNWnxge> [y,n,?] y

Installing Sun x8 10G Ethernet Adapter Driver as <SUNWnxge>

## Installing part 1 of 1.
/platform/sun4v/kernel/drv/sparcv9/nxge
[ verifying class <none> ]
[ verifying class <preserve> ]
## Executing postinstall script.

Installation of <SUNWnxge> was successful.
```

▼ To Remove the Driver from a Solaris SPARC Platform

1. **To discover the driver packages run the** `pkginfo` **command:**

TABLE 2-6

```
# pkginfo | grep SUNWnxge
system      SUNWnxge       Sun x8 10G Ethernet Adapter Driver
```

2. To remove the driver packages run the `pkgrm` command:

TABLE 2-7

`pkgrm SUNWnxge` *plus any other packages from the previous command*

CHAPTER 3

Installing the PCI Express Card

This chapter describes how to install the Sun Quad GbE UTP x8 PCI Express Card in your system and verify that it has been installed correctly.

This chapter contains the following section:

Replacing the Bracket

This section describes how to uninstall the bracket or faceplate and install one of a different size to fit in your server.

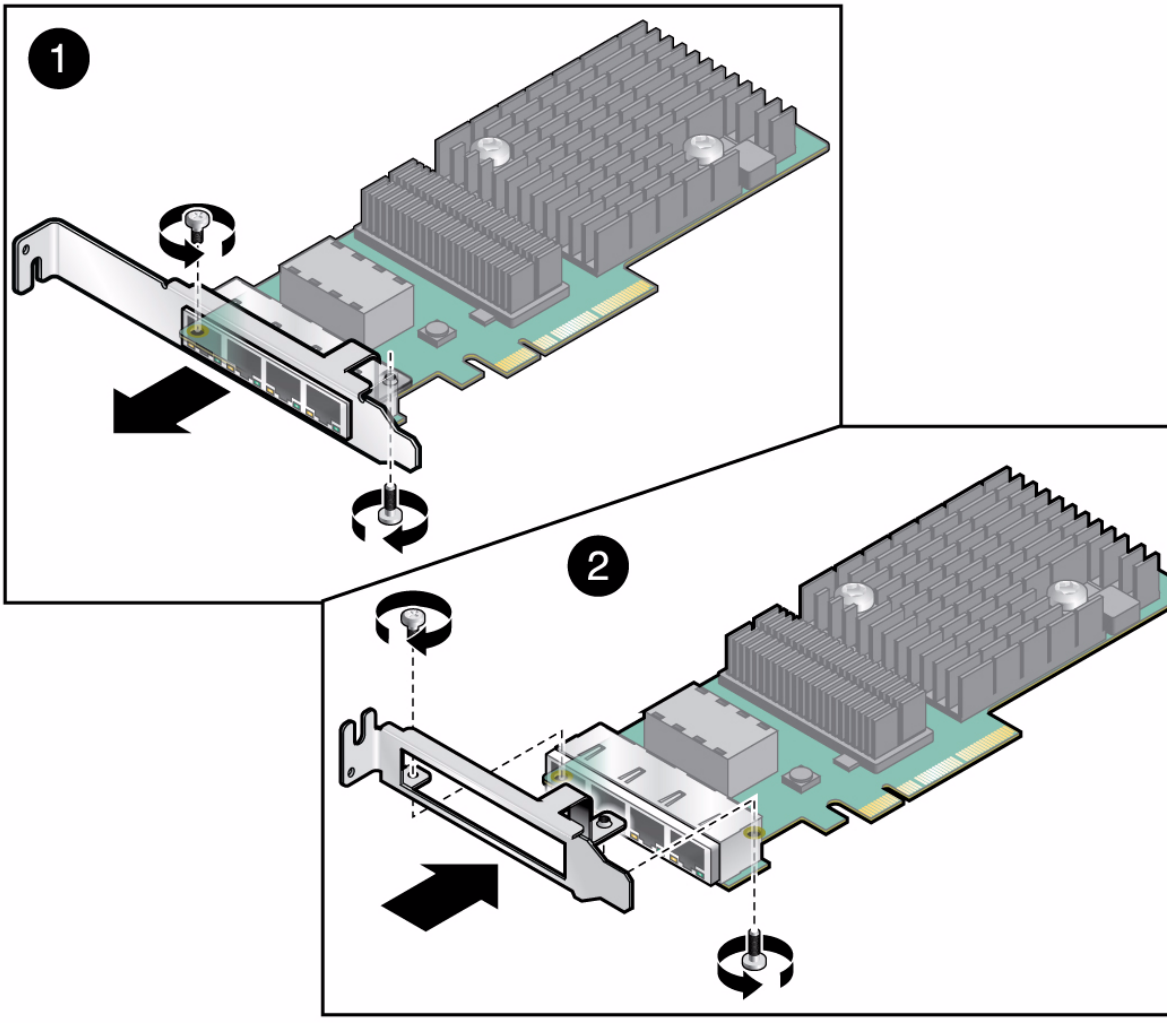
Note – The Sun Quad GbE UTP x8 PCI Express Card ships with a low profile or short bracket attached. One long screw and one short screw secure the bracket in place.

▼ To Replace the Bracket

1. **Unscrew both screws, each on an opposite side of the card (**FIGURE 3-1**).**
2. **Remove the bracket (**FIGURE 3-1**).**

FIGURE 3-1 shows replacing the bracket on the card.

FIGURE 3-1 Replacing the Bracket



3. **Fit the new bracket into place and align the screw holes (**FIGURE 3-1**).**
4. **Insert and tighten the screws (**FIGURE 3-1**).**
   Insert the long screw in the solder side of the card.

Installing the PCI Express Card

If you are installing the Sun Quad GbE UTP x8 PCI Express Card into a machine running Solaris 10, you *must* install the software *before* you install the hardware.

The following instructions describe the basic tasks required to install the PCI-express card. Refer to your system installation or service manual for detailed low profile PCI-express card installation instructions.

Caution – Sun Quad GbE UTP x8 PCI Express Card mechanical fit limitations: Do *not* install the Sun Quad GbE UTP x8 PCI Express Card in slot-2 of a Sun Fire T2000 system. The SATA cables from the adjacent card slot prevent the proper installation of an Sun Quad GbE UTP x8 PCI Express Card in this slot. (Note that this issue does not exist for Sun Fire T2000 that is RoHS compliant.)

Caution – Do *not* install the Sun Quad GbE UTP x8 PCI Express Card in slot-5 of a Sun Fire X4600 system because the heatsink on the PCI-express card will sit against the daughter card of the Sun Fire X4600 system.

▼ To Install the PCI Express Card

1. **Halt and power off your system.**
2. **Power off all of the peripherals connected to your system.**
3. **Open the system unit.**
4. **Attach the adhesive copper strip of the antistatic wrist strap to the metal casing of the power supply. Wrap the other end twice around your wrist, with the adhesive side against your skin.**
5. **Remove the filler panel from the PCI-E opening.**
6. **Holding the PCI-express card by the edges, align the PCI-express card edge connector with the PCI-E slot. Slide the PCI-express card face plate into the small slot at the end of the PCI-E opening.**
7. **Applying even pressure at both corners of the PCI-express card, push the PCI-Express PCI-express card until it is firmly seated in the slot.**

Caution – Do not use excessive force when installing the PCI-express card into the PCI-E slot. You might damage the PCI-express card's PCI connector. If the PCI-express card does not seat properly when you apply even pressure, remove the PCI-express card and carefully reinstall it.

8. **Detach the wrist strap and close the system unit.**

Verifying the Hardware Installation

After you have installed the Sun Quad GbE UTP x8 PCI Express Card, but *before* you boot your system, perform the following tasks to verify the installation. Refer to the your Solaris documentation for the detailed instructions.

▼ To Verify the Hardware Installation

1. **Power on the system, and when the banner appears, press the Stop-A key sequence to interrupt the boot process and display the OpenBoot (`ok`) prompt.**

2. List the network devices on your system.

Table 0-1

```
ok show-nets
a) /pci@7c0/pci@0/pci@9/network@0,3
b) /pci@7c0/pci@0/pci@9/network@0,2
c) /pci@7c0/pci@0/pci@9/network@0,1
d) /pci@7c0/pci@0/pci@9/network@0
e) /pci@7c0/pci@0/pci@8/network@0,1
f) /pci@7c0/pci@0/pci@8/network@0
g) /pci@7c0/pci@0/pci@2/network@0,1
h) /pci@7c0/pci@0/pci@2/network@0
i) /pci@780/pci@0/pci@1/network@0,1
j) /pci@780/pci@0/pci@1/network@0
m) MORE SELECTIONS
q) NO SELECTION
```

Checking the `.properties` output for each device is the surest way to identify the device. Since the Sun Quad GbE UTP x8 PCI Express Card has four ports, the `show-nets` command displays four lines for the PCI-express card, usually `/pci@7c0/pci@0/pci@8` or `/pci@7c0/pci@0/pci@9` correspond to PCIe slots, so look at those devices first.

TABLE 3-1

```
a) /pci@7c0/pci@0/pci@9/network@0,3
b) /pci@7c0/pci@0/pci@9/network@0,2
c) /pci@7c0/pci@0/pci@9/network@0,1
d) /pci@7c0/pci@0/pci@9/network@0
```

Note – If you do not see the device listed, check that the PCI-express card is properly seated. If necessary, reinstall the PCI-express card.

---

3. View the device that you installed.

Using the previous example, type:

TABLE 3-2

```
ok cd /pci@7c0/pci@0/pci@9/network@0
```

4. **View the** `.properties` **file for a list of device properties.**

   The `.properties` command displays the specific information about the installed PCI-express card. If you are using the Sun Quad GbE UTP x8 PCI Express Card, your output will be similar to the following:

TABLE 3-3

```
ok .properties
assigned-addresses        82090010 00000000 08000000 00000000 01000000
                          82090018 00000000 09000000 00000000 00008000
                          82090020 00000000 09008000 00000000 00008000
                          82090030 00000000 09100000 00000000 00100000
local-mac-address         00 14 4f 6c 88 08
phy-type                  mif
reg                       00090000 00000000 00000000 00000000 00000000
                          03090010 00000000 00000000 00000000 01000000
                          03090018 00000000 00000000 00000000 00008000
                          03090020 00000000 00000000 00000000 00008000
                          03090030 00000000 00000000 00000000 00100000
version                   QGC 1G Ethernet Adapter FCode 2.8 06/12/13
board-model               501-7606-03
model                     SUNW,pcie-qgc
compatible                pciex108e,abcd.108e.0.1
                          pciex108e,abcd.108e.0
                          pciex108e,abcd.1
                          pciex108e,abcd
                          pciexclass,020000
                          pciexclass,0200
address-bits              00000030
max-frame-size            00002400
network-interface-type    ethernet
device_type               network
name                      network
fcode-rom-offset          00010000
interrupts
                          00000001
class-code                00020000
subsystem-vendor-id       0000108e
revision-id               00000001
device-id                 0000abcd
vendor-id                 0000108e
ok
```

5. **Type the following when you finish looking at the** `.properties` **values:**

TABLE 3-4

```
ok device-end
```

Rebooting the System

After verifying the PCI-express card installation, use the `boot -r` command to perform a reconfiguration boot on your system.

```
ok boot -r
```

CHAPTER 4

Network Configuration

This chapter describes how to edit the network host files after the PCI-express card has been installed on your system. This chapter also describes how to set up a gigabit Ethernet network on a diskless client and install the Solaris Operating System over a gigabit Ethernet network.

This chapter contains the following sections:

Configuring the Network Host Files

After installing the driver software, you must create a `hostname.nxge`*number* file for the PCI-express card's Ethernet interface. You must also create both an IP address and a host name for its Ethernet interface in the `/etc/hosts` file.

1. **At the command line, use the `grep` command to search the `/etc/path_to_inst` file for nxge interfaces.**

Table 4-1

```
# grep nxge /etc/path_to_inst
"/pci@7c0/pci@0/pci@9/network@0" 0 "nxge"
"/pci@7c0/pci@0/pci@9/network@0,1" 1 "nxge"
"/pci@7c0/pci@0/pci@9/network@0,2" 2 "nxge"
"/pci@7c0/pci@0/pci@9/network@0,3" 3 "nxge"
```

In this example, the device instance is from a Sun Quad GbE UTP x8 PCI Express Card installed in slot 1. The instance numbers are shown in ***bold italics***.

Be sure to write down your device path and instance, which in the example is "/pci@7c0/pci@0/pci@9/network@0" 0. Your device path and instance will be similar. You need this information to make changes to the `nxge.conf` file. See "Setting Parameters Using the `nxge.conf` File" on page 28.

2. **Use the `ifconfig` command to set up the PCI-express card's nxge interface.**

Use the `ifconfig` command to assign an IP address to the network interface. Type the following at the command line, replacing *ip-address* with the PCI-express card's IP address:

```
# ifconfig nxge0 plumb ip-address up
```

Refer to the `ifconfig`(1M) man page and the Solaris documentation for more information.

- If you want a setup that remains the same after you reboot, create an `/etc/hostname.nxge`*number* file, where *number* is the instance number of the nxge interface you plan to use.

  To use the PCI-express card's nxge interface in the Step 1 example, create an `/etc/hostname.nxge0` file, where `0` is the number of the nxge interface. If the instance number were `1`, the filename would be `/etc/hostname.nxge1`. The filename would be `/etc/hostname.nxge2`, for instance 2, and so on.

- Do not create an `/etc/hostname.nxge`*number* file for a Sun Quad GbE UTP x8 PCI Express Card interface you plan to leave unused.
- The `/etc/hostname.nxge`*number* file must contain the host name for the appropriate `nxge` interface.
- The host name must have an IP address listed in the `/etc/hosts` file.

- The host name must be different from any other host name of any other interface, for example: `/etc/hostname.nxge0` and `/etc/hostname.nxge1` cannot share the same host name.

  The following example shows the `/etc/hostname.nxge`*number* file required for a system called `zardoz` that has an Sun Quad GbE UTP x8 PCI Express Card. Note the system name changes for each interface.

Table 0-1

```
# cat /etc/hostname.nxge0
zardoz
# cat /etc/hostname.nxge1
zardoz-11
# cat /etc/hostname.nxge2
zardoz-12
# cat /etc/hostname.nxge3
zardoz-13
```

3. **Create an appropriate entry in the `/etc/hosts` file for each active `nxge` interface.**

   For example:

Table 0-2

```
# cat /etc/hosts
#
# Internet host table
#
127.0.0.1     localhost
129.144.10.57 zardoz    loghost
129.144.11.83 zardoz-11
129.144.12.92 zardoz-12
129.144.13.45 zardoz-13
```

Setting Up a Gigabit Ethernet Port on a Diskless Client System

Before you can boot and operate a diskless client system across a Gigabit Ethernet port, you must first install the Gigabit Ethernet software packages into the root directory of the diskless client. You can find the Gigabit Ethernet software packages at the following web site:

http://www.sun.com/download/index.jsp?cat=Networking&tab=3&subcat=Network%20Connectivity

Refer to the *Solaris Advanced Installation Guide* and the *System Administration Guide* for more information about installing and administering diskless client systems.

▼ To Set Up a Gigabit Ethernet Port on a Diskless Client

1. **Locate the root directory of the diskless client on the host server.**

   The root directory of diskless client system is commonly installed in the host server's `/export/root/`*client-name* directory, where *client_name* is the diskless client's host name. In this procedure, the root directory is:

TABLE 4-2

`/export/root/`*client-name*

2. **Download the software for Sun Quad GbE UTP x8 PCI Express Card onto the server's drive.**

3. **Use the `pkgadd -R` command to install the software packages to the diskless client's root directory on the server.**

   Install the software packages to the client's root directory.

4. **Create a `hostname.nxge`*number* file in the diskless client's root directory.**

   Create an `/export/root/`*client-name*`/etc/hostname.nxge`*number* file for the Gigabit Ethernet interface. See "Configuring the Network Host Files" on page 19 for instructions.

5. **Edit the `hosts` file in the diskless client's root directory.**

   Edit the `/export/root/`*client-name*`/etc/hosts` file to include the IP address of the Gigabit Ethernet interface. See "Configuring the Network Host Files" on page 19 for instructions.

6. **Set the MAC address on the server side and rebuild the device tree if you want to boot from the Gigabit Ethernet port.**

7. **To boot the diskless client from the Gigabit Ethernet port, type the following boot command:**

TABLE 4-3

`ok` **`boot`** *path-to-device:link-param*

Installing the Solaris Operating System Over a Gigabit Ethernet Network

The *Solaris Advanced Installation Guide* describes the full procedure for installing the Solaris Operating System over the network. The following procedure assumes that you have created an install server, which contains the image of the Solaris CD, and that you have set up the client system to be installed over the network.

Before you can install the Solaris Operating System on a client system with a Gigabit Ethernet PCI-express card, you must first add the Gigabit Ethernet software packages to the install server. See Chapter 2 for the location of these software packages.

Note – Refer to the *Solaris Advanced Installation Guide* for more information about installing the Solaris Operating System over the network.

▼ To Install the Solaris Operating System Over a Gigabit Ethernet Network

1. **Prepare the install server and client system to install the Solaris Operating System over the network.**

   The *Solaris Advanced Installation Guide* describes how to create the install server and set up the client systems.

Note – If you want to install the client system over a network that is not part of the same subnet, you must also create a boot server. The *Solaris Advanced Installation Guide* describes how to create a boot server.

---

2. **Find the root directory of the client system.**

   The client system's root directory can be found in the install server's `/etc/bootparams` file. Use the `grep` command to search this file for the root directory.

TABLE 4-4

```
# grep client-name /etc/bootparams
client_name root=server-name:/netinstall/Solaris_10/Tools/Boot
install=server-name:/netinstall boottype=:in rootopts=:rsize=32768
```

   In this example, the root directory for the Solaris 10 client is `/netinstall`. In Step 4, you would replace *root-directory* with `/netinstall`.

Note – If the root directory is not found in the `/etc/bootparams` file, refer to the *Solaris Advanced Installation Guide* for configuration instructions.

---

3. **Download the `nxge` driver onto the install server's hard drive.**

   The package is a folder SUNWnxge.v or SUNWnxge.u, which you can download from the following web site:

   `http://www.sun.com/download/products.xml?id=44eb1efd`

4. **On the install server, install the `nxge` software to the client's root directory, as determined in Step 2.**

   Replace *root-directory* with the location of the client's root directory.

TABLE 4-5

```
# cd location where you downloaded the packages
# ls SUNWnxge*
# pkgadd -R root-directory/Solaris_10/Tools/Boot -d . SUNWnxge.v
```

Note – Perform the following steps on the *client* system.

---

5. **Shut down and halt the client system.**

   Use the `shutdown` command to go to the OpenBoot (`ok`) prompt.

```
# shutdown -i0 -g0 -y
. . .
(shutdown command messages omitted)
. . .
ok
```

6. **At the `ok` prompt, use the `show-nets` command to find the device path of the Gigabit Ethernet device.**

   The `show-nets` command lists the system devices. You should see the full paths and names of the network devices, similar to the example below.

TABLE 4-6

```
ok show-nets
a) /pci@7c0/pci@0/pci@9/network@0,3
b) /pci@7c0/pci@0/pci@9/network@0,2
c) /pci@7c0/pci@0/pci@9/network@0,1
d) /pci@7c0/pci@0/pci@9/network@0
e) /pci@7c0/pci@0/pci@8/network@0,1
f) /pci@7c0/pci@0/pci@8/network@0
g) /pci@7c0/pci@0/pci@2/network@0,1
h) /pci@7c0/pci@0/pci@2/network@0
i) /pci@780/pci@0/pci@1/network@0,1
j) /pci@780/pci@0/pci@1/network@0
m) MORE SELECTIONS
q) NO SELECTION
Enter Selection, q to quit:
```

7. **At the `ok` prompt, boot the client system using the full device path of the Gigabit Ethernet device, for example:**

TABLE 4-7

```
ok boot /pci@7c0/pci@0/pci@9/network@0
```

8. **Proceed with the Solaris Operating System installation.**

   Refer to the *Solaris Advanced Installation Guide* for more information about installing the Solaris Operating System over the network.

9. **After installing the Solaris Operating System, install the `nxge` driver software on the client system.**

   The software installed in Step 4 is required to boot the client system over the Gigabit Ethernet interface. You now need to install the software in order for the operating system to use the client's Gigabit Ethernet interfaces in normal operation.

   Before installing the `nxge` driver, ensure that the client system does not already have the driver installed. Use the `pkginfo` command to see if the `nxge` software packages are installed on the client system.

```
# pkginfo | grep SUNWnxge
```

- If the software is installed, the previous command will return the package name you typed in. In that case, skip to Step 10.
- If the software is not installed, install the software from the download center.

  See Chapter 2 for instructions on installing the required software packages.

10. **Confirm that the network host files have been configured correctly during the Solaris installation.**

    Although the Solaris software installation creates the client's network configuration files, you may need to edit these files to match your specific networking environment. See "Configuring the Network Host Files" on page 19 for more information about editing these files.

11. **Use the `dladm show-dev` command to show configuration information for all data-links or the specified data-link. By default, the system is configured to have one data-link for each known network device.**

TABLE 4-8

```
# dladm show-dev
e1000g0         link: up        speed: 1000  Mbps       duplex: full
e1000g1         link: down      speed: 0     Mbps       duplex: half
nxge2           link: up        speed: 1000  Mbps       duplex: full
nxge3           link: up        speed: 1000  Mbps       duplex: full
nxge4           link: up        speed: 1000  Mbps       duplex: full
nxge5           link: up        speed: 1000  Mbps       duplex: full
e1000g2         link: down      speed: 0     Mbps       duplex: half
e1000g3         link: down      speed: 0     Mbps       duplex: half
```

CHAPTER 5

Configuring the nxge Device Driver Parameters

The nxge device driver controls the Sun Quad GbE interfaces. You can manually set the nxge driver parameters to customize each device in your system.

This chapter lists the available device driver parameters and describes how you can set these parameters.

nxge Hardware and Software Overview

The Sun Quad GbE UTP x8 PCI Express Card provides four Gigabit Full Duplex networking interfaces. The device driver automatically sets the link speed to 1000 Mbit/sec and conforms to the IEEE 802.3 Ethernet standard. Each interface has 4 Receive DMA Channels and 6 Transmit DMA Channels to allow for parallel processing of the packets. The Sun Quad GbE UTP x8 PCI Express Card extends CPU and OS parallelism to networking with its support for hardware-based flow classification and multiple DMAs. Using CPU thread affinity to bind a given flow to a specific CPU thread, it enables a one-to-one correlation of Rx and Tx packets across the same TCP connection. This can help avoid cross-calls and context switching to deliver greater performance while reducing the need for CPU resources to support I/O processing. The Sun Quad GbE UTP x8 PCI Express Card utilizes Sun's own innovative MAC Controller.

Setting Parameters Using the nxge.conf File

Specify the driver parameter properties for each device by creating a nxge.conf file in the /platform/sun4*u*/kernel/drv/sparcv9 directory or the /platform/sun4*v*/kernel/drv/sparcv9 directory depending on your output of the uname -m command. For example:

TABLE 5-1

```
% uname -m
sun4u
OR
% uname -m
sun4v
```

Use a nxge.conf file when you need to set a particular parameter for a device in the system.

The man pages for prtconf(1M) and driver.conf(4) include additional details. The next procedure shows an example of setting parameters in a nxge.conf file.

- **To access any man page, type the** man **command plus the name of the man page.**

  For example, to access man pages for prtconf(1M), type:

TABLE 5-2

```
% man prtconf
```

▼ To Set Driver Parameters Using an nxge.conf File

1. **Obtain the hardware path names for the** nxge **devices in the device tree.**

   a. **Check the** /etc/driver_aliases **file to identify the name associated with a particular device:**

TABLE 5-3

```
# grep nxge /etc/driver_aliases
nxge "pciex108e,abcd"
```

b. Locate the path names and the associated instance numbers in the `/etc/path_to_inst` file.

```
# grep nxge /etc/path_to_inst
"/pci@7c0/pci@0/pci@9/network@0" 0 "nxge"
"/pci@7c0/pci@0/pci@9/network@0,1" 1 "nxge"
"/pci@7c0/pci@0/pci@9/network@0,2" 2 "nxge"
"/pci@7c0/pci@0/pci@9/network@0,3" 3 "nxge"
```

- In this example:
  - The first part within the double quotes specifies the hardware node name in the device tree.
  - The number not enclosed in quotes is the instance number (shown in ***bold italics*** for emphasis).
  - The last part in double quotes is the driver name.

To identify a PCI-E device unambiguously in the `nxge.conf` file, use the name, parent name, and the unit-address for the device. Refer to the `pci`(4) man page for more information about the PCI-E device specification.

In this example:

- `name = "pciex108e,abcd"`
- `parent = "/pci@7c0/pci@0/pci@9/"`
- `unit-address = "0"`

2. Set the parameters for the `nxge` devices in the `/platform/sun4u/kernel/drv/sparcv9/nxge.conf` file or the `/platform/sun4v/kernel/drv/sparcv9/nxge.conf` file.

a. The following parameters can be set using the `nxge.conf` file.

```
#
#---------------Link Configuration ----------------------
#       The link parameters depend on the type of the card
#       and the port.
#       10 gigabit related parameters ( i.e adv_10gfdx_cap)
#       apply only to 10gigabit ports.
#       Half duplex is not supported on any NIU card.
#
#       adv-autoneg-cap
#               Advertise auto-negotiation capability.
#               default is 1
# adv-autoneg-cap = 1;
#
#       adv_10gfdx_cap
```

```
#               Advertise 10gbps Full duplex  capability.
#               default is 1
# adv_10gfdx_cap = 1;
#
#       adv_1000fdx_cap
#               Advertise 1gbps Full duplex  capability.
#               default is 1
# adv_1000fdx_cap = 1;
#
#       adv_100fdx_cap
#               Advertise 100mbps Full duplex  capability.
#               default is 1
# adv_100fdx_cap = 1;
#
#       adv_10fdx_cap
#               Advertise 10mbps Full duplex  capability.
#               default is 1
# adv_10fdx_cap = 1;
#
#       adv_asmpause_cap
#               Advertise Asymmetric pause capability.
#               default is 0
# adv_asmpause_cap = 0;
#
#       adv_pause_cap
#               Advertise pause capability.
#               default is 1
# adv_pause_cap = 1;
#
#
#------- Jumbo frame support ----------------------------------
# To enable jumbo support for all nxge interfaces,
# accept_jumbo = 1;
#
# To disable jumbo support for all nxge interfaces,
# accept_jumbo = 0;
#
# Default is 0.  See the example at the end of this file for
# enabling or disabling jumbo for a particular nxge interface.
#
#
#------- Receive DMA Configuration ----------------------------
#
#  rxdma-intr-time
#       Interrupts after this number of NIU hardware ticks have
#       elapsed since the last packet was received.
#       A value of zero means no time blanking (Default = 8).
#
# rxdma-intr-pkts
```

```
#       Interrupt after this number of packets have arrived since
#       the last packet was serviced. A value of zero indicates
#       no packet blanking (Default = 20).
#
# Default Interrupt Blanking parameters.
#
# rxdma-intr-time = 8;
# rxdma-intr-pkts = 20;
#
#
#------- Classification and Load Distribution Configuration ------
#
# class-opt-****-***
#       These variables define how each IP class is configured.
#       Configuration options range from whether TCAM lookup ie
#       is enabled to flow hash generation.
#       This parameters also control how the flow template is
#        constructed and how packet is distributed within RDC
#       groups.
#
#       supported classes:
#       class-opt-ipv4-tcp class-opt-ipv4-udp class-opt-ipv4-sctp
#       class-opt-ipv4-ah class-opt-ipv6-tcp class-opt-ipv6-udp
#       class-opt-ipv6-sctp class-opt-ipv6-ah
#
#       Configuration bits (The following bits will be decoded
#       by the driver as hex format).
#
#       0010:           use MAC Port (for flow key)
#       0020:           use L2DA (for flow key)
#       0040:           use VLAN (for flow key)
#       0080:           use proto (for flow key)
#       0100:           use IP src addr (for flow key)
#       0200:           use IP dest addr (for flow key)
#       0400:           use Src Port (for flow key)
#       0800:           use Dest Port (for flow key)
#
# class-opt-ipv4-tcp = fe0;
#
```

b. The following parameters operate on a per port basis and can be set using the `nxge.conf` file.

```
#
# ------- How to set parameters for a particular interface --------
# The example below shows how to locate the device path and set a
# parameter for a particular nxge interface. (Using jumbo support as
# an example)
#
```

```
# Use the following command to find out the device paths for nxge,
#       more /etc/path_to_inst | grep nxge
#
# For example, if you see,
#       "/pci@7c0/pci@0/pci@8/network@0" 0 "nxge"
#       "/pci@7c0/pci@0/pci@8/network@0,1" 1 "nxge"
#       "/pci@7c0/pci@0/pci@8/network@0,2" 2 "nxge"
#       "/pci@7c0/pci@0/pci@8/network@0,3" 3 "nxge"
#
# then you can enable jumbo for ports 0 and 1 and disable jumbo for ports 2
# and 3 as follows,
#
# name = "pciex108e,abcd" parent = "/pci@7c0/pci@0/pci@8/" unit-address
= "0"
# accept_jumbo = 1;
# name = "pciex108e,abcd" parent = "/pci@7c0/pci@0/pci@8/" unit-address
= "0,1"
# accept_jumbo = 1;
# name = "pciex108e,abcd" parent = "/pci@7c0/pci@0/pci@8/" unit-address
= "0,2"
# accept_jumbo = 0;
# name = "pciex108e,abcd" parent = "/pci@7c0/pci@0/pci@8/" unit-address
= "0,3"
# accept_jumbo = 0;
```

c. In the following example, the ports of *all* the Sun Quad GbE UTP x8 PCI Express Card are being set for load balancing Rx traffic based on IP source address. The default value is F80 indicating Rx load balancing based on IP 5-tuple. Notice the semi-colon at the end of the last parameter.

TABLE 5-4

```
class-opt-ipv4-tcp = 100;
class-opt-ipv4-udp = 100;
```

d. The following example shows ports on two different cards being set. Only one node needs to be specified.

TABLE 5-5

```
name = "pciex108e,abcd" parent = "/pci@780/pci@0/pci@8/" unit-address = "0"
class-opt-ipv4-tcp = 0x100;

name = "pciex108e,abcd" parent = "/pci@7c0/pci@0/pci@9/" unit-address = "0"
class-opt-ipv4-tcp = 0x40;
```

3. Save the `nxge.conf` file.

CHAPTER 6

Configuring the Jumbo Frames Feature

This chapter describes how to enable the Jumbo Frames feature. It contains the following sections:

Jumbo Frames Overview

Configuring Jumbo Frames enables the Ethernet interfaces to send and receive packets larger than the standard 1500 bytes. However, the actual transfer size depends on the switch capability and the ethernet driver capability.

Note – Refer to the documentation that came with your switch for exact commands to configure Jumbo Frames support.

Checking Jumbo Frames Configurations

The jumbo frames configuration checking occurs at Layer 2 or Layer 3, depending on the configuration method.

▼ To Show the Driver Statistics in a Solaris Environment

1. **Use the `kstat` command to display driver statistics, for example:**

TABLE 6-1

```
# kstat nxge:1 |grep rdc_packets
        rdc_packets                     798982054
        rdc_packets                     792546171
        rdc_packets                     803941759
        rdc_packets                     805674872
```

The previous example diplays the receive packet counts on all of the eight Receive DMA channels on interface 1. Using the `kstat nxge:1` shows all the statistics that the driver supports for that interface.

2. **Use the kstat command to display driver statistics of a VLAN interface, for example:.**

TABLE 6-2

```
# kstat nxge:38001
module: nxge                            instance: 38001
name:   nxge38001                       class:    net
        brdcstrcv                        0
        brdcstxmt                        0
        collisions                       0
        crtime                           3842.493000352
        ierrors                          0
        ifspeed                          10000000000
        ipackets                         2116069805
        ipackets64                       6411037101
        multircv                         0
        multixmt                         0
        norcvbuf                         0
        noxmtbuf                         0
        obytes                           2757388874
        obytes64                         23380264381002
        oerrors                          0
        opackets                         37606022
        opackets64                       4332573318
        rbytes                           2937141290
        rbytes64                         47178857920554
```

Enabling Jumbo Frames in a Solaris SPARC Environment

This section describes how to enable jumbo frames in a Solaris SPARC environment.

▼ To Enable Jumbo Frames in a Solaris Environment Using `nxge.conf`

1. Enable jumbo frames for a port using the `nxge.conf` file. For example,

TABLE 6-3

```
name = "pciex108e,abcd" parent = "/pci@7c0/pci@0/pci@9/"
unit-address = "0"
accept-jumbo=1;
```

2. Reboot the system:

TABLE 6-4

```
% boot -r
```

▼ To Check Layer 2 Configuration

- **View the maximum transmission unit (MTU) configuration of an `nxge` instance at any time with the `kstat` command.**

TABLE 6-5

```
# kstat nxge:0 | grep mac_mtu
```

The `kstat mac_mtu` variable represents the complete size of the Ethernet frame, which includes the Ethernet header, maximum payload, and `crc`. This value should be equal to or less than the MTU configured on the switch.

▼ To Check Layer 3 Configuration

- **Check the Layer 3 configuration by using the `dladm` command with the *show-link* option, as shown in the following example:**

TABLE 6-6

```
# dladm show-link
e1000g0         type: non-vlan mtu: 1500       device: e1000g0
e1000g1         type: non-vlan mtu: 1500       device: e1000g1
e1000g2         type: non-vlan mtu: 1500       device: e1000g2
e1000g3         type: non-vlan mtu: 1500       device: e1000g3
nxge0           type: non-vlan mtu: 9194       device: nxge0
nxge1           type: non-vlan mtu: 9194       device: nxge1
nxge2           type: non-vlan mtu: 9194       device: nxge2
nxge3           type: non-vlan mtu: 9194       device: nxge3
nxge38001       type: vlan 38  mtu: 9194       device: nxge1
```

CHAPTER 7

Configuring Link Aggregation

This chapter describes how to configure link aggregation. It contains the following sections:

Overview of Link Aggregation

Link Aggregation allows one or more network links to be aggregated together to form a link aggregation group. This link aggregation group appears to MAC clients as a regular link. Link aggregation is defined by IEEE 802.2ad and it provides the following benefits:

- Increased bandwidth
- Linearly incremental bandwidth
- Load sharing
- Automatic configuration
- Rapid configuration and reconfiguration
- Deterministic behavior
- Low risk of duplication or mis-ordering
- Support of existing IEEE 802.3 MAC clients

Configuring Link Aggregation in a Solaris Environment

This section explains how to configure link aggregation in a Solaris environment.

▼ To Configure Link Aggregation in a Solaris Environment

1. **Aggregate** `nxge0`, `nxge1`, `nxge2`, **and** `nxge3` **to form an aggregation and a random number 33 as key.**

   a. **Unplumb the interfaces to be aggregated:**

TABLE 7-1

```
# ifconfig down unplumb nxge0
# ifconfig down unplumb nxge1
# ifconfig down unplumb nxge2
# ifconfig down unplumb nxge3
```

b. Create a link-aggregation group with key 33 without specifying mode:

TABLE 7-2

```
# dladm create-aggr -d nxge0 -d nxge1 -d nxge2 -d nxge3 33
```

As the command returns, one line appears in `/etc/aggregation.conf` file and indicates that the default mode is off, as shown in the following example:

TABLE 7-3

```
# tail -1 /etc/aggregation.conf
# Use is subject to license terms.
#
# ident "@(#)aggregation.conf   1.1     05/09/01 SMI"
#
# DO NOT EDIT OR PARSE THIS FILE!
#
# Use the dladm(1m) command to change the contents of this file.

33  L4  4  nxge0/0,nxge1/0,nxge2/0,nxge3/0 auto  off  short
# dladm show-link aggr33
aggr33          type: non-vlan  mtu: 1500       aggregation: key 33
```

2. Plumb up the interface `aggr`*key*, **which is** `aggr33` **is this case:**

TABLE 7-4

```
# ifconfig aggr33 plumb
# ifconfig aggr33
aggr33: flags=1000842<BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 8
        inet 0.0.0.0 netmask 0
        ether 0:3:ba:d8:9d:e8

# ifconfig aggr33 192.168.1.1/24 broadcast + up

# ifconfig aggr33
aggr33: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 8
        inet 192.168.1.1 netmask ffffff00 broadcast 192.168.1.255
        ether 0:3:ba:d8:9d:e8
```

3. **Show link-aggregation status again, and now the state should become attached:**

TABLE 7-5

```
# dladm show-aggr
key: 33 (0x0021)        policy: L4      address: 0:14:4f:6c:11:8 (auto)
           device       address            speed         duplex  link   state
           nxge0        0:14:4f:6c:11:8    1000  Mbps    full    up     attached
           nxge1        0:14:4f:6c:11:9    1000  Mbps    full    up     attached
           nxge2        0:14:4f:6c:11:a    1000  Mbps    full    up     attached
           nxge3        0:14:4f:6c:11:b    1000  Mbps    full    up     attached
```

4. **Use the** `dladm show-aggr -s` **command to display statistics:**

TABLE 7-6

```
# dladm show-aggr -s
key: 33                ipackets  rbytes      opackets   obytes    %ipkts    %opkts
           Total         380354    25872976      28         2648
          nxge0         95089     6468278       7          662       25.0      25.0
          nxge1         95089     6468278       7          662       25.0      25.0
          nxge2         95089     6468278       7          662       25.0      25.0
          nxge3         95087     6468142       7          662       25.0      25.0
```

5. **Use the** `dladm show-aggr -L` **command to display LACP specific information:**

TABLE 7-7

```
# dladm show-aggr -L
key: 33 (0x0021)        policy: L4      address: 0:14:4f:6c:11:8 (auto)
                LACP mode: off  LACP timer: short
    device    activity timeout aggregatable sync  coll dist defaulted  expired
    nxge0     passive  short   yes          no    no   no   no         no
    nxge1     passive  short   yes          no    no   no   no         no
    nxge2     passive  short   yes          no    no   no   no         no
    nxge3     passive  short   yes          no    no   no   no         no
```

For more information refer to the `man` pages for `dladm`, `man dladm`.

CHAPTER 8

Configuring VLANs

This chapter explains Virtual Local Area Networks (VLANs) in detail and provides configuration instructions and examples. It contains the following sections:

With multiple VLANs on a PCI-express card, a server with a single PCI-express card can have a logical presence on multiple IP subnets. By default, 128 VLANs can be defined for each VLAN-aware PCI-express card on your server.

If your network does not require multiple VLANs, you can use the default configuration, in which case no further configuration is necessary.

Overview of VLANs

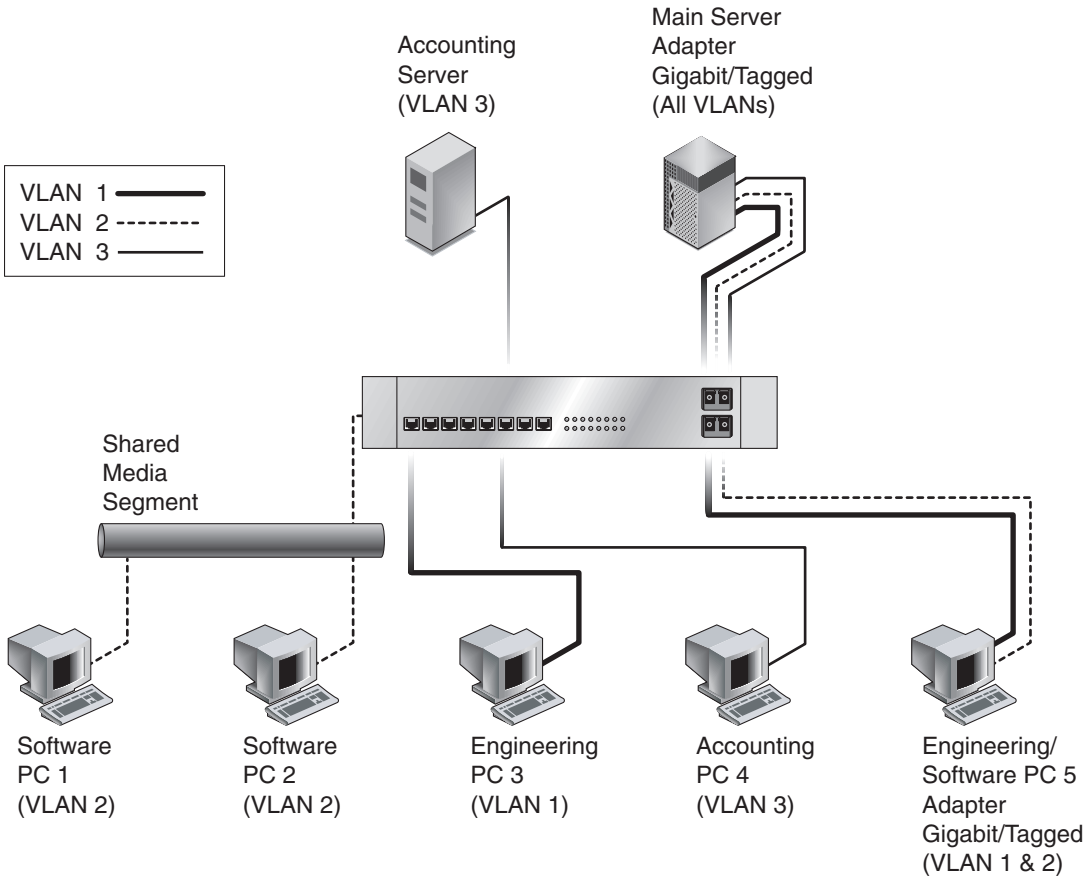
VLANs enable you to split your physical LAN into logical subparts, providing an essential tool for increasing the efficiency and flexibility of your network.

VLANs are commonly used to separate groups of network users into manageable broadcast domains, to create logical segmentation of workgroups, and to enforce security policies among each logical segment. Each defined VLAN behaves as its own separate network, with its traffic and broadcasts isolated from the others, increasing the bandwidth efficiency within each logical group.

Although VLANs are commonly used to create individual broadcast domains and/or separate IP subnets, it can be useful for a server to have a presence on more than one VLAN simultaneously. Several Sun products support multiple VLANs on a per port or per interface basis, allowing very flexible network configurations.

FIGURE 8-1 shows an example network that uses VLANs.

FIGURE 8-1 Example of Servers Supporting Multiple VLANs With Tagging PCI Express Cards



The example network has the following features:

The physical LAN network consists of a switch, two servers, and five clients. The LAN is logically organized into three different VLANs, each representing a different IP subnet.

- VLAN 1 is an IP subnet consisting of the Main Server, Client 3, and Client 5. This represents an engineering group.
- VLAN 2 includes the Main Server, Clients 1 and 2 by means of a shared media segment, and Client 5. This is a software development group.
- VLAN 3 includes the Main Server, the Accounting Server and Client 4. This is an accounting group.

The Main Server is a heavily used server that needs to be accessed from all VLANs and IP subnets. The server has a Sun Quad GbE UTP x8 PCI Express Card installed. All three IP subnets are accessed by means of the single physical PCI-express card interface. The server is attached to one of the switch's Gigabit Ethernet ports, which is configured for VLANs 1, 2, and 3. Both the PCI-express card and the connected switch port have tagging turned on. The tagging VLAN capabilities of both devices enable the sever to communicate on all three IP subnets in this network, yet continue to maintain broadcast separation among the three subnets. The following list describes the components of this network:

- The Accounting Server is available to VLAN 3 only. It is isolated from all traffic on VLANs 1 and 2. The switch port connected to the server has tagging turned off.
- Clients 1 and 2 are attached to a shared media hub that is then connected to the switch. They belong to VLAN 2 only, and are logically in the same IP subnet as the Main Server and Client 5. The switch port connected to this segment has tagging turned off.
- Client 3 is a member of VLAN 1, and can communicate only with the Main Server and Client 5. Tagging is not enabled on Client 3's switch port.
- Client 4 is a member of VLAN 3, and can communicate only with the servers. Tagging is not enabled on Client 4's switch port.
- Client 5 is a member of both VLANs 1 and 2, and has a Sun Quad GbE UTP x8 PCI Express Card installed. It is connected to switch port 10. Both the PCI-express card and the switch port are configured for VLANs 1 and 2 and have tagging enabled.

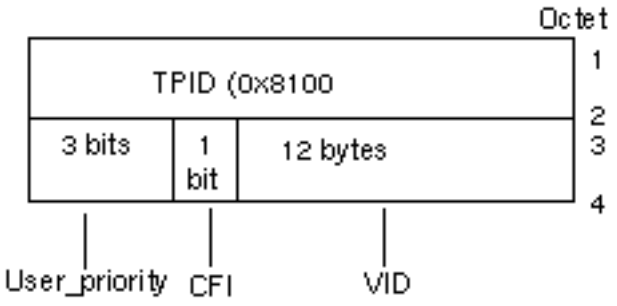
VLAN tagging is only required to be enabled on switch ports that create trunk links to other VLAN-aware Ethernet switches, or on ports connected to tag-capable end-stations, such as servers or workstations with VLAN-aware PCI-express cards.

Configuring VLANs in a Solaris Environment

VLANs can be created according to various criteria, but each VLAN must be assigned a VLAN tag or VLAN ID (VID). The VID is a 12-bit identifier between 1 and 4094 that identifies a unique VLAN. For each network interface (`nxge0`, `nxge1`, `nxge2`, and `nxge3`), 4094 possible VLAN IDs can be selected per port for up to 4 ports. Only 512 unique IDs can be used simultaneously. Because IP subnets are commonly used, it is best to use IP subnets when setting up a VLAN network interface. This means that each VID assigned to a VLAN interface of a physical network interface will belong to different subnets.

Tagging an Ethernet frame requires the addition of a tag header to the frame. The header is inserted immediately following the destination MAC address and the Source MAC address. The tag header consists of two bytes of Ethernet Tag Protocol identifier (TPID, 0x8100) and two bytes of Tag Control Information (TCI). FIGURE 8-2 shows the Ethernet Tag Header format.

FIGURE 8-2 Ethernet Tag Header Format



By default, a single VLAN is configured for every port. This groups all ports into the same broadcast domain, just as if there were no VLANs at all, VLAN tagging for the switch port is turned off.

Note – If you configure a VLAN virtual device for a PCI-express card, all traffic sent or received by that PCI-express card must be in VLAN-tagged format.

▼ To Configure Static VLANs

1. **Create one** `hostname.nxge`*number* **file for each VLAN that will be configured for each PCI-express card on the server.**

   Use the following naming format, which includes both the VID and the physical point of attachment (PPA):

   VLAN logical PPA = *1000 * VID + Device PPA*
   `nxge123000 = 1000*123 + nxge`

   This format limits the maximum number of PPAs (instances) you can configure to 1000 in the `/etc/path_to_inst` file.

   For example, on a server with the Sun Quad GbE UTP x8 PCI Express Card having an instance of 0, belonging to a member of two VLANs, with VID 123 and 224, you would use `nxge123000` and `nxge224000`, respectively, as the two VLAN PPAs.

2. **Use the** `ifconfig(1M)` **to configure a VLAN virtual device, for example:**

TABLE 8-1

```
# ifconfig nxge123000 plumb up
# ifconfig nxge224000 plumb up
```

The output of `ifconfig -a` on a system having VLAN devices `nxge123000` and `nxge224000`:

TABLE 8-2

```
# ifconfig -a
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
        inet 127.0.0.1 netmask ff000000
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
        inet 129.144.131.91 netmask ffffff00 broadcast 129.144.131.255
        ether 8:0:20:a4:4f:b8
nxge123000: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 3
        inet 199.199.123.3 netmask ffffff00 broadcast 199.199.123.255
        ether 8:0:20:a4:4f:b8
nxge224000: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 4
        inet 199.199.224.3 netmask ffffff00 broadcast 199.199.224.225
        ether 8:0:20:a4:4f:b8
```

3. **On the switch, set VLAN tagging and set VLAN ports to coincide with the VLANs you've set up on the server.**

   Using the examples in Step 2, you would set up VLAN ports 123 and 224 on the switch.

   Refer to the documentation that came with your switch for specific instructions for setting VLAN tagging and ports.

APPENDIX **A**

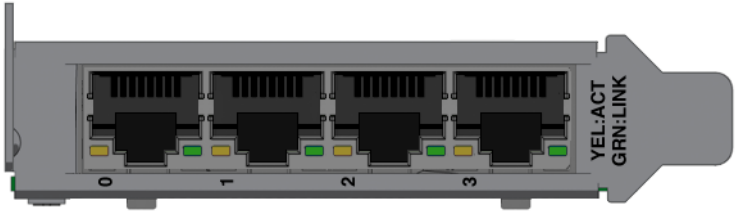
Specifications

This appendix lists the specifications for the Sun Quad GbE UTP x8 PCI Express Card. It contains the following sections:

Connectors

FIGURE A-1 shows the connectors for the Sun Quad GbE UTP x8 PCI Express Card.

FIGURE A-1 Sun Quad GbE UTP x8 PCI Express Card Connectors



The green LED indicates the link status and the yellow LED indicates the link activity.

TABLE A-1 lists the characteristics of the connectors used by the Sun Quad GbE UTP x8 PCI Express Card.

TABLE A-1 Connector Characteristics

Parameter	Description
Connector type	1x4 RJ45
Distance	100 meters

Performance Specifications

TABLE A-2 Performance Specifications

Feature	Specification
Bus type	x8 lane PCI-Express
Bus width	x8 lane PCI-Express
Bus speed (x8, encoded rate)	20 Gbit/sec uni-directional; 40 Gbit/sec bidirectional (theoretical)
Maximum Ethernet transfer rate	1 Gbps (full-duplex)

Physical Characteristics

TABLE A-3 Physical Characteristics

Dimension	Measurement
Length	167.65 mm (6.6 inches)
Width	68.90 mm (2.713 inches)
Height	Component side = 14.48 mm (0.57 inches) maximum Solder side = 2.67 mm (0.105 inches) maximum

Power Requirements

TABLE A-4 Power Requirements

Specification	Measurement
Power consumption	14.51W typical 17.83 W peak
Voltage	12V and 3.3V

APPENDIX **B**

Diagnostic Software

This appendix provides an overview of the SunVTS diagnostic application and instructions for updating the SunVTS software to recognize the PCI-express card. This appendix contains the following sections:

SunVTS Diagnostic Software

The SunVTS software executes multiple diagnostic hardware tests from a single user interface and is used to verify the configuration and functionality of most hardware controllers and devices. The SunVTS software operates primarily from a graphical user interface, enabling test parameters to be set quickly and easily while a diagnostic test operation is being performed.

The SunVTS `nettest` diagnostic can be used to test all of the networking interfaces on the system, including the interfaces on the PCI-express card.

To use the `nettest` diagnostic, you must have the SunVTS software installed on your system. Refer to your Solaris documentation for installation instructions.

Refer to the SunVTS documentation (listed in TABLE B-1) for instructions on how to run and monitor the `nettest` diagnostic. These SunVTS documents are available online at the following URL:

`http://docs.sun.com/`

Search for title of the document you want to use.

Select the document for the Solaris release on your system.

TABLE B-1 SunVTS Documentation

Title	Description	Part Number
SunVTS 6.3 User's Guide	Describes the SunVTS diagnostic environment	820-0080
SunVTS 6.3 Test Reference Manual for SPARC Platforms	Describes each SunVTS test (including the `nettest` `netlbtest`) and describes the various test options and command-line arguments	820-0081
SunVTS 6.3 Test Reference Manual for x86 Platforms	Describes each SunVTS test (including the `nettest` `netlbtest`) and describes the various test options and command-line arguments	820-0082
SunVTS Quick Reference	Provides an overview of the user interface	820-0083

Updating SunVTS to Recognize the PCI Express Card

Use SunVTS 6.3 Patch Set 2 or later. You will need to update the SunVTS configuration to recognize the PCI-express card.

▼ To Update SunVTS to Recognize the PCI Express Card

1. **Plug in a loopback cable.**
2. **Ensure that the SunVTS software and the `nxge` driver are installed on your system.**
3. **Add the following lines to the `/opt/SUNWvts/lib/conf/netlbtest.conf` and `/opt/SUNWvts/lib/conf/nettest.conf` files:**

TABLE 0-1

nxge	**nxge**	**1gbaseT**

Using the SunVTS `netlbtest`

You must have the Ethernet card and the device driver installed, a loopback connector in place, and Intervention mode enabled before running `netlbtest`. `netlbtest` cannot run if the network interface is connected to a live network, and requires that the Ethernet device be configured offline before running the test. Use the `ifconfig(1M)` command to bring the Ethernet device down before running `netlbtest`.

▼ To Use the `netlbtest`

1. **Ensure that the SunVTS software and the `nxge` driver are installed on your system.**
2. **Plug in a loopback cable.**
3. **Unplumb the interface from the system, using the `ifconfig` command**:

TABLE 0-2

```
# ifconfig nxgeinstance down
# ifconfig nxgeinstance unplumb
```

Where *instance* is the instance number of the interface.

Refer to SunVTS documentation for instructions on how to run `netlbtest`.

APPENDIX C

Installing the Driver on Windows Systems

This appendix describes how to install the nxge driver for Microsoft Windows operating systems.

This appendix contains the following sections:

Overview

The nxge Windows Driver is an NDIS 5.1 miniport driver that is used to control the Sun Microsystems Sun Quad GbE UTP x8 PCI Express Card. The driver is compatible with the following operating systems:

- Windows XP Profession, SP2, 32-bit/64-bit.
- Windows Server 2003 Standard Edition, SP1, 32-bit/64-bit.

- Windows Server 2003 Enterprise Edition, SP1, 32-bit/64-bit.

Additionally, the driver supports the following features:

- VLAN
- Plug and Play
- Jumbo Frames
- IPv6
- Receive Frame classification
- Transmit Frame classification
- Checksum offload
  - Transmit: TCP, UDP
  - Receive: IP, TCP, UDP.

Software Components

Following is a list of all the software components that make up the nxge driver for Windows systems.

- **Install.exe**: The installation package which automatically installs the device drivers and associated software components.
- **NPMPort.sys**: The kernel mode driver used to control each port of the Atlas card.
- **NPNetEx.sys**: The kernel mode driver that is installed on the non-functional nodes of the 2XGF.
- **netSun.inf**: The information file used to install NPMPort.sys.
- **NPNetEx.inf**: The information file used to install NPNetEx.sys.
- **NPCoIns.dll**: A co-installer that is used to install drivers and associate property pages with the driver.
- **NPPropPg.dll**: The property page that is used to display statistics about each device node.

When the driver is successfully installed, these components are located in the following directory:

```
Programs Files\Sun Microsystems\Sun Network Drivers
```

Build Process

This section defines the required tools and libraries as well as instructing the user on how to build an installation package. At least some familiarity with Microsoft Visual C++ .NET is assumed.

Required Tools

- Microsoft Visual C++ .NET Version 7.1.3088.
- DDKBUILD.bat Version 3.13.

  DDKBuild.bat is a utility that integrates the DDK build process with Visual Studio. It must be copied into the \VC7\Bin directory that is associated with the install directory of Microsoft Visual Studio. DDKBuild.bat and complete instructions for how to use it can be found at: http://www.hollistech.com/Resources/ddkbuild/ddkbuild.htm.
- WinZip Profession version 11.1 with command line add-on.
- WinZip Self-Extractor 3.0.

Environment Variables

- WLHBASE: Set this variable to the root directory of the Windows Driver Kit (e.g.: C:\WinDDK\6000).
- WINZIP: Set this variable to the path of the WinZip installation directory (e.g.: C:\WinZip).
- WINZIPSE: Set this variable to the path of the WinZip Self-Extractor installation directory (e.g.: C:\WinZipSE).

Libraries

- Windows Driver Kit version 6000.

▼ To Build the Installation Package

1. Opening the Neptune Solution:

a. **Start Microsoft Visual C++ .NET and open the File menu.**

b. **Select "Open Solution" and then navigate to the location of Neptune.sln and double-click.**

2. **Select the configuration:**

   a. **Open the Build menu.**

   b. **Select "Configuration Manager" and select Release as the Active Solution Configuration.**

3. **Open the Build menu and select "Build Solution".**

   The installation package, `Install.exe`, builds and resides in the `Neptune\Release` directory.

Note – The install directories for Windows Driver Kit, WinZip Profession, and WinZip Self-Extractor must not contain any embedded spaces.
The directory of the `nxge` source code must not contain any embedded spaces.
After defining the environment variables, you must restart Microsoft Visual C++ .NET.

Installing and Uninstalling Drivers

The device drivers and associated components are packaged and distributed in one executable named `Install.exe`. The package contains both 32-bit and 64-bit drivers along with the associated software components.

Installing Drivers

Drivers are installed by simply executing Install.exe and following the prompts. The proper drivers will be selected and installed automatically. Once the application completes, the drivers are ready to use – there is no need to reboot the system.

Upgrading Drivers

To upgrade drivers, simple execute the latest version of `Install.exe`. This application will automatically update the system with the upgraded software components. Once the application completes, the drivers are ready to use – there is no need to reboot the system.

▼ To Uninstall Drivers

1. **Open the Control Panel Applet and select "Add or Remove Programs".**
2. **Scroll down the list of currently installed programs to locate "Sun Microsystems Network Drivers".**
3. **Select the "Remove" button at the far right hand side of the screen.**
4. **Ignore any and all "Found New Hardware" prompts that may appear as the drivers are being removed.**

The nxge drivers and all associated software components are removed from the system.

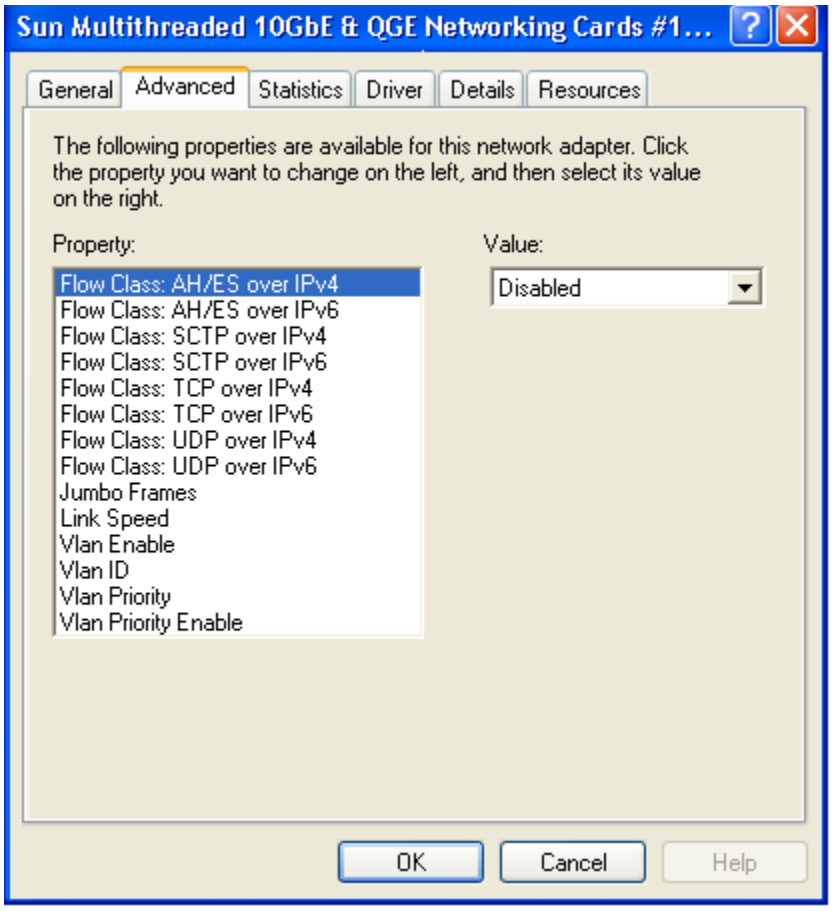
Custom Property Pages

The Atlas Windows Driver Package includes two custom property pages that are used for setting configuration parameters and display statistics. The property pages are accessible via the Device Manager.

Advanced Property Page

The Advanced Property Page contains the selectable configuration parameters for the Atlas Driver. FIGURE C-1 shows the format of the Property Page.

FIGURE C-1 Advanced Property Page



▼ To Access the Advanced Property Page

1. **Open the Device Manager.**
2. **Right click on an Driver Node (the Nodes are labeled as: "Sun Multithreaded 10GbE & QGE Networking Cards").**
3. **Select "Properties".**
4. **Select the "Advanced" Tab.**

The configuration properties are as follows:

TABLE C-1 Configurable Properties

Property Name	Description
Flow Class: AH/ES over IPv4	Enables receive flow classification of AH/ES network frames over IP version 4.
Flow Class: AH/ES over IPv6	Enables receive flow classification of AH/ES network frames over IP version 6.
Flow Class: SCTP over IPv4	Enables receive flow classification of SCTP network frames over IP version 4.
Flow Class: SCTP over IPv6	Enables receive flow classification of SCTP network frames over IP version 6.
Flow Class: TCP over IPv4	Enables receive flow classification of TCP network frames over IP version 4.
Flow Class: TCP over IPv6	Enables receive flow classification of TCP network frames over IP version 6.
Flow Class: UDP over IPv4	Enables receive flow classification of UDP network frames over IP version 4.
Flow Class: UDP over IPv6	Enables receive flow classification of UDP network frames over IP version 6.
Jumbo Frames	Enables the Atlas driver to transmit network frames that are 9200 bytes long
Link Speed & Duplex	Allows the user to select the link speed of the network connection. The selects include: 1. Auto-negotiate: Allows the link partners to select the appropriate speed. 2. 100 Mb Full Duplex: Sets the link speed to 100 Mb/sec. 3. 10 Mb Full Duplex: Sets the link speed to 10 Mb/sec.
Vlan Enable	Enables Vlan frames to be transmitted and received with the specified Vlan ID.
Vlan ID	Selects the Vlan Id. This ID will be inserted into all outgoing frames and all incoming frames will be filtered based on this ID.
Vlan Priority	Specified the priority value that will be inserted into outgoing Vlan tagged packets.
Vlan Priority Enable	Enables the specified priority to be inserted into outgoing Vlan tagged packets.

The Statistics Property Page

The Statistics Property page displays information about the current state of the network link. This information includes the number of packets transmitted, the number of packets received, the number of dropped packets, etc. The data is valuable for trouble shooting network issues. FIGURE C-2 shows the format of the Statistics Property Page.:

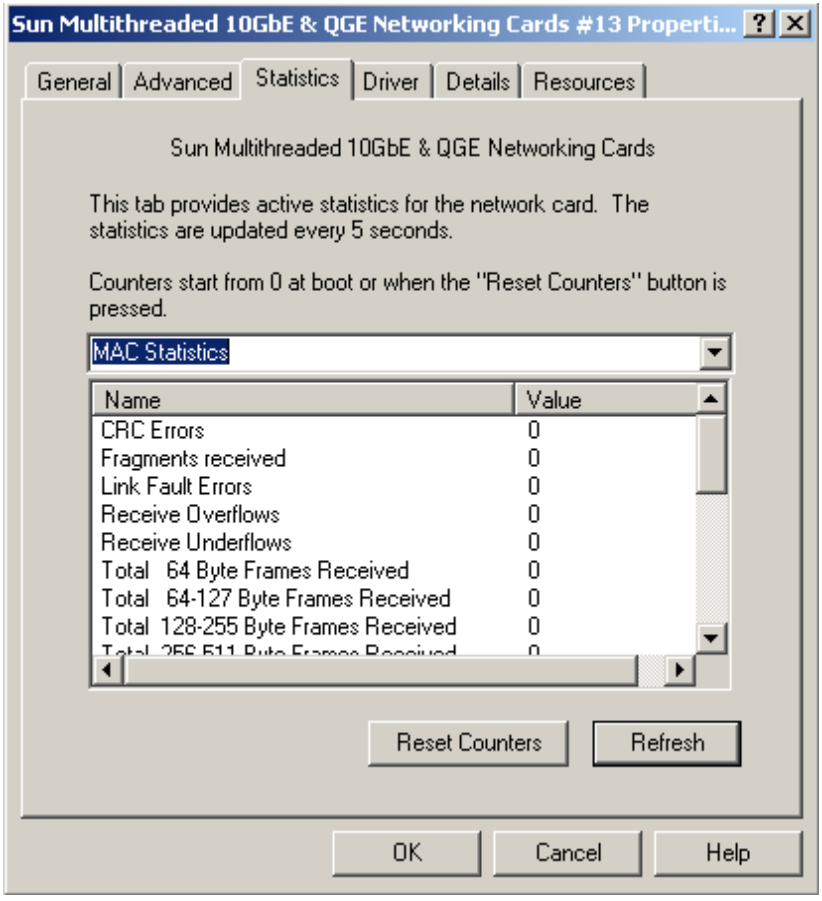


FIGURE C-2 Statistics Property Page

▼ To Access the Statistics Property Page

1. **Open the Device Manager.**
2. **Right Click on an Atlas Driver Node (the Nodes are labeled as: "Sun Multithreaded 10GbE & QGE Networking Cards").**
3. **Select "Properties".**
4. **Select the "Statistics" Tab.**

   The statistics are split up into three categories: MAC Statistics, Transmit Statistics and Receive Statistics. Each set of statistics is selectable via a drop down box at the top of the page.

MAC Statistics:

The MAC Statistics track the current state of the physical network link.

TABLE C-2 lists the available MAC statistics.

TABLE C-2 MAC Statistics

Name	Description	Notes
Alignment Errors	An alignment error is recorded when a received frame fails the checksum and contains a non-integer number of bytes.	Valid for Ports 3 and 4 only.
CRC Errors	A CRC is recorded when the received frame fails the CRC check.	
Fragments Received	This counter is updated each time a frame smaller than 64 bytes is received.	Valid for Ports 1 and 2 only.
Link Fault Errors	This counter is updated when a link fault is detected.	Valid for Ports 1 and 2 only.
Receive Overflows	This counter is updated when a packet can not be received because of lack of resources.	Valid for Ports 1 and 2 only.
Receive Underflows	This counter is updated when the receive FIFO is empty.	Valid for Ports 1 and 2 only.

TABLE C-2 MAC Statistics *(Continued)*

Name	Description	Notes
Total 64 Bytes Frames Received	This counter is updated whenever a frame of 64 bytes is received.	Valid for Ports 1 and 2 only.
Total 64-127 Byte Frames Received	This counter is updated whenever a frame, 64 – 127 bytes, is received.	Valid for Ports 1 and 2 only.
Total 128–255 Byte Frames Received	This counter is updated whenever a frame, 128 – 255 bytes, is received.	Valid for Ports 1 and 2 only.
Total 256–511 Byte Frames Received	This counter is updated whenever a frame, 256 – 511 bytes, is received.	Valid for Ports 1 and 2 only.
Total 512–1023 Byte Frames Received	This counter is updated whenever a frame, 512 – 1023 bytes, is received.	Valid for Ports 1 and 2 only.
Total 1024–1522 Byte Frames Received	This counter is updated whenever a frame, 1024 – 1522 bytes, is received.	Valid for Ports 1 and 2 only.
Total Broadcast Frames Received	This counter is updated whenever a broadcast frame is received.	Valid for Ports 1 and 2 only.
Total Bytes Received	This counter indicates the total number of bytes received.	
Total Bytes Transmitted	This counter indicates the total number of bytes transmitted.	
Total Code Violation Errors	This counter is updated when a frame containing an invalid encoding is received.	
Total Frames Received	This counter is updates to indicate the total number of frames received.	
Total Jumbo Frames Received	This counter is updated to indicate the total number of Jumbo frames received.	Valid for Ports 1 and 2 only.
Total Maximum Packet Length Errors	This counter is updated when a frames is received that exceeds the maximum allowable size.	
Total Multicast Frames Received	This counter indicates the total number of multicast frames received.	Valid for Ports 1 and 2 only.
Total Packet Sent	This counter indicates the total number of frames transmitted.	
Total Transmit Overflows	This counter will be updated whenever a protocol error, resulting in a transmit FIFO overflow, is detected.	Valid for Ports 1 and 2 only.
Total Transmit Underflows	This counter will be updated whenever a transmit data stream is interrupted resulting is a data starvation error.	Valid for Ports 1 and 2 only.

Receive DMA Statistics

The Receive DMA Statistics track the current state of the receive DMA channels. The following list of statistics is available.

TABLE C-3 Receive DMA Statistics

Name	Description	Notes
Byte Enable Bus Errors	This counter is incremented when an internal PCI bus error is detected.	*This is a fatal error.
Configuration Page Errors	This counter is incremented when a logical page violation is detected.	*This is a fatal error.
Data/Control FIFO Errors	This counter is incremented when a data or control FIFO ECC error is detected.	*This is a fatal error.
Dropped Packets	This counter is incremented whenever a received packet is dropped due to a lack of resources.	This is not an error. However excessive number of dropped packets can greatly reduce performance.
Number RBR Pre-Fetch Empty	This counter is incremented whenever receive buffer ring is empty.	This is not an error. However, excessive number of pre-fetch empty errors will greatly reduce performance.
Number RCR Shadow Parity Errors	This counter is incremented whenever a parity error is detected accessing RCR memory.	*This is a fatal error.
Outstanding Receive Packets	This counter is incremented whenever a packet has been passed up the protocol stack but not returned.	
Packets Received DMA Channel n	This counter indicates the total number of frames received on DMA channel “n”.	
Pre-fetch parity Errors	This counter is incremented whenever a parity error is detected pre-fetching receive buffers.	*This is a fatal error.
RBR Empty Errors	This counter is incremented whenever a received frame is dropped because there is no receive buffer available.	This is not a fatal error. However, excessive number of RBR empty errors will greatly reduce performance.
RBR Full Errors	This counter is incremented when a receive buffer is added to a full receive buffer ring.	*This is a fatal error and usually indicates a coding error.

TABLE C-3 Receive DMA Statistics *(Continued)*

Name	Description	Notes
RBR Logical Page Errors	This counter is incremented when a logical page error is detected in the memory allocated for receive buffers.	*This is a fatal error.
RBR Timeouts	This counter is incremented when a time out is generated accessing the receive buffer ring.	*This is a fatal error.
RCR ACK Errors	This counter is incremented when a timeout is generated before an ACK is received.	*This is a fatal error.
RCR DCF Errors	This counter is incremented when a data or control FIFO error is detected within a RCR entry.	
RCR Errors	This counter is incremented when L3 or L4 checksum error are reported in RCR entries.	
RCR Full Errors	This counter is incremented when a RCR entry is being posted to a full queue.	
RCR Inconsistencies	This counter is incremented when the receive the receive queue length is zero but the RCR head/tail pointer are not equal.	*This is a fatal error.
RCR Shadow Full	This counter is incremented when the internal RCR ring is full.	*This is a fatal error.
Response Count Mismatches on internal bus	This counter is incremented when the response data count does not match the frame length.	*This is a fatal error.
WRED Packets Dropped	This counter is incremented when packets are dropped as a result of weighted random early discard.	This is not an error. However, excessive number of WRED packets dropped errors will greatly reduce performance.

* The only recovery from a fatal error is to reboot the machine.

Transmit DMA Statistics

The Transmit DMA Statistics track the current state of the transmit DMA channels. The following list of statistics is available.

TABLE C-4 Transmit DMA Statistics

Name	Description	Notes
Mailbox Errors	This counter is incremented when there is a mailbox update error.	*This is a fatal error.
NACK Packet Read Errors	This counter is incremented when there is a time out on the NACK signal while reading packet data.	*This is a fatal error.
NACK Pre-fetch Errors	This counter is incremented when there is a time out on the NACK signal while reading packet pre-fetch data.	*This is a fatal error.
Number Packet Size Errors	This counter is incremented when the frame size exceeds the hardware limitation.	*This is a fatal error.
Outstanding Packets	This counter is incremented every time a packet has been sent out onto the network but has not been completed yet.	
Packet Pointer Errors	This counter is incremented when a memory partition error is encountered.	*This is a fatal error.
Packets Transmitted DMA Channel “n”.	This counter is incremented every time a packet has been successfully transmitted on DMA channel “n”.	
Parity Errors	This counter is incremented when a parity error is detected on the transmit buffer memory.	*This is a fatal error.
Partition Violation Errors	This counter is incremented when a memory partition error is detected during initialization.	*This is a fatal error.

TABLE C-4 Transmit DMA Statistics *(Continued)*

Name	Description	Notes
Resource Errors	This counter is incremented when a memory resource error is encountered while trying to transmit a packet.	This is not a fatal error. However, an excessive number of Resource Errors can greatly reduce performance.
Ring Over Flow Errors	This counter is incremented when the transmit ring overflows while trying to transmit a packet.	While this error is not fatal, this condition can greatly reduce performance.
Total Gathers Used	This counter is incremented every time a gather entry is removed from the gather entry ring.	

* The only recovery from a fatal error is to reboot the machine.

Event Logging

In order to facility debugging driver issues in the field, support for event logging has been added to the driver. When an error during initialization is encountered, an event is logged to the System Event log. These errors are generally fatal in the sense that the driver will not be loaded for the particular port. Failing to load the driver will result in the port being displayed with a yellow exclamation mark within Device Manager.

All logged events can be viewed using the System Event View.

▼ To Access the Event Viewer

1. **From Administrative Tools select Event Viewer.**
2. **From the left hand pane, select “System”.**

FIGURE C-3 showsvthe System Event Log:

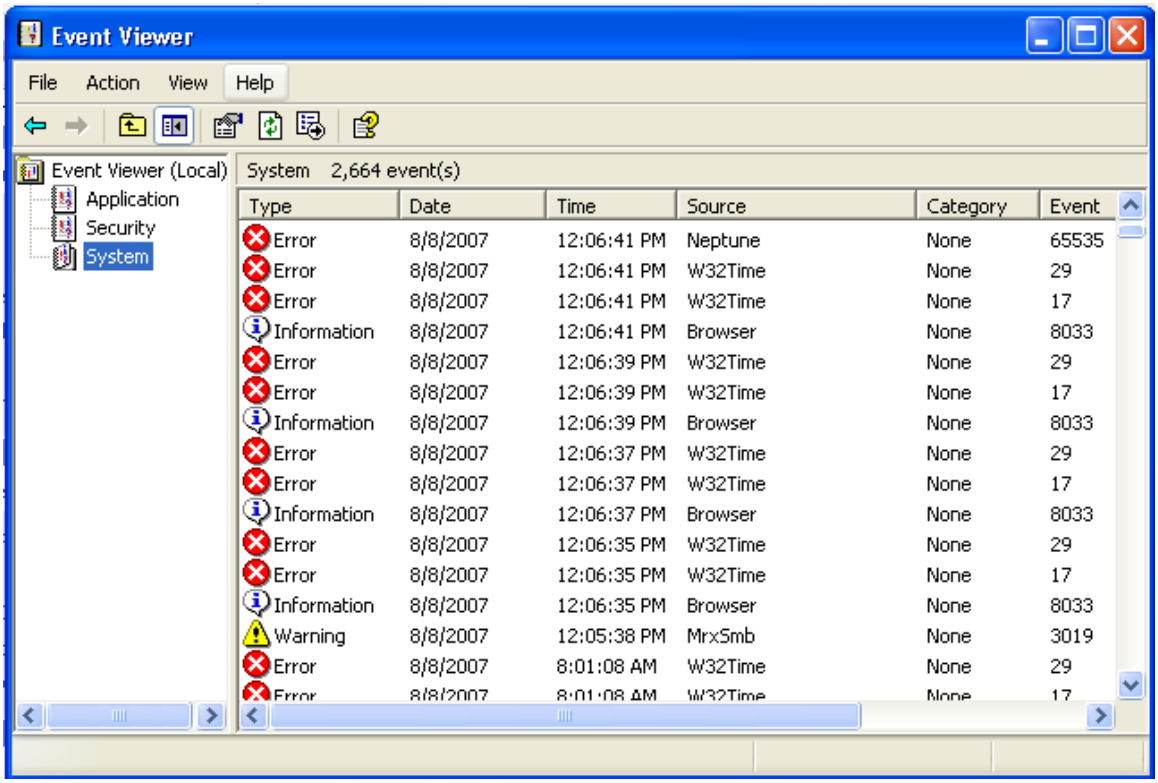


FIGURE C-3 System Event Log

All events that are relevant to the nxge driver are logged with a Source name of Neptune. Thus, in FIGURE C-3, event 65535 has been logged by the driver.

The values of the events and their meanings are defined in the following table:

TABLE C-5 Event Codes

Event (Value)	Description	Notes
Unsupported Media (65535)	The protocol stack does not support the media type used by Atlas.	Fatal Error.
Adapter Initialization Failed (65534)	Atlas failed to initialize the device context for this port.	Fatal Error.
Device Address Error (65533)	Unable to read the PCI address information for this port.	Fatal Error.

TABLE C-5 Event Codes *(Continued)*

Event (Value)	Description	Notes
Error Reading Configuration Parameters (65532)	Unable to read the configurable parameters for this port.	Fatal Error.
Failed to Initialize the DMA SGL (65531)	Unable to initialize the DMA Scatter/Gather List for this port.	Fatal Error.
Error Reading PCI Configuration Space (65530)	Unable to read the PCI configuration space for this port.	Fatal Error.
Adapter Resource Error (65529)	Unable to get the PCI resources for this port.	Fatal Error.
Error Reading Product Data (65528)	Unable to read the SEEPROM to obtain the Vital Product Data for this port.	Fatal Error.
Error no physical Port (65527)	There is no physical port for this PCI function.	Fatal Error.
Error Allocating Device Instance (65526)	Unable to allocate memory for this device instance.	Fatal Error.
Error Allocating Receive Memory (65525)	Unable to allocate memory to receive network packets.	Fatal Error.
Error Allocating Send Memory (65524)	Unable to allocate memory to transmit network packets.	Fatal Error.
Error Initializing Adapter Hardware (65523)	Failed to initialize the hardware for this port.	Fatal Error.
Error Initializing WDM Device (65522)	Failed to initialize the WDM device component for this system.	This is a non-fatal error however it will prevent the property pages from working correctly.
Error Registering Interrupt (65521)	Failed to register the interrupt for this port.	Fatal Error.
Error Allocating Function Instance (65520)	Failed to allocate the function instance for this port.	Fatal Error.